# AN617

# Fixed Point Routines

*Author:* *Frank J. Testa*
*FJT Consulting*

## INTRODUCTION

This application note presents an implementation of the following fixed point math routines for the PICmicro™ microcontroller families:

- Multiplication
- Division

Routines for the PICmicro microcontroller families are provided in a variety of fixed point formats, including both unsigned and signed two's complement arithmetic.

## FIXED POINT ARITHMETIC

Unsigned fixed point binary numbers, $A$, can be represented in the form

$$A = \sum_{k=0}^{n-1} a(k) \cdot 2^{k-r} = 2^{-r} \sum_{k=0}^{n-1} a(k) \cdot 2^{k}$$

where $n$ is the number of bits, $a(k)$ is the $k$th bit with $a(0)$ = LSb, and $r$ indicates the location of the radix point. For example, in the case where $A$ is an integer, $r$ = 0 and when $A$ is a fraction less than one, $r$ = n. The value of $r$ only affects the interpretation of the numbers in a fixed point calculation, with the actual binary representation of the numbers independent of the value of $r$. Factoring out of the above sum, it simply locates the radix point of the representation and is analogous to an exponent in a floating point system.

Using the notation Qi.j to denote a fixed point binary number with i bits to the left of the radix point and j to the right, the above n-bit format is in Qn-r.r. With care, fixed point calculations can be performed on operands in different Q formats. Although the radix point must be aligned for addition or subtraction, multiplication provides an illustrative example of the simple interpretive nature of r. Consider the unsigned product of a Q20.4 number with a Q8.8. After calling the appropriate unsigned 24x16 bit multiply for these fixed point arguments, the 40-bit fixed point result is in Q28.12, where the arguments of the Q notation are summed respectively.

Similar arguments can be made for two's complement arithmetic, where the negative representation of a positive number is obtained by reversing the value of each bit and incrementing the result by one. Producing a unique representation of zero, and covering the range $-2^{n-1}$ to $2^{n-1} - 1$, this is more easily applied in addition and subtraction operations and is therefore the most commonly used method of representing positive and negative numbers in fixed point arithmetic.

The above analysis in Q notation can be employed to build dedicated fixed point algorithms, leading to improved performance over floating point methods in cases where the size of the arguments required for the range and precision of the calculations is not large enough to destroy gains made by fixed point methods.

## FIXED POINT FORMATS

The fixed point library routines supports 8-,16-, 24- and 32-bit formats in the combinations shown in Table 1.

These general format combinations are implemented in both signed and unsigned versions. Additional unsigned routines are implemented with arguments reduced by one bit to accommodate the case of operations on signed numbers, with arguments known to be nonnegative, thereby, resulting in some performance improvement.

**TABLE 1: FIXED POINT LIBRARY ROUTINE SUMMARY TABLE**

| Division Library Names | Format | Multiplication Library Names | Format |
|---|---|---|---|
| **PIC16C5X/PIC16CXXX Routines** | | | |
| FXD0808S, FXD0808U, FXD0807U, FXD0707U | 8/8 | FXM0808S, FXM0808U, FXM0807U | 8x8 |
| FXD1608S, FXD1608U, FXD1607U, FXD1507U | 16/8 | FXM1608S, FXM1608U, FXM1607U, FXM1507U | 16x8 |
| FXD1616S, FXD1616U, FXD1515U | 16/16 | FXM1616S, FXM1616U, FXM1515U | 16x16 |
| FXD2416S, FXD2416U, FXD2315U | 24/16 | FXM2416S, FXM2416U, FXM2315U | 24x16 |
| FXD2424S, FXD2424U, FXD2323U | 24/24 | FXM2424S, FXM2424U, FXM2323U | 24x24 |
| FXD3216S, FXD3216U, FXD3115U | 32/16 | FXM3216S, FXM3216U, FXM3115U | 32x16 |
| FXD3224S, FXD3224U, FXD3123U | 32/24 | FXM3224S, FXM3224U, FXM3123U | 32x24 |
| FXD3232S, FXD3232U, FXD3131U | 32/32 | FXM3232S, FXM3232U, FXM3131U | 32x32 |
| **PIC17CXXX Functions** | | | |
| FXD0808S, FXD0808U, FXD0807U, FXD0707U | 8/8 | FXM0808S, FXM0808U | 8x8 |
| FXD1608S, FXD1608U, FXD1607U, FXD1507U | 16/8 | FXM1608S, FXM1608U | 16x8 |
| FXD1616S, FXD1616U, FXD1615U, FXD1515U | 16/16 | FXM1616S, FXM1616U | 16x16 |
| FXD2416S, FXD2416U, FXD2415U, FXD2315U | 24/16 | FXM2416S, FXM2416U | 24x16 |
| FXD2424S, FXD2424U, FXD2423U, FXD2323U | 24/24 | FXM2424S, FXM2424U | 24x24 |
| FXD3216S, FXD3216U, FXD3215U, FXD3115U | 32/16 | FXM3216S, FXM3216U | 32x16 |
| FXD3224S, FXD3224U, FXD3223U, FXD3123U | 32/24 | FXM3224S, FXM3224U | 32x24 |
| FXD3232S, FXD3232U, FXD3231U, FXD3131U | 32/32 | FXM3232S, FXM3232U | 32x32 |

Note: U - unsigned math operation, S - signed math operation

## DATA RAM REQUIREMENTS

Table 2 shows the contiguous data RAM locations that are used by the library.

**TABLE 2: DATA RAM REQUIREMENTS**

```
AARGB7    =   REMB3
AARGB6    =   REMB2
AARGB5    =   REMB1
AARGB4    =   REMB0          remainder MSB
AARGB3
AARGB2
AARGB1
AARGB0                       AARG MSB
AEXP                         AARG exponent
SIGN                         sign
FPFLAGS                      exception flags and option bits
BARGB3
BARGB2
BARGB1
BARGB0                       BARG MSB
BEXP                         BARG exponent
TEMPB3
TEMPB2
TEMPB1
TEMPB0                       temporary storage
```

These definitions are identical with those used by the IEEE 754 compliant floating point library[5], AN575.

## USAGE

Multiplication assumes the multiplicand in AARG, multiplier in BARG, and produces the result in AARG. Division assumes a dividend in AARG, divisor in BARG, and quotient in AARG with remainder in REM.

## ADDITION/SUBTRACTION

Because of the generally trivial nature of addition and subtraction, the call and return overhead outweighs the need for explicit routines and so they are not included in the library. However, the PIC16C5X/PIC16CXXX families do not have an add with carry or subtract with borrow instruction, leading to subtleties regarding production of a correct carry-out in a multiple byte add or subtract. In the case of a two byte add or subtract, the most elegant solution to these difficulties, requiring 6 cycles, appears to be given by the following code in Example 1.

**EXAMPLE 1: TWO BYTE ADDITION/SUBTRACTION ROUTINES**

```
ADD     MOVF    AARGB1,W
        ADDWF   BARGB1

        MOVF    AARGB0,W
        BTFSC   _C
        INCFSZ  AARGB0,W
        ADDWF   BARGB0

SUB     MOVF    AARGB1,W
        SUBWF   BARGB1

        MOVF    AARGB0,W
        BTFSS   _C
        INCFSZ  AARGB0,W
        SUBWF   BARGB0
```

The four instructions after the initial add/subtract, can be easily concatenated for operations involving more than two bytes. Because addition and subtraction are required in standard algorithms for multiplication and division, these issues permeate the implementation of both fixed and floating point algorithms for the PIC16C5X/PIC16CXXX families.

## MULTIPLICATION

The existing library of fixed point math routines for the PICmicro families of microcontrollers contains multiplication routines in the following format combinations:

- 8x8
- 16x8
- 16x16
- 24x16
- 24x24
- 32x16
- 32x24
- 32x32

The fixed point multiply routine FXMxxyy, takes an xx-bit multiplicand in AARG, a yy-bit multiplier in BARG and returns the (xx+yy)-bit product in AARG.

For the PIC17 family, both unsigned and signed algorithms use extended precision application of the 8x8 hardware multiply currently available. The essence of an extended precision interpretation is to view each argument as a concatenation of bytes of different orders of magnitude and evaluate the product by evaluating all 8x8 terms in the algebraic expansion. For example, the 24x16 multiply yields a 40-bit product and contains 6 individual 8x8 terms in its expansion.

$$(AARGB0\cdot 2^{16}+AARGB1\cdot 2^{8}+AARGB2\cdot 2^{0})\cdot$$
$$(BARGB0\cdot 2^{8}+BARGB1\cdot 2^{0}) =$$
$$AARGB0\cdot BARGB0\cdot 2^{24} +$$
$$(AARGB0\cdot BARGB1+AARGB1\cdot BARGB0)\cdot 2^{16} +$$
$$(AARGB1\cdot BARGB1+AARGB2\cdot BARGB0)\cdot 2^{8} +$$
$$AARGB2\cdot BARGB1\cdot 2^{0}$$

This is completely analogous to arithmetic in base $2^8 = 256$, where the respective digit products must be aligned according to their orders of magnitude before summation. It is important to note that no carryout beyond the sum of the lengths of the arguments can occur[1]. This fact is helpful in constructing algorithms for cases with a large number of terms. For example, the 32x16 case containing 8 individual 8x8 terms, can be viewed as a 24x16 product between the 3 least significant bytes of AARG with BARG, producing no carryout, followed by augmentation with the remaining two terms. This philosophy has been applied in optimizing the unsigned algorithms, using the shorter products as building blocks for the larger ones.

The signed fixed point multiply routines require proper handling of the fact that the 8x8 hardware multiply is unsigned. It can be proven (see Appendix C) that the product of signed numbers in two's complement representation can be obtained by computing their product as if they were unsigned and for each negative argument, subtract the opposite argument from the most significant bits of the product. In most cases, the optimal algorithm is to simply apply this at the end of the corresponding unsigned method to achieve the signed product.

The implementation for the PIC16CXXX family uses a standard sequential add-shift algorithm, negating both factors if BARG < 0, to produce the positive multiplier required by the method. Analogous to simple longhand binary multiplication, the multiplier bits are sequentially tested, with one indicating an add-shift and zero simply a shift. The shift is required to align the partial product for the next possible add[1]. Two examples are shown in Example 2.

## EXAMPLE 2: MULTIPLICATION EXAMPLES

FXM2416S(0xC11682,0x608B)

= FXM2416S(-4123006,24715)

= 0xE84647F896

= -101900093290

FXM1616U(0x0458,0x822C)

= FXM1616U(1112,33324)

= 0x02356F20

= 37056288

Table 3 shows PIC17CXXX Fixed Point multiplication performance data. The listed routines can be found in Appendix F.

**TABLE 3: PIC17CXXX FIXED POINT MULTIPLY PERFORMANCE DATA**

| Routine | Max Cycles | Min Cycles | Average Cycles | Program Memory | Data Memory |
|---|---|---|---|---|---|
| FXM0808S | 14 | 14 | 14 | 10 | 3 |
| FXM0808U | 9 | 9 | 9 | 5 | 3 |
| FXM1608S | 24 | 21 | 23 | 20 | 4 |
| FXM1608U | 15 | 15 | 15 | 11 | 4 |
| FXM1616S | 42 | 34 | 38 | 38 | 8 |
| FXM1616U | 29 | 29 | 29 | 25 | 7 |
| FXM2416S | 59 | 49 | 54 | 55 | 10 |
| FXM2416U | 43 | 43 | 43 | 39 | 8 |
| FXM2424S | 84 | 72 | 78 | 80 | 12 |
| FXM2424U | 68 | 68 | 68 | 64 | 12 |
| FXM3216S | 76 | 64 | 70 | 72 | 12 |
| FXM3216U | 57 | 57 | 57 | 53 | 9 |
| FXM3224S | 111 | 97 | 104 | 107 | 15 |
| FXM3224U | 93 | 93 | 93 | 89 | 15 |
| FXM3232S | 148 | 132 | 140 | 144 | 18 |
| FXM3232U | 128 | 128 | 128 | 124 | 18 |

Table 4 shows the PIC16C5X/PIC16CXXX Fixed Point Multiply performance data. The listed routines can be found in Appendix D.

**TABLE 4: PIC16C5X/PIC16CXXX FIXED POINT MULTIPLY PERFORMANCE DATA**

| Routine | Max Cycles | Min Cycles | Average Cycles | Program Memory | Data Memory |
|---|---|---|---|---|---|
| FXM0808S | 91 | 9 | 85 | 33 | 5 |
| FXM0808U | 76 | 57 | 74 | 21 | 4 |
| FXM0707U | 70 | 51 | 67 | 23 | 4 |
| FXM1608S | 110 | 11 | 85 | 44 | 7 |
| FXM1608U | 129 | 61 | 105 | 31 | 7 |
| FXM1507U | 86 | 27 | 64 | 35 | 7 |
| FXM1616S | 284 | 11 | 235 | 74 | 9 |
| FXM1616U | 259 | 110 | 214 | 58 | 9 |
| FXM1515U | 247 | 105 | 205 | 63 | 9 |
| FXM2416S | 353 | 132 | 281 | 92 | 12 |
| FXM2416U | 328 | 113 | 260 | 70 | 12 |
| FXM2315U | 321 | 108 | 248 | 76 | 12 |
| FXM2424S | 533 | 241 | 432 | 126 | 13 |
| FXM2424U | 497 | 258 | 401 | 98 | 13 |
| FXM2323U | 481 | 230 | 390 | 107 | 13 |
| FXM3216S | 440 | 48 | 327 | 98 | 9 |
| FXM3216U | 415 | 116 | 304 | 84 | 9 |
| FXM3115U | 395 | 111 | 291 | 91 | 9 |
| FXM3224S | 656 | 253 | 502 | 152 | 15 |
| FXM3224U | 620 | 201 | 470 | 151 | 15 |
| FXM3123U | 587 | 255 | 457 | 129 | 15 |
| FXM3232S | 841 | 411 | 686 | 189 | 17 |
| FXM3232U | 794 | 443 | 645 | 168 | 17 |
| FXM3131U | 787 | 392 | 631 | 168 | 17 |

## DIVISION

The fixed point divide routine FXPDxxyy, takes an xx-bit dividend in AARG, a yy-bit divisor in BARG and returns the xx-bit quotient in AARG and yy-bit remainder in REM. Unlike multiplication, division is not deterministic, requiring a trial-and-error sequential shift and subtract process. Binary division is less complicated than decimal division because the possible quotient digits are only zero or one. If the divisor is less than the partial remainder, the corresponding quotient bit is set to one followed by a shift and subtract. Otherwise, the divisor is greater than the partial remainder, the quotient bit is set to zero and only a shift is performed. The intermediate partial remainder may be restored at each stage as in restoring division, or corrected at the end as in nonrestoring division. Implementation dependent trade-offs between worst case versus average performance affect the choice between these two approaches, and therefore, macros for each method are provided.

| **Note:** | A test for divide by zero exception is not performed and must be explicitly provided by the user. |
|---|---|

The results of the division process for AARG/BARG, satisfy the relation

AARG = BARG · QUOTIENT + REMAINDER,

where the remainder has the same sign as the quotient, and represents the fraction of the result in units of the denominator BARG. Some simple examples are given in Example 3.

### EXAMPLE 3: DIVISION EXAMPLES

FXD1608S(0xC116,0x60) = 0xFF59, 0xB6

FXD1616U(0x9543,0x4AA1) = 0x0002, 0x0001

Table 5 shows the PIC17CXXX Fixed Point Divide performance data. The listed routines can be found in Appendix G

**TABLE 5: PIC17CXXX FIXED POINT DIVIDE PERFORMANCE DATA**

| Routine | Max Cycles | Min Cycles | Average Cycles | Program Memory | Data Memory |
|---|---|---|---|---|---|
| FXD0808S | 91 | 85 | 89 | 77 | 4 |
| FXD0808U | 78 | 74 | 77 | 74 | 3 |
| FXD0807U | 69 | 69 | 69 | 65 | 3 |
| FXD0707U | 64 | 64 | 64 | 60 | 3 |
| FXD1608S | 162 | 44 | 156 | 146 | 5 |
| FXD1608U | 196 | 170 | 183 | 195 | 4 |
| FXD1607U | 133 | 133 | 133 | 129 | 4 |
| FXD1507U | 128 | 128 | 128 | 124 | 4 |
| FXD1616S | 219 | 200 | 211 | 241 | 7 |
| FXD1616U | 247 | 227 | 244 | 243 | 6 |
| FXD1615U | 188 | 182 | 184 | 216 | 6 |
| FXD1515U | 182 | 177 | 179 | 218 | 6 |
| FXD2416S | 315 | 291 | 305 | 353 | 8 |
| FXD2416U | 352 | 342 | 347 | 453 | 8 |
| FXD2415U | 283 | 272 | 277 | 339 | 8 |
| FXD2315U | 275 | 266 | 270 | 330 | 8 |
| FXD2424S | 387 | 361 | 377 | 482 | 10 |
| FXD2424U | 422 | 415 | 419 | 577 | 10 |
| FXD2423U | 352 | 344 | 347 | 460 | 9 |
| FXD2323U | 344 | 337 | 341 | 448 | 9 |
| FXD3216S | 415 | 382 | 400 | 476 | 9 |
| FXD3216U | 468 | 459 | 463 | 608 | 9 |
| FXD3215U | 375 | 363 | 369 | 451 | 8 |
| FXD3115U | 368 | 357 | 362 | 442 | 8 |
| FXD3224S | 514 | 477 | 496 | 639 | 11 |
| FXD3224U | 566 | 553 | 560 | 769 | 11 |
| FXD3223U | 476 | 459 | 465 | 612 | 10 |
| FXD3123U | 466 | 451 | 457 | 600 | 10 |
| FXD3232S | 610 | 572 | 593 | 800 | 13 |
| FXD3232U | 665 | 650 | 655 | 930 | 13 |
| FXD3231U | 567 | 555 | 560 | 773 | 12 |
| FXD3131U | 558 | 547 | 552 | 758 | 12 |

Table 6 shows the PIC16C5X/PIC16CXXX Fixed Point Divide performance data. The listed routines can be found in Appendix E.

**TABLE 6: PIC16C5X/PIC16CXXX FIXED POINT DIVIDE PERFORMANCE DATA**

| Routine | Max Cycles | Min Cycles | Average Cycles | Program Memory | Data Memory |
|---|---|---|---|---|---|
| FXD0808S | 131 | 36 | 109 | 41 | 5 |
| FXD0808U | 103 | 95 | 102 | 15 | 4 |
| FXD0807U | 91 | 91 | 91 | 21 | 4 |
| FXD0707U | 83 | 83 | 83 | 44 | 4 |
| FXD1608S | 181 | 49 | 159 | 67 | 6 |
| FXD1608U | 297 | 237 | 269 | 41 | 7 |
| FXD1607U | 177 | 177 | 177 | 41 | 5 |
| FXD1507U | 169 | 169 | 169 | 44 | 5 |
| FXD1616S | 334 | 302 | 315 | 74 | 8 |
| FXD1616U | 376 | 316 | 371 | 27 | 7 |
| FXD1515U | 292 | 277 | 280 | 45 | 7 |
| FXD2416S | 447 | 408 | 427 | 140 | 8 |
| FXD2416U | 524 | 504 | 510 | 172 | 8 |
| FXD2315U | 402 | 382 | 388 | 120 | 7 |
| FXD2424S | 570 | 528 | 549 | 253 | 12 |
| FXD2424U | 641 | 624 | 630 | 226 | 13 |
| FXD2323U | 520 | 502 | 508 | 211 | 12 |
| FXD3216S | 584 | 67 | 561 | 201 | 10 |
| FXD3216U | 694 | 671 | 680 | 243 | 9 |
| FXD3115U | 534 | 509 | 518 | 160 | 9 |
| FXD3224S | 747 | 695 | 722 | 280 | 11 |
| FXD3224U | 853 | 830 | 838 | 299 | 11 |
| FXD3123U | 692 | 668 | 676 | 232 | 10 |
| FXD3232S | 909 | 855 | 885 | 357 | 13 |
| FXD3232U | 1012 | 990 | 998 | 364 | 13 |
| FXD3131U | 851 | 828 | 836 | 304 | 13 |

## REFERENCES


1. Cavanagh, J.J.F., “Digital Computer Arithmetic,” McGraw-Hill,1984.
2. Hwang, K., “Computer Arithmetic,” John Wiley & Sons, 1979.
3. Scott, N.R., “Computer Number Systems & Arithmetic,” Prentice Hall, 1985.
4. Knuth, D.E., “The Art of Computer Programming, Volume 2,” Addison-Wesley, 1981.
5. F.J.Testa, “IEEE 754 Compliant Floating Point Routines,” AN575, Embedded Control Handbook, Microchip Technology Inc., 1995.

## APPENDIX A: ALGORITHMS

Several algorithms for decimal to binary conversion are given below. The integer and fractional conversion algorithms are useful in both native assembly as well as high level languages.

### A.1 Integer conversion algorithm[3]:

Given an integer ***I***, where ***d(k)*** are the bit values of its ***n***- bit binary representation with ***d(0)*** = LSB,

$$I = \sum_{k=0}^{n-1} d(k) \cdot 2^{k}$$

```
k=0
I(k) = I
while I(k) = ! 0
   d(k) = remainder of I(k)/2
   I(k+1) = ⌈I(k)/2⌉
   k = k + 1
endw
```

where ⌈ ⌉ denotes the greatest integer function (or ceiling function).

### A.2 Fractional conversion algorithm[3]:

Given a fraction ***F***, where ***d(k)*** are the bit values of its n bit binary representation with ***d(1)*** = MSB,

$$F = \sum_{k=1}^{n} d(k) \cdot 2^{-k}$$

```
k=0
F(k) = F
while k <= n
   d(k) = ⌈F(k)· 2⌉
   F(k+1) = fractional part of F(k) · 2
   k = k + 1
endw
```

## APPENDIX B: FLOWCHARTS

**FIGURE B-1: MULTIPLICATION FLOWCHART FOR PIC16CXX**

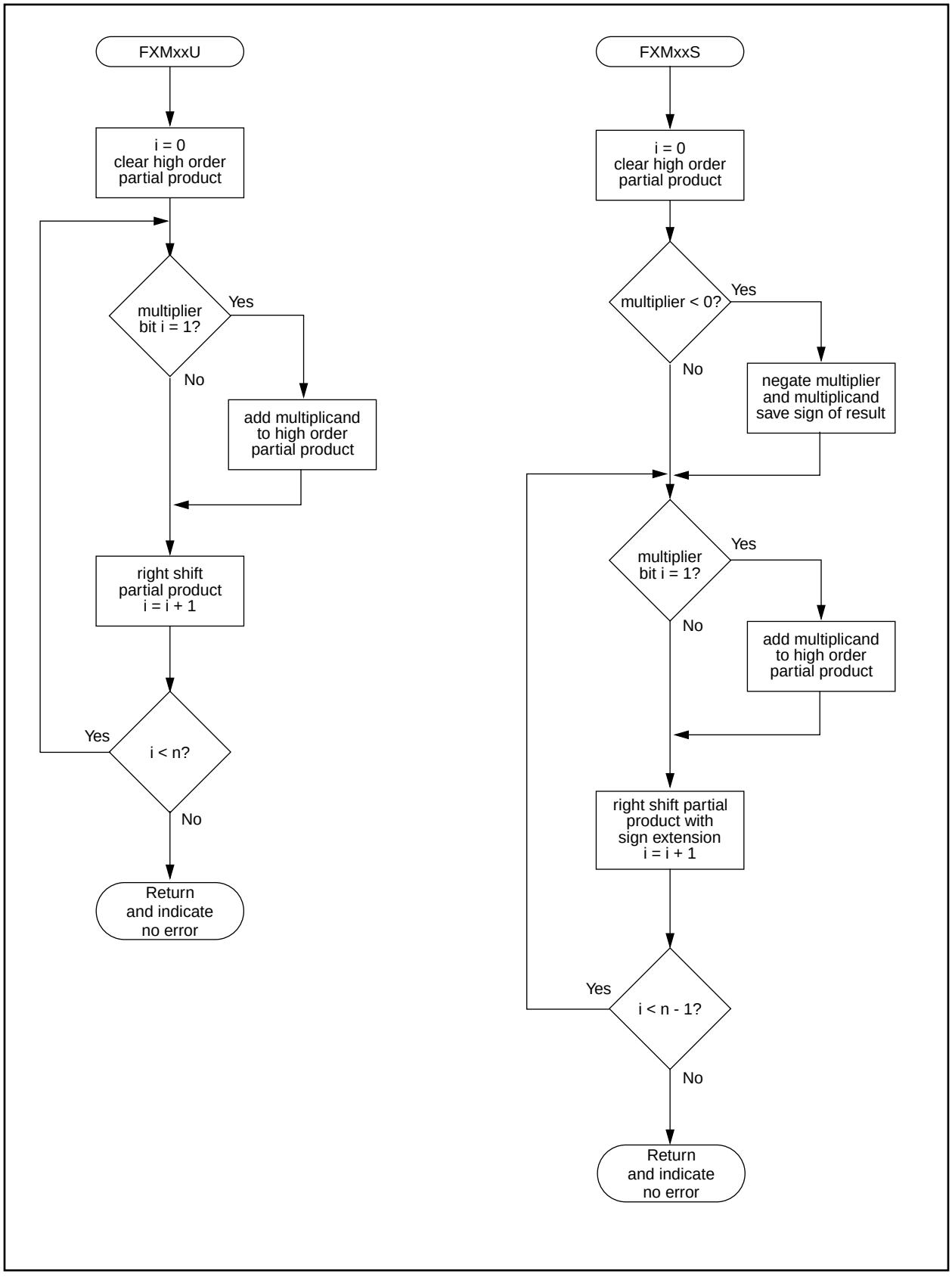


**FIGURE B-2: MULTIPLICATION FLOWCHART FOR PIC17CXXX**

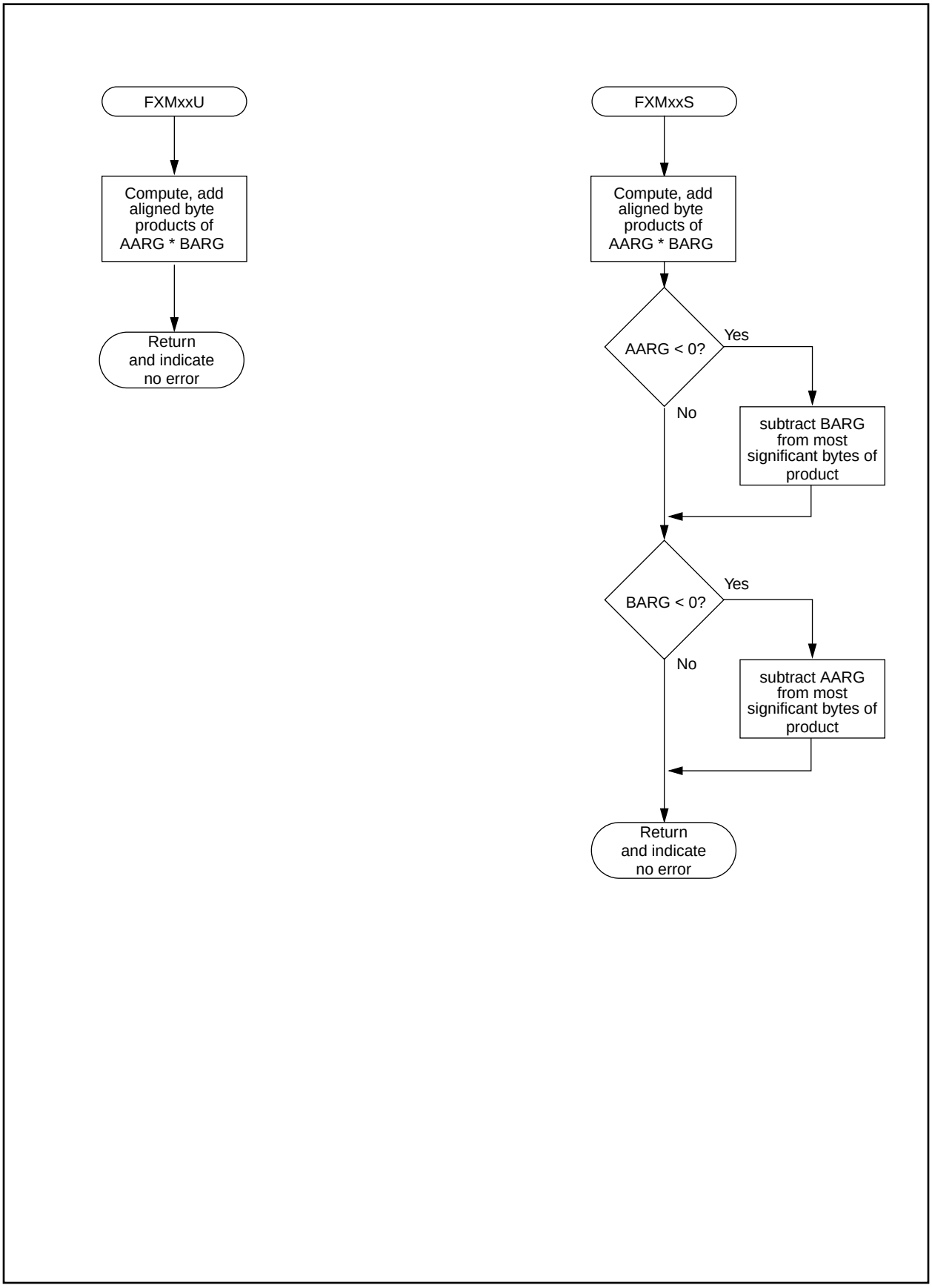
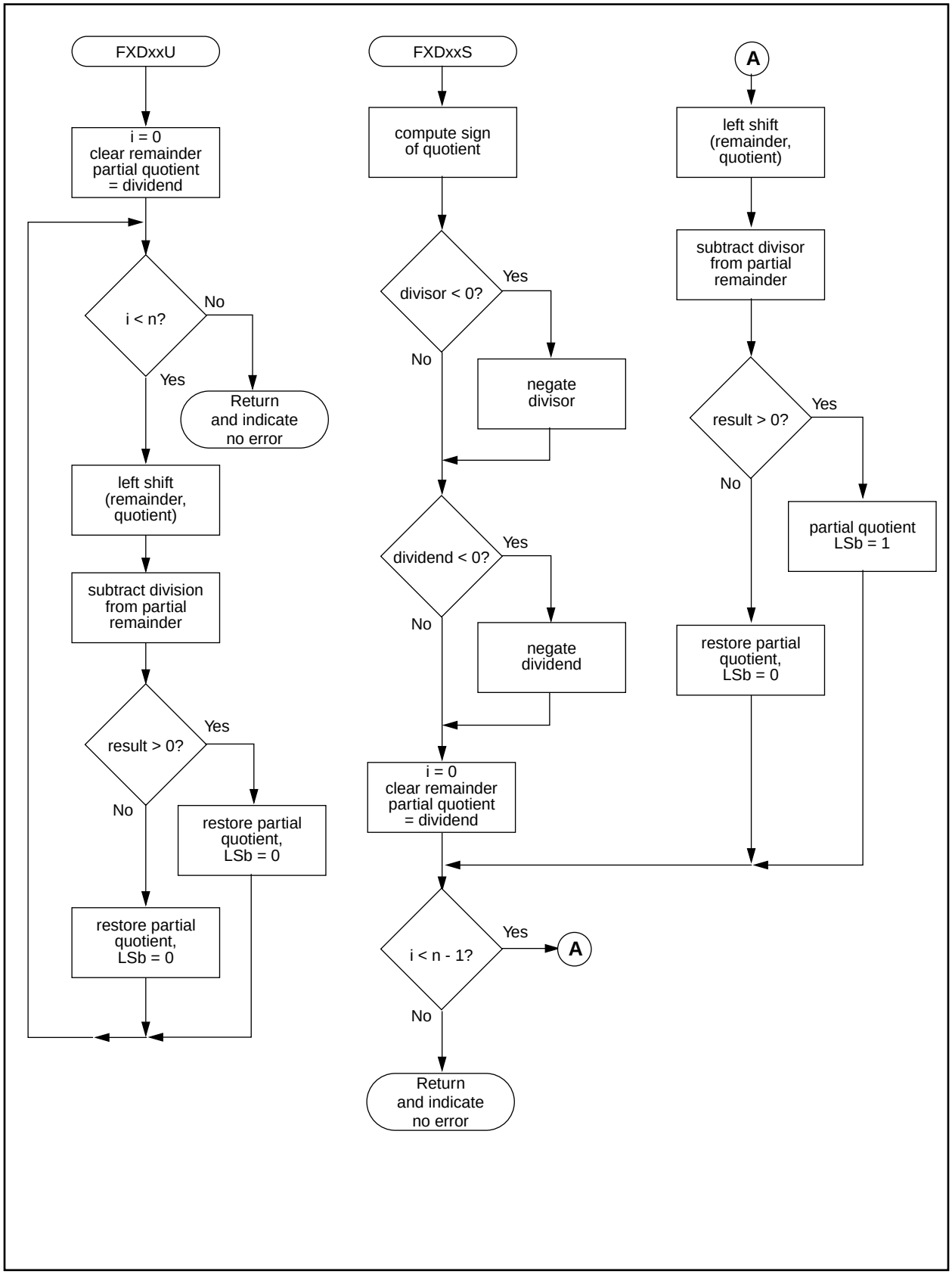


**FIGURE B-3: DIVISION FLOWCHART**



## APPENDIX C:

Consider arguments to a two's complement multiply expressed in the form

$$A = -a_{m-1}2^m + A_u \quad , \qquad B = -b_{n-1}2^n + B_u$$

where

$$A_u \equiv \sum_{i=0}^{m-1} a_i 2^i \quad , \qquad B_u \equiv \sum_{i=0}^{n-1} b_i 2^i$$

Then

$$p = A \cdot B = p' + c$$

with

$$p' \equiv A_u \cdot B_u$$

being the unsigned product of the two's complement representations and the correction term **c** given by

$$c \equiv a_{m-1} b_{n-1} 2^{m+n} - \{ a_{m-1} B_u 2^m + b_{n-1} A_u 2^n \}$$

**Case 1:**

$$a_{m-1} = b_{n-1} = 1$$

For this case with both arguments negative we obtain

$$c = 2^{m+n} - \{ B_u 2^m + A_u 2^n \}$$

where

$$2^{n-1} \le B_u < 2^n$$

and

$$2^{m-1} \le A_u < 2^m$$

yielding the bounds

$$2^{m+n} \le B_u 2^m + A_u 2^n < 2^{m+n+1} \quad .$$

The bounded quantity

$$B_u 2^m + A_u 2^n$$

can then be expressed in the form

$$B_u 2^m + A_u 2^n \equiv 2^{m+n} + T_u \quad ,$$

where $T_u < 2^{m+n}$ is the result of truncating the above bounded quantity to m+n bits. This gives the final value for the correction term in the form

$$c = -T_u$$

where the

$$2^{m+n}$$

term has been cancelled by the carry during the evaluation of

$$B_u 2^m + A_u 2^n$$

leading to the result

$$p = p' + c = p' - Tu$$

**Case 2:**

$$a_{m-1} \cdot b_{n-1} = 0$$

The case with one or both arguments positive gives the simpler result

$$c = -\{ a_{m-1} B_u 2^m + b_{n-1} A_u 2^n \}$$

where

and therefore a carry out is not possible.

$$|c| < 2^{m+n}$$

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

## APPENDIX D: MULTIPLY ROUTINES FOR THE PIC16C5X/PIC16CXXX

**Table of Contents for Appendix D**



### D.1 32x32 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;        RCS Header $Id: fxm22.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;        $Revision: 2.3 $

;       32x32 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM3232S     889        32x32 -> 64 bit signed fixed point multiply
;
;       FXM3232U     856        32x32 -> 64 bit unsigned fixed point multiply
;
;       FXM3131U     836        31x31 -> 62 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 128-168 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       32x32 Bit Multiplication Macros

SMUL3232L        macro

;       Max Timing:     2+13+6*26+25+2+7*27+26+2+7*28+27+2+6*29+28+9 = 851 clks

;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+2+6*6+5+6 = 192 clks

;       PM: 31+25+2+26+2+27+2+28+9 = 152            DM: 17


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM3232A
                RRF             BARGB3, F
                BTFSC           _C
```

```
                GOTO            ALSM3232NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3232A

                MOVWF           LOOPCOUNT

LOOPSM3232B
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            BLSM3232NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3232B

                MOVWF           LOOPCOUNT

LOOPSM3232C
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            CLSM3232NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3232C

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPSM3232D
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            DLSM3232NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3232D

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALOOPSM3232
                RRF             BARGB3, F
                BTFSS           _C
                GOTO            ALSM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALSM3232NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPSM3232
```

```
                MOVLW           0x8
                MOVWF           LOOPCOUNT

BLOOPSM3232
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            BLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLSM3232NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPSM3232

                MOVLW           0x8
                MOVWF           LOOPCOUNT

CLOOPSM3232
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            CLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLSM3232NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPSM3232

                MOVLW           0x7
```

```
                MOVWF           LOOPCOUNT

DLOOPSM3232
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            DLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

DLSM3232NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F
                DECFSZ          LOOPCOUNT, F
                GOTO            DLOOPSM3232

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                endm

UMUL3232L        macro

;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26+2+7*28+27 = 842 clks

;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+1+7*6+5+6 = 197 clks

;       PM: 38+24+2+25+2+26+2+27+9 = 155            DM: 17

                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM3232A
                RRF             BARGB3, F
                BTFSC           _C
                GOTO            ALUM3232NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3232A

                MOVWF           LOOPCOUNT
```

```
LOOPUM3232B
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            BLUM3232NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3232B

                MOVWF           LOOPCOUNT

LOOPUM3232C
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            CLUM3232NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3232C

                MOVWF           LOOPCOUNT

LOOPUM3232D
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            DLUM3232NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3232D

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3232NAP     BCF             _C
                GOTO            ALUM3232NA

BLUM3232NAP     BCF             _C
                GOTO            BLUM3232NA

CLUM3232NAP     BCF             _C
                GOTO            CLUM3232NA

DLUM3232NAP     BCF             _C
                GOTO            DLUM3232NA

ALOOPUM3232
                RRF             BARGB3, F
                BTFSS           _C
                GOTO            ALUM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM3232NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
```

```
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM3232

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM3232
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            BLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM3232NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM3232

                MOVLW           0x08
                MOVWF           LOOPCOUNT

CLOOPUM3232
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            CLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLUM3232NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
```

```
                RRF             AARGB6, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPUM3232

                MOVLW           0x08
                MOVWF           LOOPCOUNT

DLOOPUM3232
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            DLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

DLUM3232NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F
                DECFSZ          LOOPCOUNT, F
                GOTO            DLOOPUM3232

                endm



UMUL3131L        macro

;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26+2+6*28+27+8 = 822 clks

;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+2+6*6+5+6 = 192 clks

;       PM: 39+24+2+25+2+26+2+27+8 = 155            DM: 17

                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM3131A
                RRF             BARGB3, F
                BTFSC           _C
                GOTO            ALUM3131NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3131A

                MOVWF           LOOPCOUNT

LOOPUM3131B
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            BLUM3131NAP
```

```
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3131B

                MOVWF           LOOPCOUNT

LOOPUM3131C
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            CLUM3131NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3131C

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPUM3131D
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            DLUM3131NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3131D

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3131NAP     BCF             _C
                GOTO            ALUM3131NA

BLUM3131NAP     BCF             _C
                GOTO            BLUM3131NA

CLUM3131NAP     BCF             _C
                GOTO            CLUM3131NA

DLUM3131NAP     BCF             _C
                GOTO            DLUM3131NA

ALOOPUM3131
                RRF             BARGB3, F
                BTFSS           _C
                GOTO            ALUM3131NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM3131NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
```

```
                GOTO            ALOOPUM3131

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM3131
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            BLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM3131NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM3131

                MOVLW           0x08
                MOVWF           LOOPCOUNT

CLOOPUM3131
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            CLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLUM3131NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPUM3131
```

```
                MOVLW           0x07
                MOVWF           LOOPCOUNT

DLOOPUM3131
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            DLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

DLUM3131NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F
                DECFSZ          LOOPCOUNT, F
                GOTO            DLOOPUM3131

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                endm


SMUL3232        macro

;       Max Timing:     9+7*22+8*23+8*24+7*25+9 = 723 clks

;       Min Timing:     62+6 = 68 clks

;       PM: 68+6+7*22+8*23+8*24+7*25+9 = 788            DM: 16


                variable i = 0

                while i < 8

                BTFSC           BARGB3,i
                GOTO            SM3232NA#v(i)

                variable i = i + 1
```

```
                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB2,i-8
                GOTO            SM3232NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            SM3232NA#v(i)

                variable i = i + 1

                endw

                variable i = 24

                while i < 31

                BTFSC           BARGB0,i-24
                GOTO            SM3232NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

SM3232NA0       RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB3,i
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM3232NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB2,i-8
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM3232NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 24

                BTFSS           BARGB1,i-16
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM3232NA#v(i)   RLF             SIGN,W
```

```
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F

                variable i = i + 1

                endw

                variable i = 24

                while   i < 31

                BTFSS           BARGB0,i-24
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM3232NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                variable i = i + 1

                endw

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                endm


UMUL3232        macro

;       Max Timing:     9+8*21+8*22+8*23+8*24 = 729 clks

;       Min Timing:     63+6 = 69 clks

;       PM: 69+6+8*21+8*22+8*23+8*24 = 795            DM: 16
```

```
                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB3,i
                GOTO            UM3232NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB2,i-8
                GOTO            UM3232NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            UM3232NA#v(i)

                variable i = i + 1

                endw

                variable i = 24

                while i < 32

                BTFSC           BARGB0,i-24
                GOTO            UM3232NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3232NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB3,i
                GOTO            UM3232NA#v(i)
```

```
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3232NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB2,i-8
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3232NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 24

                BTFSS           BARGB1,i-16
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
```

```
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3232NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F

                variable i = i + 1

                endw


                variable i = 24

                while   i < 32

                BTFSS           BARGB0,i-24
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3232NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                variable i = i + 1

                endw


                endm


UMUL3131        macro

;       Max Timing:     9+7*21+8*22+8*23+7*24+9 = 693 clks

;       Min Timing:     62+6 = 68 clks

;       PM: 68+6+7*22+8*23+8*24+7*25+9 = 788            DM: 16
```

```
                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB3,i
                GOTO            UM3131NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB2,i-8
                GOTO            UM3131NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            UM3131NA#v(i)

                variable i = i + 1

                endw


                variable i = 24

                while i < 31

                BTFSC           BARGB0,i-24
                GOTO            UM3131NA#v(i)

                variable i = i + 1

                endw


                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3131NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB3,i
```

```
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3131NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB2,i-8
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3131NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 24

                BTFSS           BARGB1,i-16
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3131NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F

                variable i = i + 1

                endw

                variable i = 24

                while   i < 31

                BTFSS           BARGB0,i-24
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3131NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                RRF             AARGB6, F
                RRF             AARGB7, F

                endm
```

```
;*****************************************************************************************
;*****************************************************************************************

;       32x32 Bit Signed Fixed Point Multiply 32x32 -> 64

;       Input:  32 bit signed fixed point multiplicand in AARGB0
;               32 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM3232S

;       Output: 64 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     15+851+2 = 868 clks               B > 0
;                       36+851+2 = 889 clks               B < 0

;       Min Timing:     15+192 = 207 clks

;       PM: 36+152+1 = 189              DM: 17


FXM3232S        CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB6
                CLRF            AARGB7
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M3232SOK

                COMF            BARGB3, F
                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB3, F
                BTFSC           _Z
                INCF            BARGB2, F
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F
```

```
                BTFSC           BARGB0,MSB
                GOTO            M3232SX

M3232SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                SMUL3232L

                RETLW           0x00

M3232SX         CLRF            AARGB4
                CLRF            AARGB5
                CLRF            AARGB6
                CLRF            AARGB7
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       32x32 Bit Unsigned Fixed Point Multiply 32x32 -> 64

;       Input:  32 bit unsigned fixed point multiplicand in AARGB0
;               32 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM3232U

;       Output: 64 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     12+842+2 = 856 clks

;       Min Timing:     12+197 = 209 clks

;       PM: 12+155+1 = 168              DM: 17

FXM3232U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB7
                CLRF            AARGB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3232L

                RETLW           0x00
```

```
;*****************************************************************************************
;*****************************************************************************************

;       31x31 Bit Unsigned Fixed Point Divide 31x31 -> 62

;       Input:  31 bit unsigned fixed point multiplicand in AARGB0
;               31 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM3131U

;       Output: 62 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     12+822+2 = 836 clks

;       Min Timing:     12+192 = 204 clks

;       PM: 12+155+1 = 168              DM: 17

FXM3131U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB7
                CLRF            AARGB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3131L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.2 32x24 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm24.a16 2.4 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.4 $

;       32x24 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM3224S     652        32x24 -> 56 bit signed fixed point multiply
;
;       FXM3224U     630        32x24 -> 56 bit unsigned fixed point multiply
;
;       FXM3123U     610        31x23 -> 54 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 80-97 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Multiplication Macros

SMUL3224L        macro

;       Max Timing:     2+13+6*26+25+2+7*27+26+2+6*28+27+8 = 618 clks

;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+6 = 146 clks

;       PM: 25+25+2+26+2+27+8 = 115            DM: 15


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM3224A
                RRF             BARGB2,F
                BTFSC           _C
                GOTO            ALSM3224NA
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPSM3224A

                MOVWF           LOOPCOUNT

LOOPSM3224B
                RRF             BARGB1,F
                BTFSC           _C
                GOTO            BLSM3224NA
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPSM3224B

                MOVLW           0x7
                MOVWF           LOOPCOUNT
```

```
LOOPSM3224C
                RRF             BARGB0,F
                BTFSC           _C
                GOTO            CLSM3224NA
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPSM3224C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALOOPSM3224
                RRF             BARGB2,F
                BTFSS           _C
                GOTO            ALSM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

ALSM3224NA      RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                DECFSZ          LOOPCOUNT,F
                GOTO            ALOOPSM3224

                MOVLW           0x8
                MOVWF           LOOPCOUNT

BLOOPSM3224
                RRF             BARGB1,F
                BTFSS           _C
                GOTO            BLSM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

BLSM3224NA      RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
```

```
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                DECFSZ          LOOPCOUNT,F
                GOTO            BLOOPSM3224

                MOVLW           0x7
                MOVWF           LOOPCOUNT

CLOOPSM3224
                RRF             BARGB0,F
                BTFSS           _C
                GOTO            CLSM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

CLSM3224NA      RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F
                DECFSZ          LOOPCOUNT,F
                GOTO            CLOOPSM3224

                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                endm

UMUL3224L        macro

;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26 = 617 clks

;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+6 = 151 clks

;       PM: 31+24+2+25+2+26+2+27 = 139            DM: 15

                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM3224A
                RRF             BARGB2,F
                BTFSC           _C
                GOTO            ALUM3224NAP
```

```
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3224A

                MOVWF           LOOPCOUNT

LOOPUM3224B
                RRF             BARGB1,F
                BTFSC           _C
                GOTO            BLUM3224NAP
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3224B

                MOVWF           LOOPCOUNT

LOOPUM3224C
                RRF             BARGB0,F
                BTFSC           _C
                GOTO            CLUM3224NAP
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3224C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3224NAP     BCF             _C
                GOTO            ALUM3224NA

BLUM3224NAP     BCF             _C
                GOTO            BLUM3224NA

CLUM3224NAP     BCF             _C
                GOTO            CLUM3224NA

ALOOPUM3224
                RRF             BARGB2,F
                BTFSS           _C
                GOTO            ALUM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

ALUM3224NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                DECFSZ          LOOPCOUNT,F
                GOTO            ALOOPUM3224

                MOVLW           0x08
                MOVWF           LOOPCOUNT
```

```
BLOOPUM3224
                RRF             BARGB1,F
                BTFSS           _C
                GOTO            BLUM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

BLUM3224NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                DECFSZ          LOOPCOUNT,F
                GOTO            BLOOPUM3224

                MOVLW           0x08
                MOVWF           LOOPCOUNT

CLOOPUM3224
                RRF             BARGB0,F
                BTFSS           _C
                GOTO            CLUM3224NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

CLUM3224NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F
                DECFSZ          LOOPCOUNT,F
                GOTO            CLOOPUM3224

                endm
```

```
UMUL3123L       macro

;       Max Timing:     2+15+6*25+24+2+7*26+25+2+6*27+26+7 = 597 clks

;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+6 = 146 clks

;       PM: 31+24+2+25+2+26+7 = 117            DM: 15

                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM3123A
                RRF             BARGB2,F
                BTFSC           _C
                GOTO            ALUM3123NAP
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3123A

                MOVWF           LOOPCOUNT

LOOPUM3123B
                RRF             BARGB1,F
                BTFSC           _C
                GOTO            BLUM3123NAP
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3123B

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPUM3123C
                RRF             BARGB0,F
                BTFSC           _C
                GOTO            CLUM3123NAP
                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPUM3123C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3123NAP     BCF             _C
                GOTO            ALUM3123NA

BLUM3123NAP     BCF             _C
                GOTO            BLUM3123NA

CLUM3123NAP     BCF             _C
                GOTO            CLUM3123NA

ALOOPUM3123
                RRF             BARGB2,F
                BTFSS           _C
                GOTO            ALUM3123NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
```

```
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

ALUM3123NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                DECFSZ          LOOPCOUNT,F
                GOTO            ALOOPUM3123

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM3123
                RRF             BARGB1,F
                BTFSS           _C
                GOTO            BLUM3123NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

BLUM3123NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                DECFSZ          LOOPCOUNT,F
                GOTO            BLOOPUM3123

                MOVLW           0x07
                MOVWF           LOOPCOUNT

CLOOPUM3123
                RRF             BARGB0,F
                BTFSS           _C
                GOTO            CLUM3123NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F

CLUM3123NA
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F
                DECFSZ          LOOPCOUNT,F
                GOTO            CLOOPUM3123

                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                endm


SMUL3224        macro

;       Max Timing:     9+7*22+8*23+7*24+8 = 523 clks

;       Min Timing:     40+6 = 46 clks

;       PM: 46+6+7*22+8*23+7*24+8 = 566           DM: 14


                variable i = 0

                while i < 8

                BTFSC           BARGB2,i
                GOTO            SM3224NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            SM3224NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            SM3224NA#v(i)

                variable i = i + 1
```

```
                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

SM3224NA0       RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
SM3224NA#v(i)   RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
```

```
SM3224NA#v(i)   RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 23

                BTFSS           BARGB0,i-16
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
SM3224NA#v(i)   RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                variable i = i + 1

                endw

                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                endm


UMUL3224        macro

;       Max Timing:     9+8*21+8*22+8*23 = 537 clks

;       Min Timing:     41+6 = 47 clks

;       PM: 47+6+8*21+8*22+8*23 = 581           DM: 14
```

```
                variable i = 0

                BCF             _C        ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM3224NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM3224NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 24

                BTFSC           BARGB0,i-16
                GOTO            UM3224NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3224NA0       RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
```

```
                ADDWF           AARGB0,F
UM3224NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
UM3224NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 24

                BTFSS           BARGB0,i-16
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
UM3224NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
```

```
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                variable i = i + 1

                endw

                endm


UMUL3123        macro

;       Max Timing:     9+7*21+8*22+7*23+7 = 500 clks

;       Min Timing:     41+6 = 47 clks

;       PM: 47+5+7*22+8*23+7*24+7 = 565            DM: 14


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM3123NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM3123NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            UM3123NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3123NA0       RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
```

```
                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
UM3123NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
UM3123NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 23
```

```
                BTFSS           BARGB0,i-16
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3,F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2,F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1,F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0,F
UM3123NA#v(i)   RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                variable i = i + 1

                endw

                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F
                RRF             AARGB5,F
                RRF             AARGB6,F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Signed Fixed Point Multiply 32x24 -> 56

;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3

;               24 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2

;       Use:    CALL    FXM3224S

;       Output: 56 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     14+618+2 = 634 clks               B > 0
;                       32+618+2 = 652 clks               B < 0

;       Min Timing:     14+146 = 160 clks

;       PM: 36+115+1 = 152              DM: 15

FXM3224S        CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB6
                CLRF            SIGN
```

```
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M3224SOK

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M3224SX

M3224SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                SMUL3224L

                RETLW           0x00

M3224SX         CLRF            AARGB4
                CLRF            AARGB5
                CLRF            AARGB6
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
                RRF             AARGB4,F

                RETLW           0x00

;*****************************************************************************************
```

```
;*****************************************************************************************

;       32x24 Bit Unsigned Fixed Point Multiply 32x24 -> 56

;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3

;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2

;       Use:    CALL    FXM3224U

;       Output: 56 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     11+617+2 = 630 clks

;       Min Timing:     11+151 = 162 clks

;       PM: 11+139+1 = 151              DM: 15

FXM3224U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3224L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       31x23 Bit Unsigned Fixed Point Divide 31x23 -> 54

;       Input:  31 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3

;               23 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2

;       Use:    CALL    FXM3123U

;       Output: 54 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     11+597+2 = 610 clks

;       Min Timing:     11+146 = 157 clks

;       PM: 11+117+1 = 129              DM: 15

FXM3123U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            AARGB6
```

```
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3123L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.3 32x16 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm26.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.3 $

;       32x16 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM3216S     423        32x16 -> 48 bit signed fixed point multiply
;
;       FXM3216U     412        32x16 -> 48 bit unsigned fixed point multiply
;
;       FXM3115U     392        31x15 -> 46 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 65-88 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Multiplication Macros

SMUL3216L        macro

;       Max Timing:     2+13+6*26+25+2+6*27+26+7 = 393 clks

;       Min Timing:     2+7*6+5+2+6*6+5+6 = 98 clks

;       PM: 19+60 = 79            DM: 11


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM3216A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALSM3216NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3216A

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPSM3216B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLSM3216NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM3216B

                CLRF            AARGB0
                CLRF            AARGB1
```

```
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALOOPSM3216
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALSM3216NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALSM3216NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPSM3216

                MOVLW           0x7
                MOVWF           LOOPCOUNT

BLOOPSM3216
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLSM3216NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLSM3216NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPSM3216

                RLF             TEMPB0,W
                RRF             AARGB0, F
```

```
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm

UMUL3216L        macro

;       Max Timing:     2+15+6*25+24+2+7*26+25 = 400 clks

;       Min Timing:     2+7*6+5+1+7*6+5+6 = 103 clks

;       PM: 73            DM: 11

                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM3216A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM3216NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3216A

                MOVWF           LOOPCOUNT

LOOPUM3216B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM3216NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3216B

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

BLUM3216NAP
                BCF             _C
                GOTO            BLUM3216NA

ALUM3216NAP
                BCF             _C
                GOTO            ALUM3216NA

ALOOPUM3216
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM3216NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
```

```
                ADDWF           AARGB0, F

ALUM3216NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM3216

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM3216
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLUM3216NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM3216NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM3216

                endm



UMUL3115L        macro

;       Max Timing:     2+15+6*25+24+2+6*26+25+6 = 380 clks

;       Min Timing:     2+7*6+5+2+6*6+5+6 = 96 clks

;       PM: 80            DM: 11

                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM3115A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM3115NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3115A

                MOVLW           0x7
```

```
                MOVWF           LOOPCOUNT

LOOPUM3115B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM3115NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM3115B

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

BLUM3115NAP
                BCF             _C
                GOTO            BLUM3115NA

ALUM3115NAP
                BCF             _C
                GOTO            ALUM3115NA

ALOOPUM3115
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM3115NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM3115NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM3115

                MOVLW           0x07
                MOVWF           LOOPCOUNT

BLOOPUM3115
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLUM3115NA
                MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM3115NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM3115

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm


SMUL3216        macro

;       Max Timing:     5+8+7*20+7*21+5 = 305 clks

;       Min Timing:     5+24+21+7 = 57 clks

;       PM: 5+24+21+6+5+7*20+7*21+5 = 353           DM: 10


                variable i = 0

                BTFSC           SIGN,MSB
                COMF            AARGB4, F
                MOVF            AARGB4,W
                MOVWF           AARGB5
                RLF             SIGN,W

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM3216NA#v(i)
                BCF             AARGB4,7-i

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM3216NA#v(i)
                BCF             AARGB5,15-i

                variable i = i + 1

                endw
```

```
                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                CLRF            AARGB5
                RETURN

SM3216NA0
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            SM3216NA#v(i)
SM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM3216NA#v(i)
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            SM3216NA#v(i)
SM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
```

```
SM3216NA#v(i)
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm


UMUL3216        macro

;       Max Timing:     1+8+7*21+8*22 = 332 clks

;       Min Timing:     1+2*8+2*8+6 = 39 clks

;       PM: 1+2*8+2*8+6+7*21+8*22 = 362            DM: 10


                variable i = 0

                BCF       _      C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM3216NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM3216NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3216NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
```

```
                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM3216NA#v(i)
UM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3216NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB0,i-8
                GOTO            UM3216NA#v(i)
UM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3216NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                endm
```

```
UMUL3115        macro

;       Max Timing:     9+7*21+7*22+6 = 316 clks

;       Min Timing:     1+30+6 = 37 clks

;       PM: 1+30+10+7*21+7*22+6 = 348            DM: 10


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM3115NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM3115NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM3115NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM3115NA#v(i)
UM3115A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3115NA#v(i)   RRF             AARGB0, F
```

```
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            UM3115NA#v(i)
UM3115A#v(i)    MOVF            TEMPB3,W
                ADDWF           AARGB3, F
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM3115NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Signed Fixed Point Multiply 32x16 -> 32

;       Input:  16 bit signed fixed point multiplicand in AARGB0
;               16 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM3216S

;       Output: 32 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     13+393+2 = 408 clks               B > 0
;                       28+393+2 = 423 clks               B < 0
```

```
;       Min Timing:     13+98 = 111 clks

;       PM: 18+79+1 = 98              DM: 9


FXM3216S        CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M3216SOK

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M3216SX

M3216SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                SMUL3216L

                RETLW           0x00


M3216SX         CLRF            AARGB4
                CLRF            AARGB5
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F
```

```
                RRF             AARGB4,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Unsigned Fixed Point Multiply 32x16 -> 32

;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;               16 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM3216U

;       Output: 32 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     10+400+2 = 412 clks

;       Min Timing:     10+104 = 114 clks

;       PM: 10+73+1 = 84              DM: 9

FXM3216U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3216L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       31x15 Bit Unsigned Fixed Point Divide 31x15 -> 30

;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;               15 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM3115U

;       Output: 30 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     10+380+2 = 392 clks

;       Min Timing:     10+96 = 106 clks

;       PM: 10+80+1 = 91              DM: 9

FXM3115U
                CLRF            AARGB4          ; clear partial product
                CLRF            AARGB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
```

```
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3

                UMUL3115L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.4 24x24 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm44.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.3 $

;       24x24 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM2424S     535        24x24 -> 48 bit signed fixed point multiply
;
;       FXM2424U     512        24x24 -> 48 bit unsigned fixed point multiply
;
;       FXM2323U     497        23x23 -> 46 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 61-95 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       24x24 Bit Multiplication Macros

SMUL2424L        macro

;       Max Timing:     2+12+6*21+20+2+7*22+21+2+6*23+22+7 = 506 clks

;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+5 = 145 clks

;       PM: 24+20+2+21+2+22+7 = 98           DM: 13


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM2424A
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            ALSM2424NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM2424A

                MOVWF           LOOPCOUNT

LOOPSM2424B
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            BLSM2424NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM2424B

                MOVLW           0x7
                MOVWF           LOOPCOUNT
```

```
LOOPSM2424C
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            CLSM2424NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM2424C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00

ALOOPSM2424
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            ALSM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALSM2424NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPSM2424

                MOVLW           0x8
                MOVWF           LOOPCOUNT

BLOOPSM2424
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            BLSM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLSM2424NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPSM2424

                MOVLW           0x7
                MOVWF           LOOPCOUNT
```

```
CLOOPSM2424
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            CLSM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLSM2424NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPSM2424

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm

UMUL2424L        macro

;       Max Timing:     2+14+6*20+19+2+7*21+20+2+7*22+21 = 501 clks

;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+5 = 150 clks

;       PM: 23+20+2+21+2+22 = 88            DM: 13


                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM2424A
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            ALUM2424NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2424A

                MOVWF           LOOPCOUNT

LOOPUM2424B
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            BLUM2424NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2424B

                MOVWF           LOOPCOUNT

LOOPUM2424C
```

```
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            CLUM2424NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2424C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00

CLUM2424NAP
                BCF             _C
                GOTO            CLUM2424NA

BLUM2424NAP
                BCF             _C
                GOTO            BLUM2424NA

ALUM2424NAP
                BCF             _C
                GOTO            ALUM2424NA

ALOOPUM2424
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            ALUM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM2424NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM2424

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM2424
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            BLUM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM2424NA
                RRF             AARGB0, F
```

```
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM2424

                MOVLW           0x08
                MOVWF           LOOPCOUNT

CLOOPUM2424
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            CLUM2424NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLUM2424NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPUM2424

                endm


UMUL2323L        macro

;       Max Timing:     2+15+6*20+19+2+7*21+20+2+6*22+21+6 = 486 clks

;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+5 = 145 clks

;       PM: 24+20+2+21+2+22+6 = 97           DM: 13


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM2323A
                RRF             BARGB2, F
                BTFSC           _C
                GOTO            ALUM2323NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2323A

                MOVWF           LOOPCOUNT

LOOPUM2323B
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            BLUM2323NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2323B
```

```
                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPUM2323C
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            CLUM2323NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2323C

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00

CLUM2323NAP
                BCF             _C
                GOTO            CLUM2323NA

BLUM2323NAP
                BCF             _C
                GOTO            BLUM2323NA

ALUM2323NAP
                BCF             _C
                GOTO            ALUM2323NA

ALOOPUM2323
                RRF             BARGB2, F
                BTFSS           _C
                GOTO            ALUM2323NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM2323NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM2323

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM2323
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            BLUM2323NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM2323NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM2323

                MOVLW           0x07
                MOVWF           LOOPCOUNT

CLOOPUM2323
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            CLUM2323NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

CLUM2323NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F
                DECFSZ          LOOPCOUNT, F
                GOTO            CLOOPUM2323

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm


SMUL2424        macro

;       Max Timing:     8+7*17+8*18+7*19+7 = 411 clks

;       Min Timing:     46+5 = 51 clks

;       PM: 51+4+7*17+8*18+7*19+7 = 466           DM: 12


                variable i = 0

                while i < 8

                BTFSC           BARGB2,i
                GOTO            SM2424NA#v(i)
```

```
                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            SM2424NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            SM2424NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

SM2424NA0       RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM2424NA#v(i)   RLF             SIGN,W, F
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                variable i = 8
```

```
                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM2424NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 23

                BTFSS           BARGB0,i-16
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM2424NA#v(i)   RLF             TEMPB0,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                RLF             TEMPB0,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm


UMUL2424        macro
```

```
;       Max Timing:     8+8*17+8*18+8*19 = 440 clks

;       Min Timing:     49+5 = 54 clks

;       PM: 54+4+8*17+8*18+8*19 = 490            DM: 12


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM2424NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM2424NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 24

                BTFSC           BARGB0,i-16
                GOTO            UM2424NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

UM2424NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2424NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2424NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 24

                BTFSS           BARGB0,i-16
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2424NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1

                endw

                endm
```

```
UMUL2323        macro

;       Max Timing:     8+7*17+8*18+7*19+7 = 411 clks

;       Min Timing:     46+5 = 51 clks

;       PM: 51+4+7*17+8*18+7*19+7 = 466            DM: 12


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM2323NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM2323NA#v(i)

                variable i = i + 1

                endw

                variable i = 16

                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            UM2323NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETURN

UM2323NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB2,i
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2323NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB1,i-8
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2323NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                variable i = 16

                while   i < 23

                BTFSS           BARGB0,i-16
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2323NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                variable i = i + 1
```

```
                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                RRF             AARGB5, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       24x24 Bit Signed Fixed Point Multiply 24x24 -> 48

;       Input:  24 bit signed fixed point multiplicand in AARGB0
;               24 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM2424S

;       Output: 48 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     12+506+2 = 520 clks               B > 0
;                       27+506+2 = 535 clks               B < 0

;       Min Timing:     12+145 = 157 clks

;       PM: 27+98+1 = 126             DM: 13

FXM2424S        CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                CLRF            AARGB5
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M2424SOK

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
```

```
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M2424SX

M2424SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                SMUL2424L

                RETLW           0x00

M2424SX         CLRF            AARGB3
                CLRF            AARGB4
                CLRF            AARGB5
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************

;       24x24 Bit Unsigned Fixed Point Multiply 24x24 -> 48

;       Input:  24 bit unsigned fixed point multiplicand in AARGB0
;               24 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM2424U

;       Output: 48 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     9+501+2 = 512 clks

;       Min Timing:     9+150 = 159 clks

;       PM: 9+88+1 = 98             DM: 13

FXM2424U
                CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                CLRF            AARGB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                UMUL2424L

                RETLW           0x00

;*****************************************************************************************
```

```
;*****************************************************************************************

;       23x23 Bit Unsigned Fixed Point Divide 23x23 -> 46

;       Input:  23 bit unsigned fixed point multiplicand in AARGB0
;               23 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM2323U

;       Output: 46 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     9+486+2 = 497 clks

;       Min Timing:     9+145 = 154 clks

;       PM: 9+97+1 = 107              DM: 13

FXM2323U
                CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                CLRF            AARGB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                UMUL2323L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.5 24x16 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm46.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.3 $

;       24x16 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM2416S     346        24x16 -> 40 bit signed fixed point multiply
;
;       FXM2416U     334        24x16 -> 40 bit unsigned fixed point multiply
;
;       FXM2315U     319        23x15 -> 38 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 36-62 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Multiplication Macros

SMUL2416L        macro

;       Max Timing:     2+12+6*21+20+2+6*22+21+6 = 321 clks

;       Min Timing:     2+7*6+5+2+6*6+5+5 = 97 clks

;       PM: 19+20+2+21+6 = 68           DM: 12


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM2416A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALSM2416NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM2416A

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPSM2416B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLSM2416NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM2416B

                CLRF            AARGB0
                CLRF            AARGB1
```

```
                CLRF            AARGB2
                RETLW           0x00

ALOOPSM2416
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALSM2416NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALSM2416NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPSM2416

                MOVLW           0x7
                MOVWF           LOOPCOUNT

BLOOPSM2416
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLSM2416NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLSM2416NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPSM2416


                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                endm

UMUL2416L        macro

;       Max Timing:     2+14+6*20+19+2+7*21+20 = 324 clks
```

```
;       Min Timing:     2+7*6+5+1+7*6+5+5 = 102 clks

;       PM: 18+20+2+21 = 61            DM: 12


                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM2416A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM2416NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2416A

                MOVWF           LOOPCOUNT

LOOPUM2416B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM2416NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2416B

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00

BLUM2416NAP
                BCF             _C
                GOTO            BLUM2416NA

ALUM2416NAP
                BCF             _C
                GOTO            ALUM2416NA

ALOOPUM2416
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM2416NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM2416NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM2416

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM2416
                RRF             BARGB0, F
```

```
                BTFSS           _C
                GOTO            BLUM2416NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM2416NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM2416

                endm



UMUL2315L       macro

;       Max Timing:     2+15+6*20+19+2+6*21+20+5 = 309 clks

;       Min Timing:     2+7*6+5+1+6*6+5+5 = 96 clks

;       PM: 19+20+2+21+5 = 67           DM: 12


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM2315A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM2315NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2315A

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPUM2315B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM2315NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM2315B

                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00

BLUM2315NAP
                BCF             _C
                GOTO            BLUM2315NA

ALUM2315NAP
                BCF             _C
```

```
                GOTO            ALUM2315NA

ALOOPUM2315
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM2315NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM2315NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM2315

                MOVLW           0x07
                MOVWF           LOOPCOUNT

BLOOPUM2315
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLUM2315NA
                MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM2315NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM2315

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                endm


SMUL2416        macro

;       Max Timing:     8+7*17+7*18+6 = 259 clks
```

```
;       Min Timing:     30+5 = 35 clks

;       PM: 30+4+7*17+7*18+6 = 285            DM: 11


                variable i = 0

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM2416NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM2416NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

SM2416NA0       RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            SM2416NA#v(i)
SM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM2416NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                variable i = 8
```

```
                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            SM2416NA#v(i)
SM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM2416NA#v(i)   RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                endm


UMUL2416        macro

;       Max Timing:     8+8*17+8*18 = 288 clks

;       Min Timing:     33+5 = 38 clks

;       PM: 37+4+8*17+8*18 = 321            DM: 11


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM2416NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM2416NA#v(i)

                variable i = i + 1
```

```
                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

UM2416NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM2416NA#v(i)
UM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2416NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB0,i-8
                GOTO            UM2416NA#v(i)
UM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2416NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                endm
```

```
UMUL2315        macro

;       Max Timing:     8+7*17+7*18+6 = 259 clks

;       Min Timing:     31+5 = 36 clks

;       PM: 35+4+7*17+7*18+6 = 290            DM: 11


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM2315NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM2315NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

UM2315NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM2315NA#v(i)
UM2315A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2315NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1
```

```
                endw

                variable i = 8

                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            UM2315NA#v(i)
UM2315A#v(i)    MOVF            TEMPB2,W
                ADDWF           AARGB2, F
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM2315NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                RRF             AARGB4, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Signed Fixed Point Multiply 24x16 -> 40

;       Input:  24 bit signed fixed point multiplicand in AARGB0
;               16 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM2416S

;       Output: 40 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     11+321+2 = 334 clks               B > 0
;                       23+321+2 = 346 clks               B < 0

;       Min Timing:     11+97 = 108 clks

;       PM: 23+68+1 = 92             DM: 12

FXM2416S        CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00
```

```
                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F


                BTFSS           BARGB0,MSB
                GOTO            M2416SOK

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M2416SX

M2416SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                SMUL2416L

                RETLW           0x00

M2416SX         CLRF            AARGB3
                CLRF            AARGB4
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F
                RRF             AARGB3,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Unsigned Fixed Point Multiply 24x16 -> 40

;       Input:  24 bit unsigned fixed point multiplicand in AARGB0
;               16 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM2416U

;       Output: 40 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     8+324+2 = 334 clks

;       Min Timing:     8+102 = 110 clks
```

```
;       PM: 8+61+1 = 70             DM: 12

FXM2416U
                CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                UMUL2416L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       23x15 Bit Unsigned Fixed Point Divide 23x15 -> 38

;       Input:  23 bit unsigned fixed point multiplicand in AARGB0
;               15 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM2315U

;       Output: 38 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     8+309+2 = 319 clks

;       Min Timing:     8+96 = 104 clks

;       PM: 8+67+1 = 76             DM: 12

FXM2315U
                CLRF            AARGB3          ; clear partial product
                CLRF            AARGB4
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2

                UMUL2315L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.6 16x16 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;        RCS Header $Id: fxm66.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;        $Revision: 2.3 $

;       16x16 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM1616S     269        16x16 -> 32 bit signed fixed point multiply
;
;       FXM1616U     256        16x16 -> 32 bit unsigned fixed point multiply
;
;       FXM1515U     244        15x15 -> 30 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 64-73 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Multiplication Macros

SMUL1616L        macro

;       Max Timing:     2+11+6*16+15+2+6*17+16+5 = 249 clks

;       Min Timing:     2+7*6+5+2+6*6+5+4 = 96 clks

;       PM: 55            DM: 9


                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPSM1616A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALSM1616NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM1616A

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPSM1616B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLSM1616NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM1616B

                CLRF            AARGB0
                CLRF            AARGB1
```

```
                RETLW           0x00

ALOOPSM1616
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALSM1616NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALSM1616NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPSM1616

                MOVLW           0x7
                MOVWF           LOOPCOUNT

BLOOPSM1616
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLSM1616NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLSM1616NA      RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPSM1616

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                endm

UMUL1616L        macro

;       Max Timing:     2+13+6*15+14+2+7*16+15 = 248 clks

;       Min Timing:     2+7*6+5+1+7*6+5+4 = 101 clks

;       PM: 51            DM: 9

                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM1616A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM1616NAP
                DECFSZ          LOOPCOUNT, F
```

```
                GOTO            LOOPUM1616A

                MOVWF           LOOPCOUNT

LOOPUM1616B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM1616NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1616B

                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00

BLUM1616NAP
                BCF             _C
                GOTO            BLUM1616NA

ALUM1616NAP
                BCF             _C
                GOTO            ALUM1616NA

ALOOPUM1616
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM1616NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM1616NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM1616

                MOVLW           0x08
                MOVWF           LOOPCOUNT

BLOOPUM1616
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLUM1616NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM1616NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM1616

                endm
```

```
UMUL1515L       macro

;       Max Timing:     2+13+6*15+14+2+6*16+15+4 = 236 clks

;       Min Timing:     2+7*6+5+2+6*6+5+4 = 97 clks

;       PM: 56            DM: 9

                MOVLW           0x8
                MOVWF           LOOPCOUNT

LOOPUM1515A
                RRF             BARGB1, F
                BTFSC           _C
                GOTO            ALUM1515NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1515A

                MOVLW           0x7
                MOVWF           LOOPCOUNT

LOOPUM1515B
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            BLUM1515NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1515B

                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00

BLUM1515NAP
                BCF             _C
                GOTO            BLUM1515NA

ALUM1515NAP
                BCF             _C
                GOTO            ALUM1515NA

ALOOPUM1515
                RRF             BARGB1, F
                BTFSS           _C
                GOTO            ALUM1515NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

ALUM1515NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            ALOOPUM1515

                MOVLW           0x07
                MOVWF           LOOPCOUNT

BLOOPUM1515
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            BLUM1515NA
```

```
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

BLUM1515NA
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F
                DECFSZ          LOOPCOUNT, F
                GOTO            BLOOPUM1515


                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                endm



SMUL1616        macro

;       Max Timing:     5+6+7*11+7*12+4 = 176 clks

;       Min Timing:     5+24+21+5 = 55 clks

;       PM: 5+3*8+3*7+6+7*11+7*12+4 = 221           DM: 8

                variable i = 0

                BTFSC           SIGN,MSB
                COMF            AARGB2, F
                MOVF            AARGB2,W
                MOVWF           AARGB3
                RLF             SIGN,W

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM1616NA#v(i)
                BCF             AARGB2,7-i

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM1616NA#v(i)
                BCF             AARGB3,15-i

                variable i =i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                RETURN
```

```
SM1616NA0
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            SM1616NA#v(i)
SM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM1616NA#v(i)
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            SM1616NA#v(i)
SM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM1616NA#v(i)
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                endm


UMUL1616        macro

;       Max Timing:     1+6+7*11+8*12 = 180 clks

;       Min Timing:     1+2*8+2*8+4 = 37 clks

;       PM: 1+2*8+2*8+4+7*11+8*12 = 210           DM: 8
```

```
                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM1616NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM1616NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                RETURN

UM1616NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM1616NA#v(i)
UM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1616NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = i + 1

                endw

                variable i = 8

                while   i < 16

                BTFSS           BARGB0,i-8
                GOTO            UM1616NA#v(i)
UM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1616NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
```

```
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                endm


UMUL1515        macro

;       Max Timing:     7+7*11+7*12+4 = 172 clks

;       Min Timing:     1+16+14+4 = 35 clks

;       PM: 1+2*8+2*7+6+7*11+7*12+4 = 202           DM: 8


                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM1515NA#v(i)

                variable i = i + 1

                endw

                variable i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM1515NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                RETURN

UM1515NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB1,i
                GOTO            UM1515NA#v(i)
UM1515A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1515NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
```

```
                variable i = i + 1

                endw

                variable i = 8

                while   i < 15

                BTFSS           BARGB0,i-8
                GOTO            UM1515NA#v(i)
UM1515A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1515NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                RRF             AARGB3, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Signed Fixed Point Multiply 16x16 -> 32

;       Input:  16 bit signed fixed point multiplicand in AARGB0
;               16 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM1616S

;       Output: 32 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     9+249+2 = 260 clks              B > 0
;                       18+249+2 = 269 clks             B < 0

;       Min Timing:     9+96 = 105 clks

;       PM: 18+55+1 = 74             DM: 9

FXM1616S        CLRF            AARGB2          ; clear partial product
                CLRF            AARGB3
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
```

```
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M1616SOK

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M1616SX

M1616SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1

                SMUL1616L

                RETLW           0x00


M1616SX         CLRF            AARGB2
                CLRF            AARGB3
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Unsigned Fixed Point Multiply 16x16 -> 32

;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;               16 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM1616U

;       Output: 32 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     6+248+2 = 256 clks

;       Min Timing:     6+101 = 107 clks

;       PM: 6+51+1 = 58             DM: 9

FXM1616U
                CLRF            AARGB2          ; clear partial product
                CLRF            AARGB3
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
```

```
                UMUL1616L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       15x15 Bit Unsigned Fixed Point Divide 15x15 -> 30

;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;               15 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM1515U

;       Output: 30 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     6+236+2 = 244 clks

;       Min Timing:     6+97 = 103 clks

;       PM: 6+56+1 = 63             DM: 9

FXM1515U
                CLRF            AARGB2          ; clear partial product
                CLRF            AARGB3
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1

                UMUL1515L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## D.7 16x8 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm68.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.3 $

;       16x8 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM1608S     128        16x08 -> 24 bit signed fixed point multiply
;
;       FXM1608U     126        16x08 -> 24 bit unsigned fixed point multiply
;
;       FXM1507U     114        15x07 -> 22 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 24-35 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       16x08 Bit Multiplication Macros

SMUL1608L        macro

;       Max Timing:     2+11+5*16+15+4 = 112 clks

;       Min Timing:     2+6*6+5+4 = 47 clks

;       PM: 29            DM: 7


                MOVLW           0x07
                MOVWF           LOOPCOUNT

LOOPSM1608A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LSM1608NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM1608A

                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00

LOOPSM1608
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            LSM1608NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F

LSM1608NA       RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM1608

                RLF             SIGN,W
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                endm

UMUL1608L        macro

;       Max Timing:     2+13+6*15+14 = 119 clks

;       Min Timing:     2+7*6+5+4 = 54 clks

;       PM: 26            DM: 7

                MOVLW           0x08
                MOVWF           LOOPCOUNT

LOOPUM1608A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LUM1608NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1608A

                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00

LUM1608NAP
                BCF             _C
                GOTO            LUM1608NA

LOOPUM1608
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            LUM1608NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
LUM1608NA       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1608

                endm


UMUL1507L        macro

;       Max Timing:     2+13+5*15+14+3 = 107 clks
```

```
;       Min Timing:     2+6*6+5+4 = 47 clks

;       PM: 29            DM: 7

                MOVLW           0x07
                MOVWF           LOOPCOUNT

LOOPUM1507A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LUM1507NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1507A

                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00

LUM1507NAP
                BCF             _C
                GOTO            LUM1507NA

LOOPUM1507
                RRF             BARGB0, F
                BTFSS           _C
                GOTO            LUM1507NA
                MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
LUM1507NA       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM1507

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                endm



SMUL1608        macro

;       Max Timing:     3+6+6*11+3 = 78 clks

;       Min Timing:     3+21+5 = 29 clks

;       PM: 3+3*7+7+6*11+3 = 100            DM: 6

                variable i =0

                BTFSC           SIGN,MSB
                COMF            AARGB2, F
                RLF             SIGN,W

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM1608NA#v(i)
                BCF             AARGB2,7-i
```

```
                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                CLRF            AARGB2
                RETURN

SM1608NA0
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 7

                BTFSS           BARGB0,i
                GOTO            SM1608NA#v(i)
SM1608A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
SM1608NA#v(i)
                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                endm


UMUL1608        macro

;       Max Timing:     1+6+7*11 = 84 clks

;       Min Timing:     1+2*8+4 = 21 clks

;       PM: 1+2*8+4+6*7 = 63            DM: 4

                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM1608NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0         ; if we get here, BARG = 0
                CLRF            AARGB1
```

```
                RETURN

UM1608NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 8

                BTFSS           BARGB0,i
                GOTO            UM1608NA#v(i)
UM1608A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1608NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = i + 1

                endw

                endm


UMUL1507        macro

;       Max Timing:     7+6*12+4 = 83 clks

;       Min Timing:     14+3 = 17 clks

;       PM: 2*7+7+6*12+4 = 97           DM: 6

                variable i = 0

                BCF             _C              ; clear carry for first right shift

                while i < 7

                BTFSC           BARGB0,i
                GOTO            UM1507NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                CLRF            AARGB1
                RETURN

UM1507NA0       RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = 1

                while   i < 7

                BTFSS           BARGB0,i
                GOTO            UM1507NA#v(i)
UM1507A#v(i)    MOVF            TEMPB1,W
                ADDWF           AARGB1, F
```

```
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           AARGB0, F
UM1507NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F
                RRF             AARGB2, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       16x8 Bit Signed Fixed Point Multiply 16x8 -> 24

;       Input:  16 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM1608S

;       Output: 24 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     8+112+2 = 122 clks              B > 0
;                       14+112+2 = 128 clks             B < 0

;       Min Timing:     8+47 = 55 clks

;       PM: 14+29+1 = 44             DM: 7

FXM1608S        CLRF            AARGB2          ; clear partial product
                CLRF            SIGN
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMPB0
                BTFSC           TEMPB0,MSB
                COMF            SIGN,F

                BTFSS           BARGB0,MSB
                GOTO            M1608SOK

                COMF            BARGB0,F         ; make multiplier BARG > 0
                INCF            BARGB0,F

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M1608SX
```

```
M1608SOK        MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1

                SMUL1608

                RETLW           0x00

M1608SX         CLRF            AARGB2
                RLF             SIGN,W
                RRF             AARGB0,F
                RRF             AARGB1,F
                RRF             AARGB2,F

                RETLW        0x00


;*****************************************************************************************
;*****************************************************************************************

;       16x8 Bit Unsigned Fixed Point Multiply 16x8 -> 24

;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM1608U

;       Output: 24 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     5+119+2 = 126 clks

;       Min Timing:     5+54 = 59 clks

;       PM: 5+26+1 = 31              DM: 7

FXM1608U        CLRF            AARGB2          ; clear partial product
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1

                UMUL1608L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       15x7 Bit Unsigned Fixed Point Divide 15x7 -> 22

;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;               7 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM0807U

;       Output: 22 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     5+107+2 = 114 clks

;       Min Timing:     5+47 = 52 clks
```

```
;       PM: 5+29+1 = 35              DM: 7

FXM1507U        CLRF            AARGB2          ; clear partial product
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1

                UMUL1507

                RETLW           0x00

;*************************************************************************************************
;*************************************************************************************************
```

## D.8 8x8 PIC16C5X/PIC16CXXX Fixed Point Multiply Routines

```
;       RCS Header $Id: fxm88.a16 2.3 1996/10/16 14:23:23 F.J.Testa Exp $

;       $Revision: 2.3 $

;       8x8 PIC16 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARGxBARG in AARG
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks     Function
;
;       FXM0808S     82         08x08 -> 16 bit signed fixed point multiply
;
;       FXM0808U     73         08x08 -> 16 bit unsigned fixed point multiply
;
;       FXM0707U     67         07x07 -> 14 bit unsigned fixed point multiply
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 29-35 clocks can be saved by using the unrolled macros.
;

;*************************************************************************************************
;*************************************************************************************************

;       08x08 Bit Multiplication Macros

SMUL0808L        macro

;       Max Timing:     3+10+5*9+8+3 = 69 clks

;       Min Timing:     3+6*6+5+3 = 47 clks

;       PM: 21            DM: 5

                MOVLW           0x07
                MOVWF           LOOPCOUNT

                MOVF            AARGB0,W

LOOPSM0808A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LSM0808NA
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPSM0808A

                CLRF            AARGB0
                RETLW           0x00

LOOPSM0808
                RRF             BARGB0, F
                BTFSC           _C
                ADDWF           AARGB0, F
LSM0808NA       RLF             SIGN, F
                RRF             AARGB0, F
                RRF             AARGB1, F
                DECFSZ          LOOPCOUNT, F
```

```
                GOTO            LOOPSM0808

                RLF             SIGN, F
                RRF             AARGB0, F
                RRF             AARGB1, F


                endm

UMUL0808L        macro

;       Max Timing:     3+12+6*8+7 = 70 clks

;       Min Timing:     3+7*6+5+3 = 53 clks

;       PM: 19            DM: 4

                MOVLW           0x08
                MOVWF           LOOPCOUNT
                MOVF            AARGB0,W

LOOPUM0808A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LUM0808NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM0808A

                CLRF            AARGB0
                RETLW           0x00

LUM0808NAP
                BCF             _C
                GOTO            LUM0808NA

LOOPUM0808
                RRF             BARGB0, F
                BTFSC           _C
                ADDWF           AARGB0, F
LUM0808NA       RRF             AARGB0, F
                RRF             AARGB1, F
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM0808

                endm


UMUL0707L        macro

;       Max Timing:     3+12+5*8+7+2 = 64 clks

;       Min Timing:     3+6*6+5+3 = 47 clks

;       PM: 21            DM: 4

                MOVLW           0x07
                MOVWF           LOOPCOUNT
                MOVF            AARGB0,W

LOOPUM0707A
                RRF             BARGB0, F
                BTFSC           _C
                GOTO            LUM0707NAP
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM0707A
```

```
                CLRF            AARGB0
                RETLW           0x00

LUM0707NAP
                BCF             _C
                GOTO            LUM0707NA

LOOPUM0707
                RRF             BARGB0, F
                BTFSC           _C
                ADDWF           AARGB0, F
LUM0707NA       RRF             AARGB0, F
                RRF             AARGB1, F
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPUM0707

                RRF             AARGB0, F
                RRF             AARGB1, F

                endm



SMUL0808        macro

;       Max Timing:     1+6+6*5+3 = 40 clks

;       Min Timing:     1+14+3 = 18 clks

;       PM: 1+2*7+5+6*5+3 = 53            DM: 5

                variable i = 0

                MOVF            AARGB0,W

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM0808NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                RETURN

SM0808NA0       RLF             SIGN
                RRF             AARGB0
                RRF             AARGB1

                variable i = 1

                while   i < 7

                BTFSC           BARGB0,i
                ADDWF           AARGB0
SM0808NA#v(i)   RLF             SIGN
                RRF             AARGB0
                RRF             AARGB1

                variable i = i + 1

                endw
```

```
                RLF             SIGN
                RRF             AARGB0
                RRF             AARGB1

                endm


UMUL0808        macro

;       Max Timing:     2+5+7*4 = 35 clks

;       Min Timing:     2+16+3 = 21 clks

;       PM: 2+2*8+4+7*4 = 50            DM: 3

                variable i = 0

                BCF             _C              ; clear carry for first right shift
                MOVF            AARGB0,W

                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM0808NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                RETURN

UM0808NA0       RRF             AARGB0, F
                RRF             AARGB1, F

                variable i = 1

                while   i < 8

                BTFSC           BARGB0,i
                ADDWF           AARGB0, F
UM0808NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F

                variable i = i + 1

                endw

                endm


UMUL0707        macro

;       Max Timing:     2+5+6*4+2 = 33 clks

;       Min Timing:     2+14+3 = 19 clks

;       PM: 2+2*7+4+6*4+2 = 46            DM: 3

                variable i = 0

                BCF             _C              ; clear carry for first right shift
                MOVF            AARGB0,W

                while i < 7
```

```
                BTFSC           BARGB0,i
                GOTO            UM0707NA#v(i)

                variable i = i + 1

                endw

                CLRF            AARGB0          ; if we get here, BARG = 0
                RETURN

UM0707NA0       RRF             AARGB0, F
                RRF             AARGB1, F

                variable i = 1

                while   i < 7

                BTFSC           BARGB0,i
                ADDWF           AARGB0, F
UM0707NA#v(i)   RRF             AARGB0, F
                RRF             AARGB1, F

                variable i = i + 1

                endw

                RRF             AARGB0, F
                RRF             AARGB1, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Signed Fixed Point Multiply 8x8 -> 16

;       Input:  8 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0

;       Use:    CALL    FXM0808S

;       Output: 16 bit signed fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     12+69+2 = 83 clks               B > 0
;                       17+69+2 = 88 clks               B < 0

;       Min Timing:     12+47 = 59 clks
;                        6 clks                         A = 0

;       PM: 17+21+1 = 39             DM: 5

FXM0808S        CLRF            AARGB1          ; clear partial product
                CLRF            SIGN
                MOVF            AARGB0,W
                BTFSC           _Z
                RETLW           0x00

                XORWF           BARGB0,W
                MOVWF           TEMPB3
                BTFSC           TEMPB3,MSB
                COMF            SIGN, F


                BTFSS           BARGB0,MSB
```

```
                GOTO            M0808SOK
                COMF            BARGB0, F       ; make multiplier BARG > 0
                INCF            BARGB0, F
                COMF            AARGB0, F
                INCF            AARGB0, F

                BTFSC           BARGB0,MSB
                GOTO            M0808SX

M0808SOK

                SMUL0808L

                RETLW           0x00

M0808SX         CLRF            AARGB1
                RLF             SIGN, W
                RRF             AARGB0, F
                RRF             AARGB1, F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Unsigned Fixed Point Multiply 8x8 -> 16

;       Input:  8 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM0808U

;       Output: 8 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     1+70+2 = 73 clks

;       Min Timing:     1+53 = 54 clks

;       PM: 1+19+1 = 21             DM: 4

FXM0808U        CLRF            AARGB1          ; clear partial product

                UMUL0808L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       7x7 Bit Unsigned Fixed Point Divide 7x7 -> 14

;       Input:  7 bit unsigned fixed point multiplicand in AARGB0
;               7 bit unsigned fixed point multiplier in BARGB0

;       Use:    CALL    FXM0707U

;       Output: 14 bit unsigned fixed point product in AARGB0

;       Result: AARG  <--  AARG x BARG

;       Max Timing:     1+64+2 = 67 clks

;       Min Timing:     1+47 = 48 clks
```

```
;       PM: 1+21+1 = 23              DM: 4

FXM0707U        CLRF            AARGB1          ; clear partial product

                UMUL0707L

                RETLW           0x00

;*************************************************************************************************
;*************************************************************************************************
```

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

# APPENDIX E: PIC16C5X/PIC16CXX DIVIDE ROUTINES

**Table of Contents for Appendix E**



## E.1 32/32 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd22.a16 2.4 1997/02/27 03:03:17 F.J.Testa Exp $

;       $Revision: 2.4 $

;       32/32 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD3232S    929       32 bit/32 bit -> 32.32 signed fixed point divide
;
;       FXD3232U    1031      32 bit/32 bit -> 32.32 unsigned fixed point divide
;
;       FXD3131U    869       31 bit/31 bit -> 31.31 unsigned fixed point divide
;
;*************************************************************************************************
;*************************************************************************************************

;       32/32 Bit Division Macros

SDIV3232L       macro

;       Max Timing:     17+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+16 = 863 clks

;       Min Timing:     17+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 819 clks

;       PM: 17+7*38+16 = 299                                    DM: 13

                MOVF            BARGB3,W
                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3232A      RLF             AARGB0,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            SADD22LA

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22LA

SADD22LA        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3232A

                RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            SADD22L8

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
```

```
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22L8

SADD22L8        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3232B      RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            SADD22LB

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22LB

SADD22LB        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
```

```
SOK22LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3232B

                RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            SADD22L16

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22L16

SADD22L16       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3232C      RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            SADD22LC

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
```

```
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22LC

SADD22LC        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3232C

                RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            SADD22L24

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22L24

SADD22L24       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22L24        RLF             AARGB3, F

                MOVLW           7
```

```
                MOVWF           LOOPCOUNT

LOOPS3232D      RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB3,LSB
                GOTO            SADD22LD

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK22LD

SADD22LD        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C

                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK22LD         RLF             AARGB3, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3232D

                BTFSC           AARGB3,LSB
                GOTO            SOK22L
                MOVF            BARGB3,W
                ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK22L

                endm
```

```
UDIV3232L       macro

;       Max Timing:     24+6*32+31+31+6*32+31+31+6*32+31+31+6*32+31+16 = 1025 clks

;       Min Timing:     24+6*31+30+30+6*31+30+30+6*31+30+30+6*31+30+3 = 981 clks

;       PM: 359                                 DM: 13

                CLRF            TEMP

                RLF             AARGB0,W
                RLF             REMB3, F
                MOVF            BARGB3,W
                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F

                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3232A      RLF             AARGB0,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            UADD22LA

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22LA

UADD22LA        ADDWF           REMB3, F
```

```
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK22LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3232A

                RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            UADD22L8

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22L8

UADD22L8        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
```

```
                MOVLW           1
                ADDWF           TEMP, F

U0K22L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3232B      RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            UADD22LB

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            U0K22LB

UADD22LB        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

U0K22LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3232B

                RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
```

```
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            UADD22L16

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22L16

UADD22L16       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK22L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3232C      RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            UADD22LC

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22LC

UADD22LC        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK22LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3232C

                RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            UADD22L24

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22L24

UADD22L24       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
```

```
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK22L24        RLF             AARGB3, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3232D      RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB3,W
                BTFSS           AARGB3,LSB
                GOTO            UADD22LD

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK22LD

UADD22LD        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK22LD         RLF             AARGB3, F
```

```
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3232D

                BTFSC           AARGB3,LSB
                GOTO            UOK22L
                MOVF            BARGB3,W
                ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK22L

                endm

UDIV3131L       macro

;       Max Timing:     17+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+16 = 863 clks

;       Min Timing:     17+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 819 clks

;       PM: 17+7*38+16 = 299                                    DM: 13

                MOVF            BARGB3,W
                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3131A      RLF             AARGB0,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            UADD11LA

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
```

```
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11LA

UADD11LA        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK11LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3131A

                RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB0,LSB
                GOTO            UADD11L8

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11L8

UADD11L8        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
```

```
U0K11L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3131B      RLF             AARGB1,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            UADD11LB

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            U0K11LB

UADD11LB        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

U0K11LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3131B

                RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB1,LSB
                GOTO            UADD11L16

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
```

```
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11L16

UADD11L16       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK11L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3131C      RLF             AARGB2,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            UADD11LC

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11LC

UADD11LC        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK11LC         RLF             AARGB2, F
```

```
                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3131C

                RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB2,LSB
                GOTO            UADD11L24

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11L24

UADD11L24       ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK11L24        RLF             AARGB3, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3131D      RLF             AARGB3,W
                RLF             REMB3, F
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB3,W
                BTFSS           AARGB3,LSB
                GOTO            UADD11LD

                SUBWF           REMB3, F
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
```

```
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK11LD

UADD11LD        ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C

                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK11LD         RLF             AARGB3, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3131D

                BTFSC           AARGB3,LSB
                GOTO            UOK11L
                MOVF            BARGB3,W
                ADDWF           REMB3, F
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK11L

                endm

;*****************************************************************************************
;*****************************************************************************************

;       32/32 Bit Signed Fixed Point Divide 32/32 -> 32.32

;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3

;       Use:    CALL    FXD3232S

;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point remainder in REMB0, REMB1, REMB2, REMB3

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     28+863+5  = 896 clks           A > 0, B > 0
;                       38+863+28 = 929 clks           A > 0, B < 0
;                       38+863+28 = 929 clks           A < 0, B > 0
;                       48+863+5  = 916 clks           A < 0, B < 0
;                                   12 clks            A = 0

;       Min Timing:     28+819+5  = 852 clks            A > 0, B > 0
```

```
;                       38+819+28 = 885 clks            A > 0, B < 0
;                       38+819+28 = 885 clks            A < 0, B > 0
;                       48+819+5  = 872 clks            A < 0, B < 0

;       PM: 48+299+27+67 = 441                              DM: 15

FXD3232S        CLRF            SIGN
                CLRF            REMB0                   ; clear partial remainder
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                  ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3232S

                COMF            BARGB3, F
                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB3, F
                BTFSC           _Z
                INCF            BARGB2, F
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

CA3232S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3232SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C3232SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C3232SX1

C3232S          SDIV3232L

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C3232SX4
```

```
C3232SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB3, F
                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB3, F
                BTFSC           _Z
                INCF            REMB2, F
                BTFSC           _Z
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00

C3232SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C3232SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C3232SX2
                MOVF            AARGB0,W
                MOVWF           REMB0                   ; quotient = 0, remainder = AARG
                MOVF            AARGB1,W
                MOVWF           REMB1
                MOVF            AARGB2,W
                MOVWF           REMB2
                MOVF            AARGB3,W
                MOVWF           REMB3
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                GOTO            C3232SOK
C3232SX2        CLRF            AARGB0                  ; quotient = 1, remainder = 0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                INCF            AARGB3,F
                RETLW           0x00

C3232SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3232S

C3232SX4        INCF            REMB3,F                 ; increment remainder and test for
                BTFSC           _Z                      ; overflow
                INCF            REMB2,F
                BTFSC           _Z
                INCF            REMB1,F
```

```
                BTFSC           _Z
                INCF            REMB0,F
                MOVF            BARGB3,W
                SUBWF           REMB3,W
                BTFSS           _Z
                GOTO            C3232SOK
                MOVF            BARGB2,W
                SUBWF           REMB2,W
                BTFSS           _Z
                GOTO            C3232SOK
                MOVF            BARGB1,W
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C3232SOK
                MOVF            BARGB0,W
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C3232SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                CLRF            REMB1                   ; remainder, increment quotient and
                CLRF            REMB2
                CLRF            REMB3
                INCF            AARGB3,F                ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB2,F
                BTFSC           _Z
                INCF            AARGB1,F
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C3232SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF

;*****************************************************************************************
;*****************************************************************************************

;       32/32 Bit Unsigned Fixed Point Divide 32/32 -> 32.32

;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3

;       Use:    CALL    FXD3232U

;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     4+1025+2 = 1031 clks

;       Max Timing:     4+981+2 = 987 clks

;       PM: 4+359+1 = 364               DM: 13

FXD3232U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3

                UDIV3232L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

```
;       31/31 Bit Unsigned Fixed Point Divide 31/31 -> 31.31

;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2, BARGB3

;       Use:    CALL    FXD3131U

;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     4+863+2 = 869 clks

;       Min Timing:     4+819+2 = 825 clks

;       PM: 4+299+1 = 304               DM: 13

FXD3131U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3

                UDIV3131L

                RETLW           0x00

;*************************************************************************************************
;*************************************************************************************************
```

## E.2 32/24 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd24.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;       $Revision: 2.3 $

;       32/24 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD3224S    759       32 bit/24 bit -> 32.24 signed fixed point divide
;
;       FXD3224U    867       32 bit/24 bit -> 32.24 unsigned fixed point divide
;
;       FXD3123U    705       31 bit/23 bit -> 31.23 unsigned fixed point divide
;

;*************************************************************************************************
;*************************************************************************************************

;       32/24 Bit Division Macros

SDIV3224L       macro

;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+12 = 700 clks

;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 660 clks

;       PM: 11+3*58+43 = 228                                    DM: 10

                MOVF            BARGB2,W
                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                RLF             AARGB0,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3224A      RLF             AARGB0,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            SADD24LA

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
```

```
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24LA

SADD24LA        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24LA         RLF             AARGB0,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPS3224A

                RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            SADD24L8

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24L8

SADD24L8        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24L8         RLF             AARGB1,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3224B      RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            SADD24LB

                SUBWF           REMB2,F
```

```
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24LB

SADD24LB        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24LB         RLF             AARGB1,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPS3224B

                RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            SADD24L16

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24L16

SADD24L16       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24L16        RLF             AARGB2,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3224C      RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            SADD24LC
```

```
                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24LC

SADD24LC        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24LC         RLF             AARGB2,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPS3224C

                RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            SADD24L24

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24L24

SADD24L24       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24L24        RLF             AARGB3,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3224D      RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
```

```
                BTFSS           AARGB3,LSB
                GOTO            SADD24LD

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            SOK24LD

SADD24LD        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

SOK24LD         RLF             AARGB3,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPS3224D

                BTFSC           AARGB3,LSB
                GOTO            SOK24L
                MOVF            BARGB2,W
                ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
SOK24L

                endm

UDIV3224L       macro

;       Max Timing:     20+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+12 = 862 clks

;       Min Timing:     20+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 822 clks

;       PM: 18+3*75+40+12 = 295                                 DM: 11

                CLRF            TEMP

                RLF             AARGB0,W
                RLF             REMB2,F
                MOVF            BARGB2,W
                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0,F

                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                RLF             AARGB0,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3224A      RLF             AARGB0,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD24LA

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24LA

UADD24LA        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F

UOK24LA         RLF             AARGB0,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3224A

                RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD24L8

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
```

```
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24L8

UADD24L8        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F

UOK24L8         RLF             AARGB1,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3224B      RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD24LB

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24LB

UADD24LB        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
```

```
                MOVLW           1
                ADDWF           TEMP,F

UOK24LB         RLF             AARGB1,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3224B

                RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD24L16

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24L16

UADD24L16       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F

UOK24L16        RLF             AARGB2,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3224C      RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            UADD24LC

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24LC

UADD24LC        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F

UOK24LC         RLF             AARGB2,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3224C

                RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            UADD24L24

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24L24

UADD24L24       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F
```

```
UOK24L24        RLF             AARGB3,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3224D      RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                RLF             TEMP,F
                MOVF            BARGB2,W
                BTFSS           AARGB3,LSB
                GOTO            UADD24LD

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP,F
                GOTO            UOK24LD

UADD24LD        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP,F

UOK24LD         RLF             AARGB3,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3224D

                BTFSC           AARGB3,LSB
                GOTO            UOK24L
                MOVF            BARGB2,W
                ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK24L

                endm

UDIV3123L       macro
```

```
;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+12 = 700 clks

;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 660 clks

;       PM: 11+3*58+43 = 228                                    DM: 10

                MOVF            BARGB2,W
                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                RLF             AARGB0,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3123A      RLF             AARGB0,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD13LA

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13LA

UADD13LA        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13LA         RLF             AARGB0,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3123A

                RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD13L8

                SUBWF           REMB2,F
                MOVF            BARGB1,W
```

```
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13L8

UADD13L8        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13L8         RLF             AARGB1,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3123B      RLF             AARGB1,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD13LB

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13LB

UADD13LB        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13LB         RLF             AARGB1,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3123B

                RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD13L16
```

```
                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13L16

UADD13L16       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13L16        RLF             AARGB2,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3123C      RLF             AARGB2,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            UADD13LC

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13LC

UADD13LC        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13LC         RLF             AARGB2,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3123C

                RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
```

```
                GOTO            UADD13L24

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13L24

UADD13L24       ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13L24        RLF             AARGB3,F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3123D      RLF             AARGB3,W
                RLF             REMB2,F
                RLF             REMB1,F
                RLF             REMB0,F
                MOVF            BARGB2,W
                BTFSS           AARGB3,LSB
                GOTO            UADD13LD

                SUBWF           REMB2,F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1,F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0,F
                GOTO            UOK13LD

UADD13LD        ADDWF           REMB2,F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F

UOK13LD         RLF             AARGB3,F

                DECFSZ          LOOPCOUNT,F
                GOTO            LOOPU3123D

                BTFSC           AARGB3,LSB
                GOTO            UOK13L
                MOVF            BARGB2,W
                ADDWF           REMB2,F
```

```
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1,F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0,F
UOK13L

                endm

;*****************************************************************************************
;*****************************************************************************************

;       32/24 Bit Signed Fixed Point Divide 32/24 -> 32.24

;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point divisor in BARGB0, BARGB1, BARGB2

;       Use:    CALL    FXD3224S

;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point remainder in REMB0, REMB1, REMB2

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     27+700+5  = 732 clks            A > 0, B > 0
;                       34+700+25 = 759 clks            A > 0, B < 0
;                       34+700+25 = 759 clks            A < 0, B > 0
;                       44+700+5  = 749 clks            A < 0, B < 0
;                                   11 clks             A = 0

;       Min Timing:     27+660+5  = 692 clks            A > 0, B > 0
;                       34+660+25 = 719 clks            A > 0, B < 0
;                       34+660+25 = 749 clks            A < 0, B > 0
;                       44+660+5  = 709 clks            A < 0, B < 0

;       PM: 44+228+24+62 = 358                             DM: 13

FXD3224S        CLRF            SIGN
                CLRF            REMB0                       ; clear partial remainder
                CLRF            REMB1
                CLRF            REMB2
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                      ; clear exception flag

                BTFSS           BARGB0,MSB                  ; if MSB set, negate BARG
                GOTO            CA3224S

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
```

```
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

CA3224S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3224SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C3224SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C3224SX1

C3224S          SDIV3224L

                BTFSC           TEMPB3,LSB               ; test exception flag
                GOTO            C3224SX4

C3224SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB2, F
                BTFSC           _Z
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00

C3224SX1        BTFSS           BARGB0,MSB               ; test BARG exception
                GOTO            C3224SX3
                BTFSC           AARGB0,MSB               ; test AARG exception
                GOTO            C3224SX2
                MOVF            AARGB1,W
                MOVWF           REMB0
                MOVF            AARGB2,W
                MOVWF           REMB1
                MOVF            AARGB3,W
```

```
                MOVWF           REMB2
                BCF             REMB0,MSB
                RLF             AARGB1,F
                RLF             AARGB0,F
                MOVF            AARGB0,W
                MOVWF           AARGB3
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                GOTO            C3224SOK
C3224SX2        CLRF            AARGB3                  ; quotient = 1, remainder = 0
                INCF            AARGB3,F
                CLRF            AARGB2
                CLRF            AARGB1
                CLRF            AARGB0
                RETLW           0x00

C3224SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3224S

C3224SX4        INCF            REMB2,F                 ; increment remainder and test for
                BTFSC           _Z
                INCF            REMB1,F
                BTFSC           _Z
                INCF            REMB0,F
                MOVF            BARGB2,W                ; overflow
                SUBWF           REMB2,W
                BTFSS           _Z
                GOTO            C3224SOK
                MOVF            BARGB1,W
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C3224SOK
                MOVF            BARGB0,W
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C3224SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                CLRF            REMB1
                CLRF            REMB2
                INCF            AARGB3,F                ; remainder, increment quotient and
                BTFSC           _Z
                INCF            AARGB2,F
                BTFSC           _Z
                INCF            AARGB1,F                ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C3224SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************

;       32/24 Bit Unsigned Fixed Point Divide 32/24 -> 32.24

;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
```

```
;       Use:    CALL    FXD3224U

;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     3+862+2 = 867 clks

;       Min Timing:     3+822+2 = 827 clks

;       PM: 3+295+1 = 299               DM: 11

FXD3224U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2

                UDIV3224L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       31/23 Bit Unsigned Fixed Point Divide 31/23 -> 31.23

;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2

;       Use:    CALL    FXD3123U

;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     3+700+2 = 705 clks

;       Min Timing:     3+660+2 = 665 clks

;       PM: 3+228+1 = 232               DM: 10

FXD3123U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2

                UDIV3123L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

                END
```

## E.3 32/16 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd26.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;       $Revision: 2.3 $

;       32/16 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD3216S    595       32 bit/16 bit -> 32.16 signed fixed point divide
;
;       FXD3216U    703       32 bit/16 bit -> 32.16 unsigned fixed point divide
;
;       FXD3115U    541       31 bit/15 bit -> 31.15 unsigned fixed point divide
;
;*****************************************************************************************
;*****************************************************************************************

;       32/16 Bit Division Macros

SDIV3216L       macro

;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+16+6*17+16+8 = 537 clks

;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+15+6*16+15+3 = 501 clks

;       PM: 157                                 DM: 9

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3216A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            SADD26LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26LA

SADD26LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3216A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            SADD26L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26L8

SADD26L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3216B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            SADD26LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26LB

SADD26LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3216B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            SADD26L16
```

```
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26L16

SADD26L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS3216C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            SADD26LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26LC

SADD26LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3216C

                RLF             AARGB3,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            SADD26L24

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26L24

SADD26L24       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26L24        RLF             AARGB3, F

                MOVLW           7
```

```
                MOVWF           LOOPCOUNT

LOOPS3216D      RLF             AARGB3,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB3,LSB
                GOTO            SADD26LD

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK26LD

SADD26LD        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK26LD         RLF             AARGB3, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS3216D

                BTFSC           AARGB3,LSB
                GOTO            SOK26L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK26L

                endm

UDIV3216L       macro

;       Max Timing:     16+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+8 = 699 clks

;       Min Timing:     16+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 663 clks

;       PM: 240                                 DM: 9

                CLRF            TEMP

                RLF             AARGB0,W
                RLF             REMB1, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT
```

```
LOOPU3216A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD26LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26LA

UADD26LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3216A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD26L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26L8

UADD26L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26L8         RLF             AARGB1, F

                MOVLW           7
```

```
                MOVWF           LOOPCOUNT

LOOPU3216B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD26LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26LB

UADD26LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3216B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD26L16

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26L16

UADD26L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26L16        RLF             AARGB2, F
```

```
                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3216C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD26LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26LC

UADD26LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3216C

                RLF             AARGB3,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD26L24

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26L24

UADD26L24       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F
```

```
UOK26L24        RLF             AARGB3, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3216D      RLF             AARGB3,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB3,LSB
                GOTO            UADD26LD

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK26LD

UADD26LD        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK26LD         RLF             AARGB3, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3216D

                BTFSC           AARGB3,LSB
                GOTO            UOK26L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK26L

                endm

UDIV3115L       macro

;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+16+6*17+16+8 = 537 clks

;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+15+6*16+15+3 = 501 clks

;       PM: 157                                 DM: 9

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
```

```
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3115A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD15LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15LA

UADD15LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3115A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD15L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15L8

UADD15L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3115B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD15LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0, F
                GOTO            UOK15LB

UADD15LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3115B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD15L16

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15L16

UADD15L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3115C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD15LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15LC

UADD15LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3115C

                RLF             AARGB3,W
                RLF             REMB1, F
```

```
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD15L24

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15L24

UADD15L24       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15L24        RLF             AARGB3, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU3115D      RLF             AARGB3,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB3,LSB
                GOTO            UADD15LD

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK15LD

UADD15LD        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK15LD         RLF             AARGB3, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU3115D

                BTFSC           AARGB3,LSB
                GOTO            UOK15L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK15L

                endm

;*****************************************************************************************
;*****************************************************************************************

;       32/16 Bit Signed Fixed Point Divide 32/16 -> 32.16

;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
```

```
;               16 bit fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD3216S

;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     26+537+5  = 568 clks            A > 0, B > 0
;                       30+537+22 = 589 clks            A > 0, B < 0
;                       36+537+22 = 595 clks            A < 0, B > 0
;                       40+537+5  = 582 clks            A < 0, B < 0
;                                   10 clks             A = 0

;       Min Timing:     26+501+5  = 532 clks            A > 0, B > 0
;                       30+501+22 = 553 clks            A > 0, B < 0
;                       36+501+22 = 559 clks            A < 0, B > 0
;                       40+501+5  = 546 clks            A < 0, B < 0

;       PM: 40+157+21+55 = 273            DM: 12

FXD3216S        CLRF            SIGN
                CLRF            REMB0                   ; clear partial remainder
                CLRF            REMB1
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                  ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3216S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

CA3216S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3216SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C3216SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
```

```
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C3216SX1

C3216S          SDIV3216L

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C3216SX4

C3216SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                BTFSC           _Z
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00

C3216SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C3216SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C3216SX2
                MOVF            AARGB2,W
                MOVWF           REMB0
                MOVF            AARGB3,W
                MOVWF           REMB1
                BCF             REMB0,MSB
                RLF             AARGB2,F
                RLF             AARGB1,F
                RLF             AARGB0,F
                MOVF            AARGB0,W
                MOVWF           AARGB2
                MOVF            AARGB1,W
                MOVWF           AARGB3
                CLRF            AARGB0
                CLRF            AARGB1
                GOTO            C3216SOK
C3216SX2        CLRF            AARGB3                  ; quotient = 1, remainder = 0
                INCF            AARGB3,F
                CLRF            AARGB2
                CLRF            AARGB1
                CLRF            AARGB0
                RETLW           0x00

C3216SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3216S

C3216SX4        INCF            REMB1,F                 ; increment remainder and test for
```

```
                BTFSC           _Z
                INCF            REMB0,F
                MOVF            BARGB1,W                ; overflow
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C3216SOK
                MOVF            BARGB0,W                ; overflow
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C3216SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                CLRF            REMB1
                INCF            AARGB3,F                ; remainder, increment quotient and
                BTFSC           _Z
                INCF            AARGB2,F
                BTFSC           _Z
                INCF            AARGB1,F                ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C3216SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF

;*****************************************************************************************
;*****************************************************************************************

;       32/16 Bit Unsigned Fixed Point Divide 32/16 -> 32.16

;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD3216U

;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     2+699+2 = 703 clks

;       Max Timing:     2+663+2 = 667 clks

;       PM: 2+240+1 = 243               DM: 9

FXD3216U        CLRF            REMB0
                CLRF            REMB1

                UDIV3216L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       31/15 Bit Unsigned Fixed Point Divide 31/15 -> 31.15

;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD3115U

;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG
```

```
;       Max Timing:     2+537+2 = 541 clks

;       Min Timing:     2+501+2 = 505 clks

;       PM: 2+157+1 = 160               DM: 9

FXD3115U        CLRF            REMB0
                CLRF            REMB1

                UDIV3115L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## E.4 24/24 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd44.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;       $Revision: 2.3 $

;       24/24 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine       Clocks    Function
;
;       FXD2424S      581       24 bit/24 bit -> 24.24 signed fixed point divide
;
;       FXD2424U      676       24 bit/24 bit -> 24.24 unsigned fixed point divide
;
;       FXD2323U      531       23 bit/23 bit -> 23.23 unsigned fixed point divide
;*************************************************************************************
;*************************************************************************************


;       24/24 Bit Division Macros

SDIV2424L       macro

;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+12 = 526 clks

;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+3 = 494 clks

;       PM: 11+3*51+31+12 = 207                                     DM: 12

                MOVF            BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2424A      RLF             AARGB0,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            SADD44LA

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
```

```
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK44LA

SADD44LA        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK44LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2424A

                RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            SADD44L8

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK44L8

SADD44L8        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK44L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2424B      RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            SADD44LB

                SUBWF           REMB2, F
                MOVF            BARGB1,W
```

```
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK44LB

SADD44LB        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK44LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2424B

                RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            SADD44L16

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK44L16

SADD44L16       ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK44L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2424C      RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            SADD44LC
```

```
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK44LC

SADD44LC        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK44LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2424C

                BTFSC           AARGB2,LSB
                GOTO            SOK44L
                MOVF            BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK44L

                endm

UDIV2424L       macro

;       Max Timing:     20+6*28+27+27+6*28+27+27+6*28+27+12 = 671 clks

;       Min Timing:     20+6*27+26+26+6*27+26+26+6*27+26+3 = 639 clks

;       PM: 18+2*76+40+12 = 222                                 DM: 13

                CLRF            TEMP

                RLF             AARGB0,W
                RLF             REMB2, F
                MOVF            BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F

                CLRW
```

```
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2424A      RLF             AARGB0,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD44LA

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK44LA

UADD44LA        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK44LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2424A

                RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD44L8

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK44L8

UADD44L8        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK44L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2424B      RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD44LB

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK44LB

UADD44LB        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F
```

```
UOK44LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2424B

                RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD44L16

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK44L16

UADD44L16       ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK44L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2424C      RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            UADD44LC

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
```

```
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK44LC

UADD44LC        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK44LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2424C

                BTFSC           AARGB2,LSB
                GOTO            UOK44L
                MOVF            BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK44L

                endm

UDIV2323L       macro

;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+12 = 526 clks

;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+3 = 494 clks

;       PM: 11+3*51+31+12 = 207                                 DM: 12

                MOVF            BARGB2,W
                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2323A      RLF             AARGB0,W
```

```
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD33LA

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK33LA

UADD33LA        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK33LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2323A

                RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB0,LSB
                GOTO            UADD33L8

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK33L8

UADD33L8        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK33L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT
```

```
LOOPU2323B      RLF             AARGB1,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD33LB

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK33LB

UADD33LB        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK33LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2323B

                RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB1,LSB
                GOTO            UADD33L16

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK33L16

UADD33L16       ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK33L16        RLF             AARGB2, F
```

```
                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2323C      RLF             AARGB2,W
                RLF             REMB2, F
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB2,W
                BTFSS           AARGB2,LSB
                GOTO            UADD33LC

                SUBWF           REMB2, F
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK33LC

UADD33LC        ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK33LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2323C

                BTFSC           AARGB2,LSB
                GOTO            UOK33L
                MOVF            BARGB2,W
                ADDWF           REMB2, F
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK33L

                endm

;*****************************************************************************************
;*****************************************************************************************

;       24/24 Bit Signed Fixed Point Divide 24/24 -> 24.24

;       Input:  24 bit fixed point dividend in AARGB0, AARGB1,AARGB2
;               24 bit fixed point divisor in BARGB0, BARGB1, BARGB2

;       Use:    CALL    FXD2424S

;       Output: 24 bit fixed point quotient in AARGB0, AARGB1,AARGB2
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
```

```
;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     26+526+5  = 557 clks          A > 0, B > 0
;                       33+526+22 = 581 clks          A > 0, B < 0
;                       33+526+22 = 581 clks          A < 0, B > 0
;                       40+526+5  = 571 clks          A < 0, B < 0
;                                   10 clks           A = 0

;       Min Timing:     26+494+5  = 525 clks          A > 0, B > 0
;                       33+494+22 = 549 clks          A > 0, B < 0
;                       33+494+22 = 549 clks          A < 0, B > 0
;                       40+494+5  = 539 clks          A < 0, B < 0

;       PM: 40+207+21+53 = 321                          DM: 14

FXD2424S        CLRF            SIGN
                CLRF            REMB0                   ; clear partial remainder
                CLRF            REMB1
                CLRF            REMB2
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                  ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA2424S

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
                BTFSC           _Z
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F


CA2424S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C2424SX

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C2424SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C2424SX1

C2424S          SDIV2424L
```

```
                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C2424SX4

C2424SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB2, F
                BTFSC           _Z
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00

C2424SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C2424SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C2424SX2
                MOVF            AARGB0,W
                MOVWF           REMB0                   ; quotient = 0, remainder = AARG
                MOVF            AARGB1,W
                MOVWF           REMB1
                MOVF            AARGB2,W
                MOVWF           REMB2
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                GOTO            C2424SOK
C2424SX2        CLRF            AARGB0                  ; quotient = 1, remainder = 0
                CLRF            AARGB1
                CLRF            AARGB2
                INCF            AARGB2,F
                RETLW           0x00

C2424SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                INCF            TEMPB3,F
                GOTO            C2424S

C2424SX4        INCF            REMB2,F                 ; increment remainder and test for
                BTFSC           _Z                      ; overflow
                INCF            REMB1,F
                BTFSC           _Z
                INCF            REMB0,F
                MOVF            BARGB2,W
                SUBWF           REMB2,W
                BTFSS           _Z
                GOTO            C2424SOK
                MOVF            BARGB1,W
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C2424SOK
                MOVF            BARGB0,W
```

```
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C2424SOK
                CLRF            REMB0               ; if remainder overflow, clear
                CLRF            REMB1               ; remainder, increment quotient and
                CLRF            REMB2
                INCF            AARGB2,F            ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB1,F
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C2424SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************

;       24/24 Bit Unsigned Fixed Point Divide 24/24 -> 24.24

;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2

;       Use:    CALL    FXD2424U

;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     3+671+2 = 676 clks

;       Max Timing:     3+639+2 = 644 clks

;       PM: 3+222+1 = 226               DM: 13

FXD2424U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2

                UDIV2424L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       23/23 Bit Unsigned Fixed Point Divide 23/23 -> 23.23

;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2

;       Use:    CALL    FXD2323U

;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     3+526+2 = 531 clks

;       Min Timing:     3+494+2 = 499 clks

;       PM: 3+207+1 = 211               DM: 12
```

```
FXD2323U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2

                UDIV2323L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

                END
```

## E.5 24/16 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd46.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;       $Revision: 2.3 $

;       24/16 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine       Clocks  Function
;
;       FXD2416S      454     24 bit/16 bit -> 24.16 signed fixed point divide
;
;       FXD2416U      529     24 bit/16 bit -> 24.16 unsigned fixed point divide
;
;       FXD2315U      407     23 bit/15 bit -> 23.15 unsigned fixed point divide

;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Division Macros

SDIV2416L       macro

;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+8 = 403 clks

;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+3 = 375 clks

;       PM: 7+2*40+22+8 = 117                                   DM: 7

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2416A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            SADD46LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK46LA

SADD46LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK46LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2416A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            SADD46L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK46L8

SADD46L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK46L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2416B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            SADD46LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK46LB

SADD46LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK46LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2416B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            SADD46L16
```

```
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK46L16

SADD46L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK46L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS2416C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            SADD46LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK46LC

SADD46LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK46LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS2416C

                BTFSC           AARGB2,LSB
                GOTO            SOK46L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK46L

                endm

UDIV2416L       macro

;       Max Timing:     16+6*22+21+21+6*22+21+21+6*22+21+8 = 525 clks

;       Min Timing:     16+6*21+20+20+6*21+20+20+6*21+20+3 = 497 clks

;       PM: 14+31+27+31+27+31+8 = 169                   DM: 8

                CLRF            TEMP

                RLF             AARGB0,W
```

```
                RLF             REMB1, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F

                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2416A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD46LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK46LA

UADD46LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK46LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2416A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD46L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
```

```
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK46L8

UADD46L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK46L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2416B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD46LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK46LB

UADD46LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK46LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2416B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD46L16

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
```

```
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK46L16

UADD46L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK46L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2416C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD46LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP, F
                GOTO            UOK46LC

UADD46LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP, F

UOK46LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2416C

                BTFSC           AARGB2,LSB
                GOTO            UOK46L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK46L
```

```
                endm

UDIV2315L       macro

;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+8 = 403 clks

;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+3 = 375 clks

;       PM: 7+2*40+22+8 = 117                                   DM: 7

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2315A      RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD35LA

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK35LA

UADD35LA        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK35LA         RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2315A

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB0,LSB
                GOTO            UADD35L8

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK35L8

UADD35L8        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
```

```
UOK35L8         RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2315B      RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD35LB

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK35LB

UADD35LB        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK35LB         RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2315B

                RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB1,LSB
                GOTO            UADD35L16

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK35L16

UADD35L16       ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK35L16        RLF             AARGB2, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU2315C      RLF             AARGB2,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                BTFSS           AARGB2,LSB
                GOTO            UADD35LC

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0, F
                GOTO            UOK35LC

UADD35LC        ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK35LC         RLF             AARGB2, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU2315C

                BTFSC           AARGB2,LSB
                GOTO            UOK35L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK35L

                endm

;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Signed Fixed Point Divide 24/16 -> 24.16

;       Input:  24 bit fixed point dividend in AARGB0, AARGB1,AARGB2
;               16 bit fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD2416S

;       Output: 24 bit fixed point quotient in AARGB0, AARGB1,AARGB2
;               16 bit fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     25+403+5  = 433 clks            A > 0, B > 0
;                       29+403+19 = 451 clks            A > 0, B < 0
;                       32+403+19 = 454 clks            A < 0, B > 0
;                       36+403+5  = 444 clks            A < 0, B < 0
;                                   9 clks              A = 0

;       Min Timing:     25+375+5  = 405 clks            A > 0, B > 0
;                       29+375+19 = 423 clks            A > 0, B < 0
;                       32+375+19 = 426 clks            A < 0, B > 0
;                       36+375+5  = 416 clks            A < 0, B < 0

;       PM: 36+117+18+48 = 219                             DM: 10

FXD2416S        CLRF            SIGN
                CLRF            REMB0                       ; clear partial remainder
                CLRF            REMB1
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
```

```
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                  ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA2416S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

CA2416S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C2416SX

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C2416SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C2416SX1

C2416S          SDIV2416L

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C2416SX4

C2416SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                BTFSC           _Z
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00


C2416SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C2416SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C2416SX2
                MOVF            AARGB1,W
                MOVWF           REMB0
                MOVF            AARGB2,W
                MOVWF           REMB1
```

```
                BCF             REMB0,MSB
                RLF             AARGB1,F
                RLF             AARGB0,F
                MOVF            AARGB0,W
                MOVWF           AARGB2
                CLRF            AARGB0
                CLRF            AARGB1
                GOTO            C2416SOK
C2416SX2        CLRF            AARGB2                  ; quotient = 1, remainder = 0
                INCF            AARGB2,F
                CLRF            AARGB1
                CLRF            AARGB0
                RETLW           0x00

C2416SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                INCF            TEMPB3,F
                GOTO            C2416S

C2416SX4        INCF            REMB1,F                 ; increment remainder and test for
                BTFSC           _Z
                INCF            REMB0,F
                MOVF            BARGB1,W                ; overflow
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C2416SOK
                MOVF            BARGB0,W                ; overflow
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C2416SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                CLRF            REMB1
                INCF            AARGB2,F                ; remainder, increment quotient and
                BTFSC           _Z
                INCF            AARGB1,F                ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C2416SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Unsigned Fixed Point Divide 24/16 -> 24.16

;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD2416U

;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               16 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     2+525+2 = 529 clks

;       Max Timing:     2+497+2 = 501 clks

;       PM: 2+169+1 = 172               DM: 8

FXD2416U        CLRF            REMB0
```

```
                CLRF            REMB1

                UDIV2416L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       23/15 Bit Unsigned Fixed Point Divide 23/15 -> 23.15

;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD2315U

;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     2+403+2 = 407 clks

;       Min Timing:     2+375+2 = 379 clks

;       PM: 2+117+1 = 120               DM: 7

FXD2315U        CLRF            REMB0
                CLRF            REMB1

                UDIV2315L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

                END
```

## E.6 16/16 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd66.a16 2.4 1997/02/27 01:20:22 F.J.Testa Exp $

;       $Revision: 2.4 $

;       16/16 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine      Clocks    Function
;
;       FXD1616S     334       16 bit/16 bit -> 16.16 signed fixed point divide
;
;       FXD1616U     373       16 bit/16 bit -> 16.16 unsigned fixed point divide
;
;       FXD1515U     294       15 bit/15 bit -> 15.15 unsigned fixed point divide
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 65-69 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Division Macros

SDIV1616L       macro

;       Max Timing:     13+14*18+17+8 = 290 clks

;       Min Timing:     13+14*16+15+3 = 255 clks

;       PM: 42                                  DM: 7

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB1, F
                RLF             AARGB0, F

                MOVLW           D'15'
                MOVWF           LOOPCOUNT

LOOPS1616       RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB1,LSB
                GOTO            SADD66L
```

```
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK66LL

SADD66L         ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK66LL         RLF             AARGB1, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS1616

                BTFSC           AARGB1,LSB
                GOTO            SOK66L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK66L

                endm

UDIV1616L       macro

;       restore = 23 clks,  nonrestore = 17 clks

;       Max Timing:     2+15*23+22 = 369 clks

;       Min Timing:     2+15*17+16 = 273 clks

;       PM: 24                                  DM: 7

                MOVLW           D'16'
                MOVWF           LOOPCOUNT

LOOPU1616       RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK66LL
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

                BCF             _C

UOK66LL         RLF             AARGB1, F
```

```
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1616

                endm

UDIV1515L       macro

;       Max Timing:     13+14*18+17+8 = 290 clks

;       Min Timing:     13+14*17+16+3 = 270 clks

;       PM: 42                                  DM: 7

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB1, F
                RLF             AARGB0, F

                MOVLW           D'15'
                MOVWF           LOOPCOUNT

LOOPU1515       RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB1,LSB
                GOTO            UADD55L

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK55LL

UADD55L         ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK55LL         RLF             AARGB1, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1515

                BTFSC           AARGB1,LSB
                GOTO            UOK55L
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK55L
```

```
                endm

SDIV1616        macro

;       Max Timing:     7+10+6*14+14+7*14+8 = 221 clks

;       Min Timing:     7+10+6*13+13+7*13+3 = 202 clks

;       PM: 7+10+6*18+18+7*18+8 = 277   DM: 6

                variable i

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB0,LSB
                GOTO            SADD66#v(i)

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK66#v(i)

SADD66#v(i)     ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK66#v(i)      RLF             AARGB0, F

                variable i = i + 1

                endw

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
```

```
                MOVF            BARGB1,W

                BTFSS           AARGB0,LSB
                GOTO            SADD668

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK668

SADD668         ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK668          RLF             AARGB1, F

                variable i = 9

                while i < 16

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB1,LSB
                GOTO            SADD66#v(i)

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            SOK66#v(i)

SADD66#v(i)     ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

SOK66#v(i)      RLF             AARGB1, F

                variable i = i + 1

                endw

                BTFSC           AARGB1,LSB
                GOTO            SOK66
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
SOK66

                endm

UDIV1616  macro

;       restore = 20 clks,  nonrestore = 14 clks
```

```
;       Max Timing: 16*20 = 320 clks

;       Min Timing: 16*14 = 224 clks

;       PM: 16*20 = 320         DM: 6

                variable         i

                variable i = 0

                while i < 16

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

                BCF             _C

UOK66#v(i)      RLF             AARGB1, F
                RLF             AARGB0, F


                variable i = i + 1

                endw

                endm

UDIV1515        macro

;       Max Timing:     7+10+6*14+14+7*14+8 = 221 clks

;       Min Timing:     7+10+6*13+13+7*13+3 = 202 clks

;       PM:     7+10+6*18+18+7*18+8 = 277       DM: 6

                variable i

                MOVF            BARGB1,W
                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W
```

```
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0

                variable i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB0,LSB
                GOTO            UADD55#v(i)

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK55#v(i)

UADD55#v(i)     ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK55#v(i)      RLF             AARGB0, F

                variable i = i + 1

                endw

                RLF             AARGB1,W
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB0,LSB
                GOTO            UADD558

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK558

UADD558         ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK558          RLF             AARGB1, F

                variable i = 9

                while i < 16

                RLF             AARGB1,W
```

```
                RLF             REMB1, F
                RLF             REMB0, F
                MOVF            BARGB1,W

                BTFSS           AARGB1,LSB
                GOTO            UADD55#v(i)

                SUBWF           REMB1, F
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0, F
                GOTO            UOK55#v(i)

UADD55#v(i)     ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F

UOK55#v(i)      RLF             AARGB1, F

                variable i = i + 1

                endw

                BTFSC           AARGB1,LSB
                GOTO            UOK55
                MOVF            BARGB1,W
                ADDWF           REMB1, F
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0, F
UOK55

                endm

;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Signed Fixed Point Divide 16/16 -> 16.16

;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               16 bit fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD1616S

;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               16 bit fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     24+290+5  = 319 clks            A > 0, B > 0
;                       28+290+16 = 334 clks            A > 0, B < 0
;                       28+290+16 = 334 clks            A < 0, B > 0
;                       32+290+5  = 327 clks            A < 0, B < 0
;                                   8 clks              A = 0

;       Min Timing:     24+255+5  = 284 clks            A > 0, B > 0
;                       28+255+16 = 299 clks            A > 0, B < 0
;                       28+255+16 = 299 clks            A < 0, B > 0
;                       32+255+5  = 292 clks            A < 0, B < 0

;       PM: 32+42+15+39 = 128               DM: 10
```

```
FXD1616S        CLRF            SIGN
                CLRF            REMB0                   ; clear partial remainder
                CLRF            REMB1
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3                  ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA1616S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                BTFSC           _Z
                INCF            BARGB0, F

CA1616S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C1616SX

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C1616SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C1616SX1

C1616S          SDIV1616L

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C1616SX4

C1616SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                BTFSC           _Z
                INCF            REMB0, F

                RETLW           0x00

C1616SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C1616SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C1616SX2
```

```
                MOVF            AARGB0,W
                MOVWF           REMB0                   ; quotient = 0, remainder = AARG
                MOVF            AARGB1,W
                MOVWF           REMB1
                CLRF            AARGB0
                CLRF            AARGB1
                GOTO            C1616SOK
C1616SX2        CLRF            AARGB0                  ; quotient = 1, remainder = 0
                CLRF            AARGB1
                INCF            AARGB1,F
                RETLW            0x00

C1616SX3        COMF            AARGB0,F                ; numerator = 0x7FFF + 1
                COMF            AARGB1,F
                INCF            TEMPB3,F
                GOTO            C1616S

C1616SX4        INCF            REMB1,F                 ; increment remainder and test for
                BTFSC           _Z                      ; overflow
                INCF            REMB0,F
                MOVF            BARGB1,W
                SUBWF           REMB1,W
                BTFSS           _Z
                GOTO            C1616SOK
                MOVF            BARGB0,W
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C1616SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                CLRF            REMB1                   ; remainder, increment quotient and
                INCF            AARGB1,F                ; test for overflow exception
                BTFSC           _Z
                INCF            AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C1616SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF

;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Unsigned Fixed Point Divide 16/16 -> 16.16

;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD1616U

;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               16 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     2+369+2 = 373 clks

;       Min Timing:     2+273+2 = 277 clks

;       PM: 2+24+1 = 27         DM: 7

FXD1616U        CLRF            REMB0
                CLRF            REMB1

                UDIV1616L

                RETLW           0x00
```

```
;*****************************************************************************************
;*****************************************************************************************

;       15/15 Bit Unsigned Fixed Point Divide 15/15 -> 15.15

;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1

;       Use:    CALL    FXD1515U

;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     2+290+2 = 294 clks

;       Min Timing:     2+270+2 = 274 clks

;       PM: 2+42+1 = 45         DM: 7

FXD1515U        CLRF            REMB0
                CLRF            REMB1

                UDIV1515L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

## E.7 16/8 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;        RCS Header $Id: fxd68.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;        $Revision: 2.3 $

;       16/8 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD1608S    203       16 bit/8 bit -> 16.08 signed fixed point divide
;
;       FXD1608U    294       16 bit/8 bit -> 16.08 unsigned fixed point divide
;
;       FXD1607U    174       16 bit/7 bit -> 16.07 unsigned fixed point divide
;
;       FXD1507U    166       15 bit/7 bit -> 15.07 unsigned fixed point divide
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 41-50 clocks can be saved by using the unrolled macros.
;

;*****************************************************************************************
;*****************************************************************************************

;       16/08 Bit Division Macros


SDIV1608L       macro

;       Max Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks

;       Min Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks

;       PM: 42                                  DM: 5

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           6
                MOVWF           LOOPCOUNT

LOOPS1608A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
```

```
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS1608A

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPS1608B      RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS1608B

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1608L  macro

;       Max Timing: 2+7*12+11+3+7*24+23 = 291 clks

;       Min Timing: 2+7*11+10+3+7*17+16 = 227 clks

;       PM: 39                                  DM: 7

                MOVLW           8
                MOVWF           LOOPCOUNT

LOOPU1608A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK68A
                ADDWF           REMB0, F
                BCF             _C
UOK68A          RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1608A

                CLRF            TEMP

                MOVLW           8
                MOVWF           LOOPCOUNT
```

```
LOOPU1608B      RLF             AARGB1,W
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F
                CLRF            AARGB5
                CLRW
                BTFSS           _C
                INCFSZ          AARGB5,W
                SUBWF           TEMP, F

                BTFSC           _C
                GOTO            UOK68B
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                CLRF            AARGB5
                CLRW
                BTFSC           _C
                INCFSZ          AARGB5,W
                ADDWF           TEMP, F

                BCF             _C
UOK68B          RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1608B

                endm

UDIV1607L       macro

;       Max Timing:     7+6*11+10+10+6*11+10+2 = 171 clks

;       Min Timing:     7+6*11+10+10+6*11+10+2 = 171 clks

;       PM: 39                                  DM: 5

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU1607A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1607A

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
```

```
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU1607B      RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1607B

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1507L       macro

;       Max Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks

;       Min Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks

;       PM: 42                                  DM: 5

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           6
                MOVWF           LOOPCOUNT

LOOPU1507A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1507A

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
```

```
                ADDWF           REMB0, F
                RLF             AARGB1, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU1507B      RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU1507B

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

SDIV1608        macro

;       Max Timing:     3+5+14*8+2 = 122 clks

;       Min Timing:     3+5+14*8+2 = 122 clks

;       PM: 122                                 DM: 4

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = i + 1

                endw

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
```

```
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                variable i = 9

                while i < 16

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1608  macro

;       restore = 9/21 clks,  nonrestore = 8/14 clks

;       Max Timing: 8*9+1+8*21 = 241 clks

;       Min Timing: 8*8+1+8*14 = 177 clks

;       PM: 241                                 DM: 6

                variable i = 0

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK68#v(i)
                ADDWF           REMB0, F
                BCF             _C
UOK68#v(i)      RLF             AARGB0, F

                variable i = i + 1

                endw

                CLRF            TEMP

                variable i = 8

                while i < 16

                RLF             AARGB1,W
                RLF             REMB0, F
                RLF             TEMP, F
                MOVF            BARGB0,W
```

```
                SUBWF           REMB0, F
                CLRF            AARGB5
                CLRW
                BTFSS           _C
                INCFSZ          AARGB5,W
                SUBWF           TEMP, F

                BTFSC           _C
                GOTO            UOK68#v(i)
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                CLRF            AARGB5
                CLRW
                BTFSC           _C
                INCFSZ          AARGB5,W
                ADDWF           TEMP, F

                BCF             _C
UOK68#v(i)      RLF             AARGB1, F

                variable i = i + 1

                endw

                endm

UDIV1607        macro

;       Max Timing:     5+15*8+2 = 127 clks

;       Min Timing:     5+15*8+2 = 127 clks

;       PM: 127                                 DM: 4

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                variable i = 1

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = i + 1

                endw

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
```

```
                RLF             AARGB1, F

                variable i = 9

                while i < 16

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1507        macro

;       Max Timing:     3+5+14*8+2 = 122 clks

;       Min Timing:     3+5+14*8+2 = 122 clks

;       PM: 122                                 DM: 4

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                variable i = i + 1

                endw

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
```

```
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                variable i = 9

                while i < 16

                RLF             AARGB1,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLF             AARGB1, F

                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       16/8 Bit Signed Fixed Point Divide 16/8 -> 16.08

;       Input:  16 bit signed fixed point dividend in AARGB0, AARGB1
;               8 bit signed fixed point divisor in BARGB0

;       Use:    CALL    FXD1608S

;       Output: 16 bit signed fixed point quotient in AARGB0, AARGB1
;               8 bit signed fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     23+163+5 = 191 clks             A > 0, B > 0
;                       24+163+13 = 200 clks            A > 0, B < 0
;                       27+163+13 = 203 clks            A < 0, B > 0
;                       28+163+5 = 196 clks             A < 0, B < 0
;                                  7 clks               A = 0

;       Min Timing:     23+163+5 = 191 clks             A > 0, B > 0
;                       24+163+13 = 200 clks            A > 0, B < 0
;                       27+163+13 = 203 clks            A < 0, B > 0
;                       28+163+5 = 196 clks             A < 0, B < 0

;       PM: 28+42+12+34 = 116               DM: 8

FXD1608S        CLRF            SIGN
                CLRF            REMB0                       ; clear partial remainder
                MOVF            AARGB0,W
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           TEMP
```

```
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3          ; clear exception flag

                BTFSS           BARGB0,MSB      ; if MSB set, negate BARG
                GOTO            CA1608S

                COMF            BARGB0, F
                INCF            BARGB0, F

CA1608S         BTFSS           AARGB0,MSB      ; if MSB set, negate AARG
                GOTO            C1608SX

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

C1608SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C1608SX1

C1608S          SDIV1608

                BTFSC           TEMPB3,LSB      ; test exception flag
                GOTO            C1608SX4
C1608SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                BTFSC           _Z
                INCF            AARGB0, F

                COMF            REMB0, F
                INCF            REMB0, F

                RETLW           0x00

C1608SX1        BTFSS           BARGB0,MSB      ; test BARG exception
                GOTO            C1608SX3
                BTFSC           AARGB0,MSB      ; test AARG exception
                GOTO            C1608SX2
                MOVF            AARGB1,W
                MOVWF           REMB0
                BCF             REMB0,MSB
                RLF             AARGB1,F
                RLF             AARGB0,F
                MOVF            AARGB0,W
                MOVWF           AARGB1
                CLRF            AARGB0
                GOTO            C1608SOK
C1608SX2        CLRF            AARGB1          ; quotient = 1, remainder = 0
                INCF            AARGB1,F
                CLRF            AARGB0
                RETLW           0x00

C1608SX3        COMF            AARGB0,F        ; numerator = 0x7FFF + 1
                COMF            AARGB1,F
                INCF            TEMPB3,F
                GOTO            C1608S
```

```
C1608SX4        INCF            REMB0,F             ; increment remainder and test for
                MOVF            BARGB0,W            ; overflow
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C1608SOK
                CLRF            REMB0               ; if remainder overflow, clear
                INCF            AARGB1,F            ; remainder, increment quotient and
                BTFSC           _Z
                INCF            AARGB0,F            ; test for overflow exception
                BTFSS           AARGB0,MSB
                GOTO            C1608SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF

;*****************************************************************************************
;*****************************************************************************************

;       16/8 Bit Unsigned Fixed Point Divide 16/8 -> 16.08

;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               8 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD1608U

;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               8 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+291+2 = 294 clks

;       Min Timing:     1+227+2 = 230 clks

;       PM: 1+39+1 = 41         DM: 7

FXD1608U        CLRF            REMB0

                UDIV1608L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       16/7 Bit Unsigned Fixed Point Divide 16/7 -> 16.07

;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD1607U

;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+171+2 = 174 clks

;       Min Timing:     1+171+2 = 174 clks

;       PM: 1+39+1 = 41         DM: 5

FXD1607U        CLRF            REMB0

                UDIV1607L
```

```
                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       15/7 Bit Unsigned Fixed Point Divide 15/7 -> 15.07

;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD1507U

;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+163+2 = 166 clks

;       Min Timing:     1+163+2 = 166 clks

;       PM: 1+42+1 = 44         DM: 5

FXD1507U        CLRF            REMB0

                UDIV1507L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

                END
```

## E.8 8/8 PIC16C5X/PIC16CXXX Fixed Point Divide Routines

```
;       RCS Header $Id: fxd88.a16 2.3 1996/10/16 14:23:57 F.J.Testa Exp $

;       $Revision: 2.3 $

;       8/8 PIC16 FIXED POINT DIVIDE ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG in AARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD0808S    109       8 bit/8 bit -> 08.08 signed fixed point divide
;
;       FXD0808U    100       8 bit/8 bit -> 08.08 unsigned fixed point divide
;
;       FXD0807U     88       8 bit/7 bit -> 08.07 unsigned fixed point divide
;
;       FXD0707U     80       7 bit/7 bit -> 07.07 unsigned fixed point divide
;
;       The above timings are based on the looped macros. If space permits,
;       approximately 19-25 clocks can be saved by using the unrolled macros.
;

;*************************************************************************************************
;*************************************************************************************************

;       08/08 Bit Division Macros


SDIV0808L       macro

;       Max Timing:     3+5+2+5*11+10+2 = 77 clks

;       Min Timing:     3+5+2+5*11+10+2 = 77 clks

;       PM: 22                                  DM: 4

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           6
                MOVWF           LOOPCOUNT

LOOPS0808A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
```

```
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPS0808A

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

UDIV0808L  macro

;       Max Timing: 2+7*12+11 = 97 clks

;       Min Timing: 2+7*11+10 = 89 clks

;       PM: 13                                  DM: 4

                MOVLW           8
                MOVWF           LOOPCOUNT

LOOPU0808A      RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK88A
                ADDWF           REMB0, F
                BCF             _C
UOK88A          RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU0808A

                endm

UDIV0807L       macro

;       Max Timing:     7+6*11+10+2 = 85 clks

;       Min Timing:     7+6*11+10+2 = 85 clks

;       PM: 19                                  DM: 4

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           7
                MOVWF           LOOPCOUNT

LOOPU0807       RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
```

```
                GOTO            LOOPU0807

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

UDIV0707L       macro

;       Max Timing:     3+5+2+5*11+10+2 = 77 clks

;       Min Timing:     3+5+2+5*11+10+2 = 77 clks

;       PM: 22                                  DM: 4

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                MOVLW           6
                MOVWF           LOOPCOUNT

LOOPU0707       RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                DECFSZ          LOOPCOUNT, F
                GOTO            LOOPU0707

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

SDIV0808        macro

;       Max Timing:     3+5+6*8+2 = 58 clks

;       Min Timing:     3+5+6*8+2 = 58 clks

;       PM: 58                                  DM: 3

                variable i

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F
```

```
                i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                i= i + 1

                endw

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

UDIV0808  macro

;       restore = 9 clks,  nonrestore = 8 clks

;       Max Timing: 8*9 = 72 clks

;       Min Timing: 8*8 = 64 clks

;       PM: 72                                  DM: 3

                variable         i

                i = 0

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK88#v(i)
                ADDWF           REMB0, F
                BCF             _C
UOK88#v(i)      RLF             AARGB0, F

                i= i + 1

                endw

                endm

UDIV0807        macro

;       Max Timing:     5+7*8+2 = 63 clks

;       Min Timing:     5+7*8+2 = 63 clks

;       PM: 63                                  DM: 3

                variable i
```

```
                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                i = 1

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                i= i + 1

                endw

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

UDIV0707        macro

;       Max Timing:     3+5+6*8+2 = 58 clks

;       Min Timing:     3+5+6*8+2 = 58 clks

;       PM: 58                                  DM: 3

                variable i

                MOVF            BARGB0,W
                SUBWF           REMB0, F
                RLF             AARGB0, F

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W
                ADDWF           REMB0, F
                RLF             AARGB0, F

                i = 2

                while i < 8

                RLF             AARGB0,W
                RLF             REMB0, F
                MOVF            BARGB0,W

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLF             AARGB0, F

                i= i + 1

                endw
```

```
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

;*****************************************************************************************
;*****************************************************************************************

;       8/8 Bit Signed Fixed Point Divide 8/8 -> 08.08

;       Input:  8 bit signed fixed point dividend in AARGB0
;               8 bit signed fixed point divisor in BARGB0

;       Use:    CALL    FXD0808S

;       Output: 8 bit signed fixed point quotient in AARGB0
;               8 bit signed fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     21+77+5 = 103 clks              A > 0, B > 0
;                       22+77+10 = 109 clks             A > 0, B < 0
;                       22+77+10 = 109 clks             A < 0, B > 0
;                       23+77+5 = 105 clks              A < 0, B < 0
;                                 6 clks                A = 0

;       Min Timing:     21+77+5 = 103 clks              A > 0, B > 0
;                       22+77+10 = 109 clks             A > 0, B < 0
;                       22+77+10 = 109 clks             A < 0, B > 0
;                       23+77+5 = 105 clks              A < 0, B < 0

;       PM: 23+22+9+25 = 79               DM: 7

FXD0808S        CLRF            SIGN
                CLRF            REMB0             ; clear partial remainder
                MOVF            AARGB0,W
                BTFSC           _Z
                RETLW           0x00

                XORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                COMF            SIGN,F

                CLRF            TEMPB3            ; clear exception flag

                BTFSS           BARGB0,MSB        ; if MSB set, negate BARG
                GOTO            CA0808S

                COMF            BARGB0, F
                INCF            BARGB0, F

CA0808S         BTFSS           AARGB0,MSB        ; if MSB set, negate AARG
                GOTO            C0808SX

                COMF            AARGB0, F
                INCF            AARGB0, F

C0808SX         MOVF            AARGB0,W
                IORWF           BARGB0,W
                MOVWF           TEMP
                BTFSC           TEMP,MSB
                GOTO            C0808SX1

C0808S          SDIV0808L
```

```
                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C0808SX4

C0808SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB0, F
                INCF            AARGB0, F

                COMF            REMB0, F
                INCF            REMB0, F

                RETLW           0x00

C0808SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C0808SX3
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C0808SX2
                MOVF            AARGB0,W                ; quotient = 0, remainder = AARG
                MOVWF           REMB0
                CLRF            AARGB0
                GOTO            C0808SOK
C0808SX2        CLRF            AARGB0                  ; quotient = 1, remainder = 0
                INCF            AARGB0,F
                RETLW           0x00

C0808SX3        COMF            AARGB0,F                ; numerator = 0x7F + 1
                INCF            TEMPB3,F
                GOTO            C0808S

C0808SX4        INCF            REMB0,F                 ; increment remainder and test for
                MOVF            BARGB0,W                ; overflow
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            C0808SOK
                CLRF            REMB0                   ; if remainder overflow, clear
                INCF            AARGB0,F                ; remainder, increment quotient and
                BTFSS           AARGB0,MSB              ; test for overflow exception
                GOTO            C0808SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF



;*****************************************************************************************
;*****************************************************************************************

;       8/8 Bit Unsigned Fixed Point Divide 8/8 -> 08.08

;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               8 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD0808U

;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               8 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+97+2 = 100 clks

;       Min Timing:     1+89+2 = 92 clks

;       PM: 1+13+1 = 15         DM: 4
```

```
FXD0808U        CLRF            REMB0

                UDIV0808L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       8/7 Bit Unsigned Fixed Point Divide 8/7 -> 08.07

;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD0807U

;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+85+2 = 88 clks

;       Min Timing:     1+85+2 = 88 clks

;       PM: 1+19+1 = 21         DM: 4

FXD0807U        CLRF            REMB0

                UDIV0807L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************

;       7/7 Bit Unsigned Fixed Point Divide 7/7 -> 07.07

;       Input:  7 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0

;       Use:    CALL    FXD0707U

;       Output: 7 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0

;       Result: AARG, REM  <--  AARG / BARG

;       Max Timing:     1+77+2 = 80 clks

;       Min Timing:     1+77+2 = 80 clks

;       PM: 1+22+1 = 44         DM: 4

FXD0707U        CLRF            REMB0

                UDIV0707L

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

**NOTES:**

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

## APPENDIX F: PIC17CXXX MULTIPLY ROUTINES

```
;        RCS Header $Id: fxm.a17 2.2 1996/06/11 21:42:11 F.J.Testa Exp $

;        $Revision: 2.2 $

;       PIC17 FIXED POINT MULTIPLY ROUTINES
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: product AARG*BARG in AARG
;
;       All timings are worst case cycle counts
;
;       Routine     Clocks     Function
;
;
;       FXM0808S      11       08x08 -> 16 bit signed fixed point multiply
;
;       FXM0808U       6       08x08 -> 16 bit unsigned fixed point multiply
;
;       FXM1608S      21       16x08 -> 24 bit signed fixed point multiply
;
;       FXM1608U      12       16x08 -> 24 bit unsigned fixed point multiply
;
;       FXM1616S      39       16x16 -> 32 bit signed fixed point multiply
;
;       FXM1616U      26       16x16 -> 32 bit unsigned fixed point multiply
;
;       FXM2416S      56       24x16 -> 40 bit signed fixed point multiply
;
;       FXM2416U      40       24x16 -> 40 bit unsigned fixed point multiply
;
;       FXM2424S      81       24x24 -> 48 bit signed fixed point multiply
;
;       FXM2424U      65       24x24 -> 48 bit unsigned fixed point multiply
;
;       FXM3216S      73       32x16 -> 48 bit signed fixed point multiply
;
;       FXM3216U      54       32x16 -> 48 bit unsigned fixed point multiply
;
;       FXM3224S     108       32x24 -> 56 bit signed fixed point multiply
;
;       FXM3224U      90       32x24 -> 56 bit unsigned fixed point multiply
;
;       FXM3232S     145       32x32 -> 64 bit signed fixed point multiply
;
;       FXM3232U     125       32x32 -> 64 bit unsigned fixed point multiply
;
;*****************************************************************************************
;*****************************************************************************************
;
;       8x8 Bit Signed Fixed Point Multiply 08 x 08 -> 16
;
;       Input:  8 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0
;
;       Use:    CALL    FXM0808S
;
;       Output: 16 bit signed fixed point product in AARGB0, AARGB1
;
;       Result: AARG  <--  AARG * BARG
;
```

```
;       Max Timing:     11 clks
;
;       Min Timing:     11 clks
;
;       PM: 10              DM: 3
;
FXM0808S        MOVFP           AARGB0,WREG
                MULWF           BARGB0
                BTFSC           BARGB0,MSB
                SUBWF           PRODH,F
                MOVFP           BARGB0,WREG
                BTFSC           AARGB0,MSB
                SUBWF           PRODH,F
                MOVPF           PRODH,AARGB0
                MOVPF           PRODL,AARGB1
                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       8x8 Bit Unsigned Fixed Point Multiply 08 x 08 -> 16
;
;       Input:  8 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0
;
;       Use:    CALL    FXM0808U
;
;       Output: 16 bit unsigned fixed point product in AARGB0, AARGB1
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     6 clks
;
;       Min Timing:     6 clks
;
;       PM: 5               DM: 3
;
FXM0808U        MOVFP           BARGB0,WREG
                MULWF           AARGB0
                MOVPF           PRODH,AARGB0
                MOVPF           PRODL,AARGB1
                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16x8 Bit Signed Fixed Point Multiply 16 x 08 -> 24
;
;       Input:  16 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0
;
;       Use:    CALL    FXM1608S
;
;       Output: 24 bit signed fixed point product in AARGB0, AARGB1
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     21 clks
;
;       Min Timing:     18 clks
;
;       PM: 20              DM: 4
;
FXM1608S        MOVFP           BARGB0,WREG
                MULWF           AARGB1
                MOVPF           AARGB1,TEMP
```

```
                MOVPF           PRODH,AARGB1
                MOVPF           PRODL,AARGB2
                MULWF           AARGB0
                BTFSC           AARGB0,MSB
                SUBWF           PRODH,F
                BTFSS           BARGB0,MSB
                GOTO            SIGN1608OK
                MOVFP           TEMP,WREG
                SUBWF           AARGB1,F
                MOVFP           AARGB0,WREG
                SUBWFB          PRODH,F

SIGN1608OK      CLRF            AARGB0,F
                MOVFP           PRODL,WREG
                ADDWF           AARGB1,F
                MOVFP           PRODH,WREG
                ADDWFC          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16x8 Bit Unsigned Fixed Point Multiply 16 x 08 -> 24
;
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0
;
;       Use:    CALL    FXM1608U
;
;       Output: 24 bit unsigned fixed point product in AARGB0, AARGB1, AARGB2
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     12 clks
;
;       Min Timing:     12 clks
;
;       PM: 11              DM: 4
;
FXM1608U        MOVFP           BARGB0,WREG
                MULWF           AARGB1
                MOVPF           PRODH,AARGB1
                MOVPF           PRODL,AARGB2
                MULWF           AARGB0
                MOVPF           PRODH,AARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16x16 Bit Signed Fixed Point Multiply 16 x 16 -> 32
;
;       Input:  16 bit signed fixed point multiplicand in AARGB0, AARGB1
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM1616S
;
;       Output: 32 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3
;
```

```
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     39 clks
;
;       Min Timing:     31 clks
;
;       PM: 38              DM: 8
;
FXM1616S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1

                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB2
                MOVPF           PRODL,AARGB3

                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB0
                MOVPF           PRODL,AARGB1

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                BTFSS           BARGB0,MSB
                GOTO            TSIGN1616A
                MOVFP           TEMPB1,WREG
                SUBWF           AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN1616A      BTFSS           TEMPB0,MSB
                RETLW           0x00
                MOVFP           BARGB1,WREG
                SUBWF           AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16x16 Bit Unsigned Fixed Point Multiply 16 x 16 -> 32
;
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0, AARGB1
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM1616U
;
;       Output: 32 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3
```

```
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     26 clks
;
;       Min Timing:     26 clks
;
;       PM: 25              DM: 7
;
;
FXM1616U        MOVPF           AARGB1,TEMPB1

                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB2
                MOVPF           PRODL,AARGB3

                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB0
                MOVPF           PRODL,AARGB1

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                RETLW           0x00

;*************************************************************************************************
;*************************************************************************************************
;
;       24x16 Bit Signed Fixed Point Multiply 24 x 16 -> 40
;
;       Input:  24 bit signed fixed point multiplicand in AARGB0, AARGB1, AARGB2
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM2416S
;
;       Output: 40 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     56 clks
;
;       Min Timing:     46 clks
;
;       PM: 55              DM: 10
;
FXM2416S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
```

```
                MOVFP           AARGB2,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB3
                MOVPF           PRODL,AARGB4

                MOVFP           AARGB1,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB1
                MOVPF           PRODL,AARGB2

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                CLRF            AARGB0,W
                ADDWFC          AARGB0,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                BTFSS           BARGB0,MSB
                GOTO            TSIGN2416A
                MOVFP           TEMPB2,WREG
                SUBWF           AARGB2,F
                MOVFP           TEMPB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN2416A      BTFSS           TEMPB0,MSB
                RETLW           0x00
                MOVFP           BARGB1,WREG
                SUBWF           AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       24x16 Bit Unsigned Fixed Point Multiply 24 x 16 -> 40
;
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0, AARGB1, AARGB2
```

```
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM2416U
;
;       Output: 40 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     40 clks
;
;       Min Timing:     40 clks
;
;       PM: 39              DM: 8
;
;
FXM2416U        MOVPF           AARGB2,TEMPB2

                MOVFP           AARGB2,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB3
                MOVPF           PRODL,AARGB4

                MOVFP           AARGB1,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB1
                MOVPF           PRODL,AARGB2

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                CLRF            AARGB0,W
                ADDWFC          AARGB0,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       24x24 Bit Signed Fixed Point Multiply 24 x 24 -> 48
```

```
;
;       Input:  24 bit signed fixed point multiplicand in AARGB0, AARGB1, AARGB2
;               24 bit signed fixed point multiplier in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXM2424S
;
;       Output: 48 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     81 clks
;
;       Min Timing:     69 clks
;
;       PM: 80              DM: 12
;
FXM2424S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2

                MOVFP           AARGB2,WREG
                MULWF           BARGB2
                MOVPF           PRODH,AARGB4
                MOVPF           PRODL,AARGB5

                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB2
                MOVPF           PRODL,AARGB3

                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                CLRF            AARGB1,W
                ADDWFC          AARGB1,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
```

```
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            AARGB0,W
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB0,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                BTFSS           BARGB0,MSB
                GOTO            TSIGN2424A
                MOVFP           TEMPB2,WREG
                SUBWF           AARGB2,F
                MOVFP           TEMPB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN2424A      BTFSS           TEMPB0,MSB
                RETLW           0x00
                MOVFP           BARGB2,WREG
                SUBWF           AARGB2,F
                MOVFP           BARGB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       24x24 Bit Unsigned Fixed Point Multiply 24 x 24 -> 48
;
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXM2424U
;
;       Output: 48 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     65 clks
;
;       Min Timing:     65 clks
;
;       PM: 64              DM: 12
;
;
FXM2424U        MOVPF           AARGB0,TEMPB0
```

```
MOVPF           AARGB1,TEMPB1
MOVPF           AARGB2,TEMPB2

MOVFP           AARGB2,WREG
MULWF           BARGB2
MOVPF           PRODH,AARGB4
MOVPF           PRODL,AARGB5

MOVFP           AARGB1,WREG
MULWF           BARGB1
MOVPF           PRODH,AARGB2
MOVPF           PRODL,AARGB3

MULWF           BARGB2
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F

MOVFP           TEMPB2,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F

MOVFP           AARGB0,WREG
MULWF           BARGB2
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
MOVFP           AARGB0,WREG
MULWF           BARGB1
CLRF            AARGB1,W
ADDWFC          AARGB1,F
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F

MOVFP           TEMPB2,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
CLRF            AARGB0,W
ADDWFC          AARGB1,F
ADDWFC          AARGB0,F

MOVFP           TEMPB1,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F
CLRF            WREG,F
ADDWFC          AARGB0,F

MOVFP           TEMPB0,WREG
MULWF           BARGB0
```

```
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       32x16 Bit Signed Fixed Point Multiply 32 x 16 -> 48
;
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM3216S
;
;       Output: 48 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     73 clks
;
;       Min Timing:     61 clks
;
;       PM: 72              DM: 12
;
FXM3216S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB4
                MOVPF           PRODL,AARGB5

                MOVFP           AARGB2,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB2
                MOVPF           PRODL,AARGB3

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
```

```
                ADDWFC          AARGB2,F
                MOVFP           AARGB1,WREG
                MULWF           BARGB0
                CLRF            AARGB1,W
                ADDWFC          AARGB1,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                CLRF            AARGB0,W
                ADDWFC          AARGB0,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                BTFSS           BARGB0,MSB
                GOTO            TSIGN3216A
                MOVFP           TEMPB3,WREG
                SUBWF           AARGB3,F
                MOVFP           TEMPB2,WREG
                SUBWFB          AARGB2,F
                MOVFP           TEMPB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN3216A      BTFSS           TEMPB0,MSB
                RETLW           0x00
                MOVFP           BARGB1,WREG
                SUBWF           AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       32x16 Bit Unsigned Fixed Point Multiply 32 x 16 -> 48
;
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;
;       Use:    CALL    FXM3216U
;
;       Output: 48 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     54 clks
;
;       Min Timing:     54 clks
;
;       PM: 53              DM: 9
```

```
;
;
FXM3216U        MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB4
                MOVPF           PRODL,AARGB5

                MOVFP           AARGB2,WREG
                MULWF           BARGB0
                MOVPF           PRODH,AARGB2
                MOVPF           PRODL,AARGB3

                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                MOVFP           AARGB1,WREG
                MULWF           BARGB0
                CLRF            AARGB1,W
                ADDWFC          AARGB1,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                CLRF            AARGB0,W
                ADDWFC          AARGB0,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
```

```
;       32x24 Bit Signed Fixed Point Multiply 32 x 24 -> 56
;
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit signed fixed point multiplier in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXM3224S
;
;       Output: 56 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     108 clks
;
;       Min Timing:     94 clks
;
;       PM: 107              DM: 15
;
FXM3224S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB2
                MOVPF           PRODH,AARGB5
                MOVPF           PRODL,SIGN

                MOVFP           AARGB2,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB3
                MOVPF           PRODL,AARGB4

                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                CLRF            WREG,F
                ADDWFC          AARGB3,F

                MOVF            TEMPB3,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                CLRF            WREG,F
                ADDWFC          AARGB3,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                MOVFP           AARGB1,WREG
                MULWF           BARGB1
                CLRF            AARGB2,W
                ADDWFC          AARGB2,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
```

```
                MOVFP           TEMPB3,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            AARGB1,W
                ADDWFC          AARGB2,F
                ADDWFC          AARGB1,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F

                MOVFP           AARGB0,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                MOVFP           AARGB0,WREG
                MULWF           BARGB0
                CLRF            AARGB0,W
                ADDWFC          AARGB0,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB1,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB0,F

                MOVFP           TEMPB0,WREG
                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F

                MOVFP           SIGN,AARGB6

                BTFSS           BARGB0,MSB
                GOTO            TSIGN3224A
                MOVFP           TEMPB3,WREG
                SUBWF           AARGB3,F
                MOVFP           TEMPB2,WREG
                SUBWFB          AARGB2,F
                MOVFP           TEMPB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN3224A      BTFSS           TEMPB0,MSB
                RETLW           0x00
```

```
                MOVFP           BARGB2,WREG
                SUBWF           AARGB2,F
                MOVFP           BARGB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       32x24 Bit Unsigned Fixed Point Multiply 32 x 24 -> 56
;
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXM3224U
;
;       Output: 56 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     90 clks
;
;       Min Timing:     90 clks
;
;       PM: 89              DM: 15
;
;
FXM3224U        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB2
                MOVPF           PRODH,AARGB5
                MOVPF           PRODL,SIGN

                MOVFP           AARGB2,WREG
                MULWF           BARGB1
                MOVPF           PRODH,AARGB3
                MOVPF           PRODL,AARGB4

                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                CLRF            WREG,F
                ADDWFC          AARGB3,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                CLRF            WREG,F
                ADDWFC          AARGB3,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB2
```

```
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
MOVFP           AARGB1,WREG
MULWF           BARGB1
CLRF            AARGB2,W
ADDWFC          AARGB2,F
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F

MOVFP           TEMPB3,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            AARGB1,W
ADDWFC          AARGB2,F
ADDWFC          AARGB1,F

MOVFP           TEMPB2,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
CLRF            WREG,F
ADDWFC          AARGB1,F

MOVFP           TEMPB1,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F

MOVFP           AARGB0,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F
MOVFP           AARGB0,WREG
MULWF           BARGB0
CLRF            AARGB0,W
ADDWFC          AARGB0,F
MOVPF           PRODL,WREG
ADDWF           AARGB1,F
MOVPF           PRODH,WREG
ADDWFC          AARGB0,F

MOVFP           TEMPB0,WREG
MULWF           BARGB2
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
CLRF            WREG,F
ADDWFC          AARGB1,F
ADDWFC          AARGB0,F

MOVFP           SIGN,AARGB6
```

```
                RETLW   0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       32x32 Bit Signed Fixed Point Multiply 32 x 32 -> 64
;
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               32 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2, BARGB3
;
;       Use:    CALL    FXM3232S
;
;       Output: 64 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6, AARGB7
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     145 clks
;
;       Min Timing:     129 clks
;
;       PM: 144              DM: 18
;
FXM3232S        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB3
                MOVPF           PRODL,TBLPTRL
                MOVPF           PRODH,TBLPTRH

                MOVFP           AARGB2,WREG
                MULWF           BARGB2
                MOVPF           PRODL,AARGB5
                MOVPF           PRODH,AARGB4

                MULWF           BARGB3
                MOVPF           PRODL,WREG
                ADDWF           TBLPTRH,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB5,F
                CLRF            WREG,F
                ADDWFC          AARGB4,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           TBLPTRH,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB5,F
                CLRF            WREG,F
                ADDWFC          AARGB4,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB3
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                MOVFP           AARGB1,WREG
                MULWF           BARGB2
                CLRF            AARGB3,W
```

```
ADDWFC          AARGB3,F
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F

MOVFP           TEMPB3,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB5,F
MOVPF           PRODH,WREG
ADDWFC          AARGB4,F
CLRF            AARGB2,W
ADDWFC          AARGB3,F
ADDWFC          AARGB2,F

MOVFP           TEMPB2,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F

MOVFP           TEMPB1,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F

MOVFP           AARGB0,WREG
MULWF           BARGB2
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
MOVFP           AARGB0,WREG
MULWF           BARGB1
CLRF            AARGB1,W
ADDWFC          AARGB1,F
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F

MOVFP           TEMPB0,WREG
MULWF           BARGB3
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F
ADDWFC          AARGB1,F

MOVFP           TEMPB0,WREG
MULWF           BARGB0
MOVPF           PRODH,AARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB1,F
CLRF            WREG,F
ADDWFC          AARGB0,F

MOVFP           TEMPB3,WREG
```

```
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F
                CLRF            WREG,F
                ADDWFC          AARGB2,F
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                MOVFP           TBLPTRL,AARGB7
                MOVFP           TBLPTRH,AARGB6

                BTFSS           BARGB0,MSB
                GOTO            TSIGN3232A
                MOVFP           TEMPB3,WREG
                SUBWF           AARGB3,F
                MOVFP           TEMPB2,WREG
                SUBWFB          AARGB2,F
                MOVFP           TEMPB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           TEMPB0,WREG
                SUBWFB          AARGB0,F

TSIGN3232A      BTFSS           TEMPB0,MSB
                RETLW           0x00
                MOVFP           BARGB3,WREG
                SUBWF           AARGB3,F
                MOVFP           BARGB2,WREG
                SUBWFB          AARGB2,F
                MOVFP           BARGB1,WREG
                SUBWFB          AARGB1,F
                MOVFP           BARGB0,WREG
                SUBWFB          AARGB0,F

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       32x32 Bit Unsigned Fixed Point Multiply 32 x 32 -> 64
;
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               32 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2, BARGB3
```

```
;
;       Use:    CALL    FXM3232U
;
;       Output: 64 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6, AARGB7
;
;       Result: AARG  <--  AARG * BARG
;
;       Max Timing:     125 clks
;
;       Min Timing:     125 clks
;
;       PM: 124              DM: 18
;
;
FXM3232U        MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3

                MOVFP           AARGB3,WREG
                MULWF           BARGB3
                MOVPF           PRODL,TBLPTRL
                MOVPF           PRODH,TBLPTRH

                MOVFP           AARGB2,WREG
                MULWF           BARGB2
                MOVPF           PRODL,AARGB5
                MOVPF           PRODH,AARGB4

                MULWF           BARGB3
                MOVPF           PRODL,WREG
                ADDWF           TBLPTRH,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB5,F
                CLRF            WREG,F
                ADDWFC          AARGB4,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB2
                MOVPF           PRODL,WREG
                ADDWF           TBLPTRH,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB5,F
                CLRF            WREG,F
                ADDWFC          AARGB4,F

                MOVFP           AARGB1,WREG
                MULWF           BARGB3
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB4,F
                MOVFP           AARGB1,WREG
                MULWF           BARGB2
                CLRF            AARGB3,W
                ADDWFC          AARGB3,F
                MOVPF           PRODL,WREG
                ADDWF           AARGB4,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB3,F

                MOVFP           TEMPB3,WREG
                MULWF           BARGB1
                MOVPF           PRODL,WREG
                ADDWF           AARGB5,F
```

```
MOVPF           PRODH,WREG
ADDWFC          AARGB4,F
CLRF            AARGB2,W
ADDWFC          AARGB3,F
ADDWFC          AARGB2,F

MOVFP           TEMPB2,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F

MOVFP           TEMPB1,WREG
MULWF           BARGB1
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F

MOVFP           AARGB0,WREG
MULWF           BARGB2
MOVPF           PRODL,WREG
ADDWF           AARGB3,F
MOVPF           PRODH,WREG
ADDWFC          AARGB2,F
MOVFP           AARGB0,WREG
MULWF           BARGB1
CLRF            AARGB1,W
ADDWFC          AARGB1,F
MOVPF           PRODL,WREG
ADDWF           AARGB2,F
MOVPF           PRODH,WREG
ADDWFC          AARGB1,F

MOVFP           TEMPB0,WREG
MULWF           BARGB3
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F
ADDWFC          AARGB1,F

MOVFP           TEMPB0,WREG
MULWF           BARGB0
MOVPF           PRODH,AARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB1,F
CLRF            WREG,F
ADDWFC          AARGB0,F

MOVFP           TEMPB3,WREG
MULWF           BARGB0
MOVPF           PRODL,WREG
ADDWF           AARGB4,F
MOVPF           PRODH,WREG
ADDWFC          AARGB3,F
CLRF            WREG,F
ADDWFC          AARGB2,F
ADDWFC          AARGB1,F
ADDWFC          AARGB0,F
```

```
                MOVFP           TEMPB2,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB3,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB2,F
                CLRF            WREG,F
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F

                MOVFP           TEMPB1,WREG
                MULWF           BARGB0
                MOVPF           PRODL,WREG
                ADDWF           AARGB2,F
                MOVPF           PRODH,WREG
                ADDWFC          AARGB1,F
                CLRF            WREG,F
                ADDWFC          AARGB0,F

                MOVFP           TBLPTRL,AARGB7
                MOVFP           TBLPTRH,AARGB6

                RETLW           0x00

;*************************************************************************************************
;*************************************************************************************************
```

**NOTES:**

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

# APPENDIX G: PIC17CXXX DIVIDE ROUTINES

**Table of Contents for Appendix G**



## G.1 PIC17CXXX Fixed Point Divide Routines A

```
;       RCS Header $Id: fxda.a17 2.4 1997/03/22 03:11:13 F.J.Testa Exp $

;       $Revision: 2.4 $

;       PIC17 FIXED POINT DIVIDE ROUTINES A
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;
;       FXD3232S    630       32 bit/32 bit -> 32.32 signed fixed point divide
;
;       FXD3232U    683       32 bit/32 bit -> 32.32 unsigned fixed point divide
;
;       FXD3231U    588       32 bit/31 bit -> 32.31 unsigned fixed point divide
;
;       FXD3131U    579       31 bit/31 bit -> 31.31 unsigned fixed point divide
;
;
;       FXD3224S    529       32 bit/24 bit -> 32.24 signed fixed point divide
;
;       FXD3224U    584       32 bit/24 bit -> 32.24 unsigned fixed point divide
;
;       FXD3223U    489       32 bit/23 bit -> 32.23 unsigned fixed point divide
;
;       FXD3123U    481       31 bit/23 bit -> 31.23 unsigned fixed point divide
;

;*************************************************************************************************
;*************************************************************************************************
;
;       32/32 Bit Division Macros
;
SDIV3232        macro
;
;       Max Timing:     9+14+30*18+10 = 573 clks
;
;       Min Timing:     9+14+30*17+3 = 536 clks
;
;       PM: 9+14+30*24+10 = 753         DM: 12
;
                variable i

                MOVFP           BARGB3,WREG
```

```
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK22#v(i)

SADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK22#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
```

```
                GOTO            SADD228
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK228

SADD228         ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK228          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK22#v(i)

SADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK22#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD2216
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
```

```
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK2216

SADD2216        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2216         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK22#v(i)

SADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK22#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD2224
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0, F
                GOTO            SOK2224

SADD2224        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2224         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB3,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK22#v(i)

SADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK22#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            SOK22
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
SOK22

                endm


UDIV3232  macro
;
;       restore = 25/30 clks,  nonrestore = 17/20 clks
```

```
;
;       Max Timing: 16*25+1+16*30 = 881 clks
;
;       Min Timing: 16*17+1+16*20 = 593 clks
;
;       PM:  16*25+1+16*30 = 881               DM: 13
;
                variable        i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK22#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                variable i = D'8'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
```

```
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK22#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D'16'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK22#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                variable i = D'24'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
```

```
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK22#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                endm

NDIV3232        macro

;       Max Timing:     16+31*21+10 = 677 clks
;
;       Min Timing: 16+31*20+3 = 639 clks
;
;       PM: 16+31*29+10 = 925           DM: 13
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
```

```
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK22#v(i)

NADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK22#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD228
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK228

NADD228         ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK228          RLCF            AARGB1, F

                variable i = D'9'
```

```
                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK22#v(i)

NADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK22#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD2216
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2216

NADD2216        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
```

```
                ADDWFC          TEMP, F

NOK2216         RLCF            AARGB2, F

                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK22#v(i)

NADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK22#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD2224
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2224

NADD2224        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
```

```
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK2224         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB3,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK22#v(i)

NADD22#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK22#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            NOK22
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK22
                endm


UDIV3231        macro
```

```
;
;       Max Timing:     14+31*18+10 = 582 clks
;
;       Min Timing:     14+31*17+3 = 544 clks
;
;       PM: 14+31*24+10 = 768                   DM: 12
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK21#v(i)

UADD21#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK21#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
```

```
                GOTO            UADD218
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK218

UADD218         ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK218          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK21#v(i)

UADD21#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK21#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD2116
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
```

```
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK2116

UADD2116        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2116         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK21#v(i)

UADD21#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK21#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD2124
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0, F
                GOTO            UOK2124

UADD2124        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2124         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK21#v(i)

UADD21#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK21#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK21
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK21

                endm


UDIV3131        macro
;
;       Max Timing:     9+14+30*18+10 = 573 clks
```

```
;
;       Min Timing:     9+14+30*17+3 = 536 clks
;
;       PM: 9+14+30*24+10 = 753         DM: 12
;
                variable i

                MOVFP           BARGB3,WREG
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK11#v(i)

UADD11#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK11#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw
```

```
                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD118
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK118

UADD118         ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK118          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK11#v(i)

UADD11#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK11#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
```

```
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD1116
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK1116

UADD1116        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1116         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK11#v(i)

UADD11#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK11#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
```

```
                BTFSS           AARGB2,LSB
                GOTO            UADD1124
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK1124

UADD1124        ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1124         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB3, F
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB3,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3, F
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK11#v(i)

UADD11#v(i)     ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK11#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK11
                MOVFP           BARGB3,WREG
                ADDWF           REMB3, F
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
```

```
UOK11

                endm


;*****************************************************************************************
;*****************************************************************************************
;
;       32/24 Bit Division Macros
;
SDIV3224        macro
;
;       Max Timing:     7+11+30*15+8 = 476 clks
;
;       Min Timing:     7+11+30*14+3 = 441 clks
;
;       PM: 7+11+30*19+8 = 596          DM: 10
;
                variable i

                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK24#v(i)

SADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK24#v(i)      RLCF            AARGB0, F
```

```
                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD248
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK248

SADD248         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK248          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK24#v(i)

SADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK24#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD2416
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK2416

SADD2416        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2416         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK24#v(i)

SADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK24#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD2424
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK2424

SADD2424        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2424         RLCF            AARGB3, F

                variable i = D'25'
```

```
                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB3,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK24#v(i)

SADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK24#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            SOK24
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
SOK24

                endm


UDIV3224  macro
;
;       restore = 20/25 clks,  nonrestore = 14/17 clks
;
;       Max Timing: 16*20+1+16*25 = 721 clks
;
;       Min Timing: 16*14+1+16*17 = 497 clks
;
;       PM:  16*20+1+16*25 = 721               DM: 11
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK24#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                variable i = D'8'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK24#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D'16'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
```

```
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK24#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                variable i = D'24'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK24#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                endm

NDIV3224        macro
;
;       Max Timing:     13+31*18+8 = 579 clks
;
;       Min Timing: 13+31*17+3 = 543 clks
;
;       PM: 13+31*24+8 = 765            DM: 11
;
                variable i
```

```
                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK24#v(i)

NADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK24#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD248
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK248
```

```
NADD248         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK248          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK24#v(i)

NADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK24#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD2416
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2416

NADD2416        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
```

```
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK2416         RLCF            AARGB2, F

                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK24#v(i)

NADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK24#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD2424
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2424

NADD2424        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
```

```
NOK2424         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB3,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK24#v(i)

NADD24#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK24#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            NOK24
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK24
                endm


UDIV3223        macro
;
;       Max Timing:     11+31*15+8 = 484 clks
;
;       Min Timing:     11+31*14+3 = 448 clks
;
;       PM: 11+31*19+8 = 608                    DM: 10
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
```

```
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK23#v(i)

UADD23#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK23#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD238
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK238

UADD238         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK238          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
```

```
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK23#v(i)

UADD23#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK23#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD2316
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK2316

UADD2316        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2316         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK23#v(i)

UADD23#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
```

```
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK23#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD2324
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK2324

UADD2324        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2324         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK23#v(i)

UADD23#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK23#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK23
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
```

```
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK23

                endm


UDIV3123        macro
;
;       Max Timing:     7+11+30*15+8 = 476 clks
;
;       Min Timing:     7+11+30*14+3 = 441 clks
;
;       PM: 7+11+30*19+8 = 596          DM: 10
;
                variable i

                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK13#v(i)

UADD13#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK13#v(i)      RLCF            AARGB0, F

                variable i = i + 1
```

```
                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD138
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK138

UADD138         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK138          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK13#v(i)

UADD13#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK13#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD1316
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
```

```
                GOTO            UOK1316

UADD1316        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1316         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK13#v(i)

UADD13#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK13#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD1324
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK1324

UADD1324        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1324         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'
```

```
                RLCF            AARGB3,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK13#v(i)

UADD13#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK13#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK13
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK13

                endm


;*****************************************************************************************
;*****************************************************************************************
;
;       32/32 Bit Signed Fixed Point Divide 32/32 -> 32.32
;
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;
;       Use:    CALL    FXD3232S
;
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point remainder in REMB0, REMB1, REMB2, REMB3
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     27+573+5  = 605 clks            A > 0, B > 0
;                       34+573+23 = 630 clks            A > 0, B < 0
;                       34+573+23 = 630 clks            A < 0, B > 0
;                       41+573+5  = 619 clks            A < 0, B < 0
;                                   12 clks             A = 0
;
;       Min Timing:     27+536+5  = 568 clks            A > 0, B > 0
;                       34+536+23 = 593 clks            A > 0, B < 0
;                       31+536+23 = 593 clks            A < 0, B > 0
;                       41+536+5  = 582 clks            A < 0, B < 0
;
```

```
;       PM: 41+753+22+54 = 870            DM: 14
;
FXD3232S        CLRF            SIGN,F
                CLRF            REMB0,F             ; clear partial remainder
                CLRF            REMB1,F
                CLRF            REMB2,F
                CLRF            REMB3,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W            ; clear exception flag

                BTFSS           BARGB0,MSB          ; if MSB set, negate BARG
                GOTO            CA3232S

                COMF            BARGB3, F
                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB3, F
                ADDWFC          BARGB2, F
                ADDWFC          BARGB1, F
                ADDWFC          BARGB0, F

CA3232S         BTFSS           AARGB0,MSB          ; if MSB set, negate AARG
                GOTO            C3232SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

C3232SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C3232SX1

C3232S          SDIV3232

                BTFSC           TEMPB3,LSB          ; test exception flag
                GOTO            C3232SX4

C3232SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
```

```
                ADDWFC          AARGB0, F

                COMF            REMB3, F
                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB3, F
                ADDWFC          REMB2, F
                ADDWFC          REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00

C3232SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C3232SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C3232SX2
                MOVPF           AARGB0,REMB0        ; quotient = 0, remainder = AARG
                MOVPF           AARGB1,REMB1
                MOVPF           AARGB2,REMB2
                MOVPF           AARGB3,REMB3
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                CLRF            AARGB2,F
                CLRF            AARGB3,F
                GOTO            C3232SOK
C3232SX2        CLRF            AARGB0,F            ; quotient = 1, remainder = 0
                CLRF            AARGB1,F
                CLRF            AARGB2,F
                CLRF            AARGB3,F
                INCF            AARGB3,F
                RETLW           0x00

C3232SX3        COMF            AARGB0,F            ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3232S

C3232SX4        INCF            REMB3,F             ; increment remainder and test for
                CLRF            WREG,F              ; overflow
                ADDWFC          REMB2,F
                ADDWFC          REMB1,F
                ADDWFC          REMB0,F
                MOVFP           BARGB3,WREG
                CPFSEQ          REMB3
                GOTO            C3232SOK
                MOVFP           BARGB2,WREG
                CPFSEQ          REMB2
                GOTO            C3232SOK
                MOVFP           BARGB1,WREG
                CPFSEQ          REMB1
                GOTO            C3232SOK
                MOVFP           BARGB0,WREG
                CPFSEQ          REMB0
                GOTO            C3232SOK
                CLRF            REMB0,F             ; if remainder overflow, clear
                CLRF            REMB1,F             ; remainder, increment quotient and
                CLRF            REMB2,F
                CLRF            REMB3,W
                INCF            AARGB3,F            ; test for overflow exception
                ADDWFC          AARGB2,F
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F
                BTFSS           AARGB0,MSB
```

```
                GOTO                C3232SOK
                BSF                 FPFLAGS,NAN
                RETLW               0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       32/32 Bit Unsigned Fixed Point Divide 32/32 -> 32.32
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;
;       Use:    CALL    FXD3232U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1AARGB2,AARGB3
;               32 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     4+677+2 = 683 clks
;
;       Min Timing:     4+639+2 = 645 clks
;
;       PM: 4+925+1 = 930               DM: 13
;
FXD3232U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F
                CLRF            REMB3, F

                NDIV3232

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       32/31 Bit Unsigned Fixed Point Divide 32/31 -> 32.31
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;
;       Use:    CALL    FXD3231U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     4+582+2 = 588 clks
;
;       Min Timing:     4+544+2 = 550 clks
;
;       PM: 4+768+1 = 773               DM: 12
;
FXD3231U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F
                CLRF            REMB3, F

                UDIV3231

                RETLW           0x00
```

```
;*****************************************************************************************
;*****************************************************************************************
;
;       31/31 Bit Unsigned Fixed Point Divide 31/31 -> 31.31
;
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;
;       Use:    CALL    FXD3131U
;
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     4+573+2 = 579 clks
;
;       Min Timing:     4+536+2 = 542 clks
;
;       PM: 4+753+1 = 758               DM: 12
;
FXD3131U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F
                CLRF            REMB3, F

                UDIV3131

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       32/24 Bit Signed Fixed Point Divide 32/24 -> 32.24
;
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD3224S
;
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     25+476+5  = 506 clks            A > 0, B > 0
;                       30+476+21 = 527 clks            A > 0, B < 0
;                       32+476+21 = 529 clks            A < 0, B > 0
;                       37+476+5  = 518 clks            A < 0, B < 0
;                                   11 clks             A = 0
;
;       Min Timing:     25+441+3  = 469 clks            A > 0, B > 0
;                       30+441+19 = 490 clks            A > 0, B < 0
;                       32+441+19 = 492 clks            A < 0, B > 0
;                       37+441+3  = 481 clks            A < 0, B < 0
;
;       PM: 37+596+20+51 = 704            DM: 12
;
FXD3224S        CLRF            SIGN,F
                CLRF            REMB0,F               ; clear partial remainder
                CLRF            REMB1,F
                CLRF            REMB2,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                IORWF           AARGB2,W
```

```
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W                ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3224S

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
                ADDWFC          BARGB1, F
                ADDWFC          BARGB0, F

CA3224S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3224SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

C3224SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C3224SX1

C3224S          SDIV3224

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C3224SX4

C3224SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB2, F
                ADDWFC          REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00
```

```
C3224SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C3224SX3
                BTFSC           AARGB0,MSB            ; test AARG exception
                GOTO            C3224SX2
                MOVPF           AARGB1,REMB0
                MOVPF           AARGB2,REMB1
                MOVPF           AARGB3,REMB2
                BCF             REMB0,MSB
                RLCF            AARGB1,F
                RLCF            AARGB0,F
                MOVFP           AARGB0,AARGB3
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                CLRF            AARGB2,F
                GOTO            C3224SOK
C3224SX2        CLRF            AARGB3,F              ; quotient = 1, remainder = 0
                INCF            AARGB3,F
                CLRF            AARGB2,F
                CLRF            AARGB1,F
                CLRF            AARGB0,F
                RETLW           0x00

C3224SX3        COMF            AARGB0,F              ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3224S

C3224SX4        INCF            REMB2,F               ; increment remainder and test for
                CLRF            WREG,F
                ADDWFC          REMB1,F
                ADDWFC          REMB0,F
                MOVFP           BARGB2,WREG           ; overflow
                CPFSEQ          REMB2
                GOTO            C3224SOK
                MOVFP           BARGB1,WREG
                CPFSEQ          REMB1
                GOTO            C3224SOK
                MOVFP           BARGB0,WREG
                CPFSEQ          REMB0
                GOTO            C3224SOK
                CLRF            REMB0,F               ; if remainder overflow, clear
                CLRF            REMB1,F
                CLRF            REMB2,W
                INCF            AARGB3,F               ; remainder, increment quotient and
                ADDWFC          AARGB2,F
                ADDWFC          AARGB1,F              ; test for overflow exception
                ADDWFC          AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C3224SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       32/24 Bit Unsigned Fixed Point Divide 32/24 -> 32.24
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2, AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD3224U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2, AARGB3
```

```
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+579+2 = 584 clks
;
;       Min Timing:     3+543+2 = 548 clks
;
;       PM: 3+765+1 = 769               DM: 11
;
FXD3224U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                NDIV3224

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       32/23 Bit Unsigned Fixed Point Divide 32/23 -> 32.23
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2, AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD3223U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2, AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+484+2 = 489 clks
;
;       Min Timing:     3+448+2 = 453 clks
;
;       PM: 3+608+1 = 612               DM: 10
;
FXD3223U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                UDIV3223

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       31/23 Bit Unsigned Fixed Point Divide 31/23 -> 31.23
;
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2, AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD3123U
;
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2, AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+476+2 = 481 clks
;
;       Min Timing:     3+441+2 = 446 clks
```

```
;
;       PM: 3+596+1 = 600               DM: 10
;
FXD3123U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                UDIV3123

                RETLW           0x00

;*****************************************************************************************************
;*****************************************************************************************************
```

## G.2 PIC17CXXX Fixed Point Divide Routines B

```
;        RCS Header $Id: fxdb.a17 2.4 1997/03/22 03:11:13 F.J.Testa Exp $

;        $Revision: 2.4 $

;       PIC17 FIXED POINT DIVIDE ROUTINES B
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD2416S    328       24 bit/16 bit -> 24.16 signed fixed point divide
;
;       FXD2416U    365       24 bit/16 bit -> 24.16 unsigned fixed point divide
;
;       FXD2415U    294       24 bit/15 bit -> 24.15 unsigned fixed point divide
;
;       FXD2315U    287       23 bit/15 bit -> 23.15 unsigned fixed point divide
;
;
;       FXD1616S    227       16 bit/16 bit -> 16.16 signed fixed point divide
;
;       FXD1616U    244       16 bit/16 bit -> 16.16 unsigned fixed point divide
;
;       FXD1615U    197       16 bit/15 bit -> 16.15 unsigned fixed point divide
;
;       FXD1515U    191       15 bit/15 bit -> 15.15 unsigned fixed point divide
;
;
;       FXD1608S    159       16 bit/08 bit -> 16.08 signed fixed point divide
;
;       FXD1608U    196       16 bit/08 bit -> 16.08 unsigned fixed point divide
;
;       FXD1607U    130       16 bit/07 bit -> 16.07 unsigned fixed point divide
;
;       FXD1507U    125       15 bit/07 bit -> 15.07 unsigned fixed point divide
;
;
;       FXD0808S     88       08 bit/08 bit -> 08.08 signed fixed point divide
;
;       FXD0808U     75       08 bit/08 bit -> 08.08 unsigned fixed point divide
;
;       FXD0807U     66       08 bit/07 bit -> 08.07 unsigned fixed point divide
;
;       FXD0707U     61       07 bit/07 bit -> 07.07 unsigned fixed point divide
;
;*****************************************************************************************
;*****************************************************************************************
;
;       24/16 Bit Division Macros
;
SDIV2416        macro
;
;       Max Timing:     5+8+22*12+6 = 283 clks
;
;       Min Timing:     5+8+22*11+3 = 258 clks
```

```
;
;       PM: 5+8+22*14+6 = 327           DM: 8
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK46#v(i)

SADD46#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK46#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD468
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK468

SADD468         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK468          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'
```

```
                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK46#v(i)

SADD46#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK46#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD4616
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK4616

SADD4616        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK4616         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK46#v(i)

SADD46#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK46#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            SOK46
```

```
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
SOK46

                endm


UDIV2416  macro
;
;       restore = 15/20 clks,  nonrestore = 11/14 clks
;
;       Max Timing: 16*15+1+8*20 = 401 clks
;
;       Min Timing: 16*11+1+8*14 = 289 clks
;
;       PM:  16*15+1+8*20 = 401         DM: 8
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK46#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                variable i = D'8'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK46#v(i)      RLCF            AARGB1, F
```

```
                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D'16'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK46#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw


                endm

NDIV2416        macro
;
;       Max Timing:     10+23*15+6 = 361 clks
;
;       Min Timing: 10+23*14+3 = 335 clks
;
;       PM: 10+23*19+6 = 450            DM: 8
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
```

```
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK46#v(i)

NADD46#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK46#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD468
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK468

NADD468         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK468          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK46#v(i)

NADD46#v(i)     ADDWF           REMB1, F
```

```
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK46#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD4616
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK4616

NADD4616        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK4616         RLCF            AARGB2, F

                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK46#v(i)

NADD46#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK46#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            NOK46
                MOVFP           BARGB1,WREG
```

```
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK46
                endm


UDIV2415        macro
;
;       Max Timing:     8+23*12+6 = 290 clks
;
;       Min Timing:     8+23*11+3 = 264 clks
;
;       PM: 8+23*14+6 = 336                     DM: 8
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK45#v(i)

UADD45#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK45#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD458
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK458

UADD458         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
```

```
UOK458          RLCF            AARGB1, F

                variable i = D’9’

                while i < D’16’

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK45#v(i)

UADD45#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK45#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD4516
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK4516

UADD4516        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK4516         RLCF            AARGB2, F


                variable i = D’17’

                while i < D’24’

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK45#v(i)

UADD45#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK45#v(i)      RLCF            AARGB2, F
```

```
                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            UOK45
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK45

                endm


UDIV2315        macro
;
;       Max Timing:     5+8+22*12+6 = 283 clks
;
;       Min Timing:     5+8+22*11+3 = 258 clks
;
;       PM: 5+8+22*14+6 = 327           DM: 8
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK35#v(i)

UADD35#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK35#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw
```

```
                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD358
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK358

UADD358         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK358          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK35#v(i)

UADD35#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK35#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD3516
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK3516

UADD3516        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK3516         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
```

```
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK35#v(i)

UADD35#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK35#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw


                BTFSC           AARGB2,LSB
                GOTO            UOK35
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK35

                endm

;*****************************************************************************************************
;*****************************************************************************************************
;
;       16/16 Bit Division Macros
;
SDIV1616        macro
;
;       Max Timing:     5+8+14*12+6 = 187 clks
;
;       Min Timing:     5+8+14*11+6 = 173 clks
;
;       PM: 5+8+14*14+6 = 215           DM: 6
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
```

```
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            SADD66#v(i)

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK66#v(i)

SADD66#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK66#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            SADD668

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK668

SADD668         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK668          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB1,LSB
                GOTO            SADD66#v(i)

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK66#v(i)

SADD66#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK66#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw
```

```
                BTFSC           AARGB1,LSB
                GOTO            SOK66
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
SOK66

                endm

UDIV1616  macro
;
;       restore = 15 clks,  nonrestore = 11 clks
;
;       Max Timing: 8*15+8*15 = 240 clks
;
;       Min Timing: 8*11+8*11 = 176 clks
;
;       PM: 8*15+8*15 = 240             DM: 6
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F

                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C
UOK66#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                variable i = D'8'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F

                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
```

```
                BCF             _C
UOK66#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                endm

NDIV1616        macro
;
;       Max Timing:     9+15*15+6 = 240 clks
;
;       Min Timing:     9+15*14+6 = 225 clks
;
;       PM: 9+15*19+6 = 300             DM: 7
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB1, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD66#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK66#v(i)

NADD66#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK66#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD668
```

```
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK668

NADD668         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK668          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD66#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK66#v(i)

NADD66#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK66#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                BTFSC           AARGB1,LSB
                GOTO            NOK66
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK66
                endm

UDIV1615        macro
;
;       Max Timing:     7+15*12+6 = 193 clks
;
;       Min Timing:     7+15*11+6 = 178 clks
;
;       PM: 7+15*14+6 = 213             DM: 6
;
                variable i

                RLCF            AARGB0,W
```

```
                RLCF            REMB1, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            UADD65#v(i)

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK65#v(i)

UADD65#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK65#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            UADD658

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK658

UADD658         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK658          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB1,LSB
                GOTO            UADD65#v(i)

                SUBWF           REMB1, F
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK65#v(i)

UADD65#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK65#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                BTFSC           AARGB1,LSB
                GOTO            UOK65
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK65

                endm


UDIV1515        macro
;
;       Max Timing:     5+8+14*12+6 = 187 clks
;
;       Min Timing:     5+8+14*11+6 = 173 clks
;
;       PM: 5+8+14*14+6 = 215           DM: 6
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            UADD55#v(i)

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK55#v(i)
```

```
UADD55#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK55#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB0,LSB
                GOTO            UADD558

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK558

UADD558         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK558          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG

                BTFSS           AARGB1,LSB
                GOTO            UADD55#v(i)

                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK55#v(i)

UADD55#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK55#v(i)      RLCF            AARGB1, F


                variable i = i + 1

                endw

                BTFSC           AARGB1,LSB
                GOTO            UOK55
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK55
```

```
                endm

;_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
;
;       Extra 16 Bit Divide Macros
;
DIV1616         macro
;
;       Timing: restore = 16 clks,  nonrestore = 13 clks        16*16 = 256 clks
;
                variable i

                variable i = D'0'

                while i < D'16'

                RLCF            AARGB1, F
                RLCF            AARGB0, F
                RLCF            REMB1, F
                RLCF            REMB0, F

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F

                BTFSS           _C
                GOTO            RS1616_#v( i )

                BSF             AARGB1,LSB
                GOTO            OK1616_#v( i )

RS1616_#v( i )  MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             AARGB1,LSB

OK1616_#v(i)

                variable i = i + 1

                endw

                endm

DIVMAC          macro
;
;       Timing: restore = 19 clks,  nonrestore = 14 clks        16*19 = 304 clks
;
                variable i

                variable i = D'0'

                while i < D'16'

                RLCF            AARGB1, F
                RLCF            AARGB0, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            notz#v( i )
                MOVFP           BARGB1,WREG
                SUBWF           REMB1,W
```

```
notz#v( i )     BTFSS           _C
                GOTO            nosub#v( i )

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BSF             AARGB1,LSB
                GOTO            ok#v(i)

nosub#v(i)      BCF             AARGB1,LSB

ok#v(i)

                variable i = i + 1

                endw

                endm


;*****************************************************************************************
;*****************************************************************************************
;
;       16/08 Bit Division Macros
;
SDIV1608        macro
;
;       Max Timing:     3+5+14*8+2 = 122 clks
;
;       Min Timing:     3+5+14*8+2 = 122 clks
;
;       PM: 3+5+14*8+2 = 122            DM: 4
;
                variable i

                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
```

```
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLCF            AARGB1, F

                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1608  macro
;
;       restore = 9/15 clks,  nonrestore = 8/11 clks
;
;       Max Timing: 8*9+1+8*15 = 193 clks       max
;
;       Min Timing: 8*8+1+8*11 = 153 clks       min
;
;       PM: 8*9+1+8*15 = 193            DM: 4
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK68#v(i)
                ADDWF           REMB0, F
                BCF             _C
UOK68#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D'8'
```

```
                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F

                BTFSC           _C
                GOTO            UOK68#v(i)
                MOVFP           BARGB0,WREG
                ADDWF           REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C
UOK68#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                endm

NDIV1608        macro
;
;       Max Timing:     7+15*12+3 = 190 clks
;
;       Min Timing: 7+15*11+3 = 175 clks
;
;       PM: 7+15*14+3 = 220             DM: 5
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB0,WREG

                BTFSS           AARGB0,LSB
                GOTO            NADD68#v(i)
                SUBWF           REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK68#v(i)

NADD68#v(i)     ADDWF           REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK68#v(i)      RLCF            AARGB0, F
```

```
                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB0,WREG

                BTFSS           AARGB0,LSB
                GOTO            NADD688
                SUBWF           REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK688

NADD688         ADDWF           REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK688          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB0,WREG

                BTFSS           AARGB1,LSB
                GOTO            NADD68#v(i)
                SUBWF           REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK68#v(i)

NADD68#v(i)     ADDWF           REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK68#v(i)      RLCF            AARGB1, F


                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                MOVFP           BARGB0,WREG
                ADDWF           REMB0, F

                endm


UDIV1607        macro
;
;       Max Timing:     5+15*8+2 = 127 clks
;
;       Min Timing:     5+15*8+2 = 127 clks
;
;       PM: 5+15*8+2 = 127              DM: 4
;
                variable i
```

```
                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB1,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F
                RLCF            AARGB1, F

                variable i = i + 1

                endw

                BTFSS           AARGB1,LSB
                ADDWF           REMB0, F

                endm

UDIV1507        macro
;
;       Max Timing:     3+5+14*8+2 = 122 clks
;
;       Min Timing:     3+5+14*8+2 = 122 clks
;
;       PM: 3+5+14*8+2 = 122            DM: 4
;
                variable i
```

```
        MOVFP           BARGB0,WREG
        SUBWF           REMB0, F
        RLCF            AARGB0, F

        RLCF            AARGB0,W
        RLCF            REMB0, F
        MOVFP           BARGB0,WREG
        ADDWF           REMB0, F
        RLCF            AARGB0, F

        variable i = D'2'

        while i < D'8'

        RLCF            AARGB0,W
        RLCF            REMB0, F
        MOVFP           BARGB0,WREG

        BTFSC           AARGB0,LSB
        SUBWF           REMB0, F
        BTFSS           AARGB0,LSB
        ADDWF           REMB0, F
        RLCF            AARGB0, F

        variable i = i + 1

        endw

        RLCF            AARGB1,W
        RLCF            REMB0, F
        MOVFP           BARGB0,WREG

        BTFSC           AARGB0,LSB
        SUBWF           REMB0, F
        BTFSS           AARGB0,LSB
        ADDWF           REMB0, F
        RLCF            AARGB1, F

        variable i = D'9'

        while i < D'16'

        RLCF            AARGB1,W
        RLCF            REMB0, F
        MOVFP           BARGB0,WREG

        BTFSC           AARGB1,LSB
        SUBWF           REMB0, F
        BTFSS           AARGB1,LSB
        ADDWF           REMB0, F
        RLCF            AARGB1, F

        variable i = i + 1

        endw

        BTFSS           AARGB1,LSB
        ADDWF           REMB0, F

        endm

;*****************************************************************************************
```

```
;*****************************************************************************************
;
;       08/08 Bit Division Macros
;
SDIV0808        macro
;
;       Max Timing:     3+5+6*8+2 = 58 clks
;
;       Min Timing:     3+5+6*8+2 = 58 clks
;
;       PM: 3+5+6*8+2 = 58              DM: 3
;
                variable i

                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = i + 1

                endw

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm


UDIV0808  macro
;
;       restore = 9 clks,  nonrestore = 8 clks
;
;       Max Timing: 8*9 = 72 clks       max
;
;       Min Timing: 8*8 = 64 clks       min
;
;       PM: 8*9 = 72            DM: 3
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
```

```
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F

                BTFSC           _C
                GOTO            UOK88#v(i)
                ADDWF           REMB0, F
                BCF             _C
UOK88#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                endm


UDIV0807        macro
;
;       Max Timing:     5+7*8+2 = 63 clks
;
;       Min Timing:     5+7*8+2 = 63 clks
;
;       PM: 5+7*8+2 = 63                DM: 3
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = i + 1

                endw

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm


UDIV0707        macro
;
;       Max Timing:     3+5+6*8+2 = 58 clks
;
;       Min Timing:     3+5+6*8+2 = 58 clks
;
;       PM: 3+5+6*8+2 = 58              DM: 3
;
                variable i
```

```
                MOVFP           BARGB0,WREG
                SUBWF           REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB0, F
                MOVFP           BARGB0,WREG

                BTFSC           AARGB0,LSB
                SUBWF           REMB0, F
                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F
                RLCF            AARGB0, F

                variable i = i + 1

                endw

                BTFSS           AARGB0,LSB
                ADDWF           REMB0, F

                endm

;*****************************************************************************************
;*****************************************************************************************
;
;       24/16 Bit Signed Fixed Point Divide 24/16 -> 24.16
;
;       Input:  24 bit fixed point dividend in AARGB0, AARGB1, AARGB2
;               16 bit fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD2416S
;
;       Output: 24 bit fixed point quotient in AARGB0, AARGB1, AARGB2
;               16 bit fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     23+283+5 = 311 clks             A > 0, B > 0
;                       26+283+17 = 326 clks            A > 0, B < 0
;                       28+283+17 = 328 clks            A < 0, B > 0
;                       31+283+5 = 319 clks             A < 0, B < 0
;                                   9 clks              A = 0
;
;       Min Timing:     23+258+5 = 286 clks             A > 0, B > 0
;                       26+258+17 = 301 clks            A > 0, B < 0
;                       28+258+17 = 303 clks            A < 0, B > 0
;                       31+258+5 = 294 clks             A < 0, B < 0
;
;       PM: 30+327+16+41 = 414             DM: 9
;
FXD2416S        CLRF            SIGN,F
                CLRF            REMB0,F                   ; clear partial remainder
                CLRF            REMB1,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
```

```
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W            ; clear exception flag

                BTFSS           BARGB0,MSB          ; if MSB set go & negate BARG
                GOTO            CA2416S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                ADDWFC          BARGB0, F

CA2416S         BTFSS           AARGB0,MSB          ; if MSB set go & negate AARGa
                GOTO            C2416SX

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

C2416SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C2416SX1

C2416S          SDIV2416

                BTFSC           TEMPB3,LSB          ; test exception flag
                GOTO            C2416SX4

C2416SOK        BTFSS           SIGN,MSB            ; negate
                RETLW           0x00

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00

C2416SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C2416SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C2416SX2
                MOVPF           AARGB1,REMB0
                MOVPF           AARGB2,REMB1
                BCF             REMB0,MSB
                RLCF            AARGB1,F
                RLCF            AARGB0,F
```

```
                MOVFP           AARGB0,AARGB2
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                GOTO            C2416SOK
C2416SX2        CLRF            AARGB2,F            ; quotient = 1, remainder = 0
                INCF            AARGB2,F
                CLRF            AARGB1,F
                CLRF            AARGB0,F
                RETLW           0x00

C2416SX3        COMF            AARGB0,F            ; numerator = 0x7FFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                INCF            TEMPB3,F
                GOTO            C2416S

C2416SX4        INCF            REMB1,F             ; increment remainder and test for
                CLRF            WREG,F
                ADDWFC          REMB0,F
                MOVFP           BARGB1,WREG         ; overflow
                CPFSEQ          REMB1
                GOTO            C2416SOK
                MOVFP           BARGB0,WREG         ; overflow
                CPFSEQ          REMB0
                GOTO            C2416SOK
                CLRF            REMB0,W             ; if remainder overflow, clear
                CLRF            REMB1,W
                INCF            AARGB2,F            ; remainder, increment quotient and
                ADDWFC          AARGB1,F            ; test for overflow exception
                ADDWFC          AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C2416SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       24/16 Bit Unsigned Fixed Point Divide 24/16 -> 24.16
;
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD2416U
;
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+361+2 = 365 clks
;
;       Min Timing:     2+335+2 = 339 clks
;
;       PM: 2+450+1 = 453               DM: 8
;
FXD2416U        CLRF            REMB0, F
                CLRF            REMB1, F

                NDIV2416

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

```
;
;       24/15 Bit Unsigned Fixed Point Divide 24/15 -> 24.15
;
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD2415U
;
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+290+2 = 294 clks
;
;       Min Timing:     2+264+2 = 268 clks
;
;       PM: 2+336+1 = 339               DM: 8
;
FXD2415U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV2415

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       23/15 Bit Unsigned Fixed Point Divide 23/15 -> 23.15
;
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD2315U
;
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+283+2 = 287 clks
;
;       Min Timing:     2+258+2 = 262 clks
;
;       PM: 2+327+1 = 330               DM: 8
;
FXD2315U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV2315

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16/16 Bit Signed Fixed Point Divide 16/16 -> 16.16
;
;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               16 bit fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD1616S
;
;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               16 bit fixed point remainder in REMB0, REMB1
```

```
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     22+187+5  = 214 clks            A > 0, B > 0
;                       25+187+15 = 227 clks            A > 0, B < 0
;                       25+187+15 = 227 clks            A < 0, B > 0
;                       28+187+5  = 220 clks            A < 0, B < 0
;                                   8 clks              A = 0
;
;       Min Timing:     22+173+5  = 200 clks            A > 0, B > 0
;                       25+173+15 = 213 clks            A > 0, B < 0
;                       25+173+15 = 213 clks            A < 0, B > 0
;                       28+173+5  = 206 clks            A < 0, B < 0
;
;       PM: 27+215+14+34 = 290            DM: 8
;
FXD1616S        CLRF            SIGN,F
                CLRF            REMB0,F             ; clear partial remainder
                CLRF            REMB1,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W           ; clear exception flag

                BTFSS           BARGB0,MSB         ; if MSB set go & negate BARG
                GOTO            CA1616S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                ADDWFC          BARGB0, F

CA1616S         BTFSS           AARGB0,MSB         ; if MSB set go & negate AARGa
                GOTO            C1616SX

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                ADDWFC          AARGB0, F

C1616SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C1616SX1

C1616S          SDIV1616

                BTFSC           TEMPB3,LSB           ; test exception flag
                GOTO            C1616SX4

C1616SOK        BTFSS           SIGN,MSB             ; negate
                RETLW           0x00

                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB1, F
                ADDWFC          AARGB0, F
```

```
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00

C1616SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C1616SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C1616SX2
                MOVPF           AARGB0,REMB0        ; quotient = 0, remainder = AARG
                MOVPF           AARGB1,REMB1
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                GOTO            C1616SOK
C1616SX2        CLRF            AARGB0,F          ; quotient = 1, remainder = 0
                CLRF            AARGB1,F
                INCF            AARGB1,F
                RETLW           0x00

C1616SX3        COMF            AARGB0,F          ; numerator = 0x7FFF + 1
                COMF            AARGB1,F
                INCF            TEMPB3,F
                GOTO            C1616S

C1616SX4        INCF            REMB1,F           ; increment remainder and test for
                CLRF            WREG,F            ; overflow
                ADDWFC          REMB0,F
                MOVFP           BARGB1,WREG
                CPFSEQ          REMB1
                GOTO            C1616SOK
                MOVFP           BARGB0,WREG
                CPFSEQ          REMB0
                GOTO            C1616SOK
                CLRF            REMB0,F           ; if remainder overflow, clear
                CLRF            REMB1,W           ; remainder, increment quotient and
                INCF            AARGB1,F          ; test for overflow exception
                ADDWFC          AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C1616SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF



;*****************************************************************************************
;*****************************************************************************************
;
;       16/16 Bit Unsigned Fixed Point Divide 16/16 -> 16.16
;
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD1616U
;
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+240+2 = 244 clks
;
;       Min Timing:     2+176+2 = 180 clks
;
```

```
;       PM: 2+240+1 = 243               DM: 6
;
FXD1616U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV1616

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16/15 Bit Unsigned Fixed Point Divide 16/15 -> 16.15
;
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD1615U
;
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+193+2 = 197 clks
;
;       Min Timing:     2+178+2 = 182 clks
;
;       PM: 2+213+1 = 216               DM: 6
;
FXD1615U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV1615

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       15/15 Bit Unsigned Fixed Point Divide 15/15 -> 15.15
;
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD1515U
;
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+187+2 = 191 clks
;
;       Min Timing:     2+173+2 = 177 clks
;
;       PM: 2+215+1 = 218               DM: 6
;
FXD1515U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV1515

                RETLW           0x00

;*****************************************************************************************
```

```
;*****************************************************************************************
;
;       16/8 Bit Signed Fixed Point Divide 16/08 -> 16.08
;
;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               8 bit fixed point divisor in BARGB0
;
;       Use:    CALL    FXD1608S
;
;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               8 bit fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     21+122+5 = 148 clks             A > 0, B > 0
;                       22+122+13 = 157 clks            A > 0, B < 0
;                       24+122+13 = 159 clks            A < 0, B > 0
;                       25+122+5 = 152 clks             A < 0, B < 0
;                                  7 clks               A = 0
;
;       Min Timing:     21+122+5 = 148 clks             A > 0, B > 0
;                       22+122+13 = 157 clks            A > 0, B < 0
;                       24+122+13 = 159 clks            A < 0, B > 0
;                       25+122+5 = 152 clks             A < 0, B < 0
;
;       PM: 25+122+12+30 = 189            DM: 6
;
FXD1608S        CLRF            SIGN,F
                CLRF            REMB0,F             ; clear partial remainder
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W            ; clear exception flag

                BTFSS           BARGB0,MSB          ; if MSB set go & negate BARG
                GOTO            CA1608S

                COMF            BARGB0, F
                INCF            BARGB0, F

CA1608S         BTFSS           AARGB0,MSB          ; if MSB set go & negate AARGa
                GOTO            C1608SX

                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB1, F
                ADDWFC          AARGB0, F

C1608SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C1608SX1

C1608S          SDIV1608

                BTFSC           TEMPB3,LSB          ; test exception flag
                GOTO            C1608SX4

C1608SOK        BTFSS           SIGN,MSB            ; negate
```

```
                RETLW           0x00

                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB1, F
                ADDWFC          AARGB0, F

                COMF            REMB0, F
                INCF            REMB0, F

                RETLW           0x00

C1608SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C1608SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C1608SX2
                MOVPF           AARGB1,REMB0
                BCF             REMB0,MSB
                RLCF            AARGB1,F
                RLCF            AARGB0,F
                MOVFP           AARGB0,AARGB1
                CLRF            AARGB0,F
                GOTO            C1608SOK
C1608SX2        CLRF            AARGB1,F            ; quotient = 1, remainder = 0
                INCF            AARGB1,F
                CLRF            AARGB0,F
                RETLW           0x00

C1608SX3        COMF            AARGB0,F            ; numerator = 0x7FFF + 1
                COMF            AARGB1,F
                INCF            TEMPB3,F
                GOTO            C1608S

C1608SX4        INCF            REMB0,F             ; increment remainder and test for
                MOVFP           BARGB0,WREG         ; overflow
                CPFSEQ          REMB0
                GOTO            C1608SOK
                CLRF            REMB0,W             ; if remainder overflow, clear
                INCF            AARGB1,F            ; remainder, increment quotient and
                ADDWFC          AARGB0,F            ; test for overflow exception
                BTFSS           AARGB0,MSB
                GOTO            C1608SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       16/8 Bit Unsigned Fixed Point Divide 16/08 -> 16.08
;
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               8 bit unsigned fixed point divisor in BARGB0
;
;       Use:    CALL    FXD1608U
;
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               8 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+193+2 = 196 clks
;
;       Min Timing:     1+153+2 = 156 clks
;
```

```
;       PM: 1+193+1 = 195               DM: 4
;
FXD1608U        CLRF            REMB0, F

                UDIV1608

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       16/7 Bit Unsigned Fixed Point Divide 16/07 -> 16.07
;
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0
;
;       Use:    CALL    FXD1607U
;
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+127+2 = 130 clks
;
;       Min Timing:     1+127+2 = 130 clks
;
;       PM: 1+127+1 = 129               DM: 4
;
FXD1607U        CLRF            REMB0, F

                UDIV1607

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;       15/7 Bit Unsigned Fixed Point Divide 15/07 -> 15.07
;
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0
;
;       Use:    CALL    FXD1507U
;
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+122+2 = 125 clks
;
;       Min Timing:     1+122+2 = 125 clks
;
;       PM: 1+122+1 = 124               DM: 4
;
FXD1507U        CLRF            REMB0, F

                UDIV1507

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       8/8 Bit Signed Fixed Point Divide 08/08 -> 08.08
;
```

```
;       Input:  8 bit fixed point dividend in AARGB0
;               8 bit fixed point divisor in BARGB0
;
;       Use:    CALL    FXD0808S
;
;       Output: 8 bit fixed point quotient in AARGB0
;               8 bit fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     19+58+5 = 82 clks              A > 0, B > 0
;                       20+58+10 = 88 clks             A > 0, B < 0
;                       20+58+10 = 88 clks             A < 0, B > 0
;                       21+58+5 = 84 clks              A < 0, B < 0
;                                  6 clks              A = 0
;
;       Min Timing:     19+58+5 = 82 clks              A > 0, B > 0
;                       20+58+10 = 88 clks             A > 0, B < 0
;                       20+58+10 = 88 clks             A < 0, B > 0
;                       21+58+5 = 84 clks              A < 0, B < 0
;
;       PM: 20+58+9+23 = 110               DM: 5
;
FXD0808S        CLRF            SIGN,F
                CLRF            REMB0,F             ; clear partial remainder
                MOVPF           AARGB0,WREG
                BTFSC           _Z
                RETLW           0x00

                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W            ; clear exception flag

                BTFSS           BARGB0,MSB
                GOTO            CA0808S

                COMF            BARGB0, F
                INCF            BARGB0, F

CA0808S         BTFSS           AARGB0,MSB
                GOTO            C0808SX

                COMF            AARGB0, F
                INCF            AARGB0, F

C0808SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C0808SX1

C0808S          SDIV0808

                BTFSC           TEMPB3,LSB          ; test exception flag
                GOTO            C0808SX4

C0808SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB0, F
                INCF            AARGB0, F

                COMF            REMB0, F
                INCF            REMB0, F
```

```
                RETLW           0x00

C0808SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C0808SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C0808SX2
                MOVPF           AARGB0,REMB0        ; quotient = 0, remainder = AARG
                CLRF            AARGB0,F
                GOTO            C0808SOK
C0808SX2        CLRF            AARGB0,F            ; quotient = 1, remainder = 0
                INCF            AARGB0,F
                RETLW           0x00

C0808SX3        COMF            AARGB0,F            ; numerator = 0x7F + 1
                INCF            TEMPB3,F
                GOTO            C0808S

C0808SX4        INCF            REMB0,F             ; increment remainder and test for
                MOVFP           BARGB0,WREG         ; overflow
                CPFSEQ          REMB0
                GOTO            C0808SOK
                CLRF            REMB0,F             ; if remainder overflow, clear
                INCF            AARGB0,F            ; remainder, increment quotient and
                BTFSS           AARGB0,MSB          ; test for overflow exception
                GOTO            C0808SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF



;*****************************************************************************************
;*****************************************************************************************
;
;       8/8 Bit Unsigned Fixed Point Divide 08/08 -> 08.08
;
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               8 bit unsigned fixed point divisor in BARGB0
;
;       Use:    CALL    FXD0808U
;
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               8 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+72+2 = 75 clks
;
;       Min Timing:     1+64+2 = 67 clks
;
;       PM: 1+72+1 = 74         DM: 3
;
FXD0808U        CLRF            REMB0, F

                UDIV0808

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       8/7 Bit Unsigned Fixed Point Divide 08/07 -> 08.07
;
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;
```

```
;       Use:    CALL    FXD0807U
;
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+63+2 = 66 clks
;
;       Min Timing:     1+63+2 = 66 clks
;
;       PM: 1+63+1 = 65         DM: 3
;
FXD0807U        CLRF            REMB0, F

                UDIV0807

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       7/7 Bit Unsigned Fixed Point Divide 07/07 -> 07.07
;
;       Input:  7 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;
;       Use:    CALL    FXD0707U
;
;       Output: 7 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     1+58+2 = 61 clks
;
;       Min Timing:     1+58+2 = 61 clks
;
;       PM: 1+58+1 = 60         DM: 3
;
FXD0707U        CLRF            REMB0, F

                UDIV0707

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
```

## G.3 PIC17CXXX Fixed Point Divide Routines C

```
;       RCS Header $Id: fxdc.a17 2.4 1997/03/22 03:11:13 F.J.Testa Exp $

;       $Revision: 2.4 $

;       PIC17 FIXED POINT DIVIDE ROUTINES C
;
;       Input:  fixed point arguments in AARG and BARG
;
;       Output: quotient AARG/BARG followed by remainder in REM
;
;       All timings are worst case cycle counts
;
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;
;       Routine     Clocks    Function
;
;       FXD3216S    429       32 bit/16 bit -> 32.16 signed fixed point divide
;
;       FXD3216U    485       32 bit/16 bit -> 32.16 unsigned fixed point divide
;
;       FXD3215U    390       32 bit/15 bit -> 32.15 unsigned fixed point divide
;
;       FXD3115U    383       31 bit/15 bit -> 31.15 unsigned fixed point divide
;
;
;       FXD2424S    404       24 bit/24 bit -> 24.24 signed fixed point divide
;
;       FXD2424U    440       24 bit/24 bit -> 24.24 unsigned fixed point divide
;
;       FXD2423U    369       24 bit/23 bit -> 24.23 unsigned fixed point divide
;
;       FXD2323U    361       23 bit/23 bit -> 23.23 unsigned fixed point divide
;


;*************************************************************************************************
;*************************************************************************************************
;
;       32/16 Bit Division Macros
;
SDIV3216        macro
;
;       Max Timing:     5+8+30*12+6 = 379 clks
;
;       Min Timing:     5+8+30*11+6 = 349 clks
;
;       PM: 5+8+30*14+6 = 439           DM: 8
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
```

```
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK26#v(i)

SADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK26#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD268
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK268

SADD268         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK268          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK26#v(i)

SADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK26#v(i)      RLCF            AARGB1, F
```

```
                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD2616
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK2616

SADD2616        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2616         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK26#v(i)

SADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK26#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD2624
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK2624

SADD2624        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK2624         RLCF            AARGB3, F

                variable i = D'25'
```

```
                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB3,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK26#v(i)

SADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK26#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            SOK26
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
SOK26

                endm

UDIV3216  macro
;
;       restore = 15/20 clks,  nonrestore = 11/14 clks
;
;       Max Timing: 16*15+1+16*20 = 561 clks
;
;       Min Timing: 16*11+1+16*14 = 401 clks
;
;       PM:  16*15+1+16*20 = 561               DM: 9
;
                variable         i

                variable i = D'0'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK26#v(i)      RLCF            AARGB0, F
```

```
                variable i = i + 1

                endw

                variable i = D’8’

                while i < D’16’

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK26#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D’16’

                while i < D’24’

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK26#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                variable i = D’24’

                while i < D’32’
```

```
                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK26#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                endm

NDIV3216        macro
;
;       Max Timing:     10+31*15+6 = 481 clks
;
;       Min Timing: 10+31*14+6 = 450 clks
;
;       PM: 10+31*19+6 = 605            DM: 9
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK26#v(i)
```

```
NADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK26#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD268
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK268

NADD268         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK268          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK26#v(i)

NADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK26#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
```

```
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD2616
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2616

NADD2616        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK2616         RLCF            AARGB2, F

                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK26#v(i)

NADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK26#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD2624
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK2624

NADD2624        ADDWF           REMB1, F
```

```
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK2624         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB3,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK26#v(i)

NADD26#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK26#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            NOK26
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK26
                endm

UDIV3215        macro
;
;       Max Timing:     8+31*12+6 = 386 clks
;
;       Min Timing:     8+31*11+6 = 355 clks
;
;       PM: 8+31*14+6 = 448             DM: 8
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F
```

```
                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK25#v(i)

UADD25#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK25#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD258
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK258

UADD258         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK258          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK25#v(i)

UADD25#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK25#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw
```

```
                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD2516
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK2516

UADD2516        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2516         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK25#v(i)

UADD25#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK25#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD2524
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK2524

UADD2524        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK2524         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
```

```
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK25#v(i)

UADD25#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK25#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK25
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK25

                endm


UDIV3115        macro
;
;       Max Timing:     5+8+30*12+6 = 379 clks
;
;       Min Timing:     5+8+30*11+6 = 349 clks
;
;       PM: 5+8+30*14+6 = 439           DM: 8
;
                variable i

                MOVFP           BARGB1,WREG
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD15#v(i)
```

```
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK15#v(i)

UADD15#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK15#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD158
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK158

UADD158         ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK158          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK15#v(i)

UADD15#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK15#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD1516
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
```

```
                GOTO            UOK1516

UADD1516        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1516         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK15#v(i)

UADD15#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK15#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD1524
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK1524

UADD1524        ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK1524         RLCF            AARGB3, F

                variable i = D'25'

                while i < D'32'

                RLCF            AARGB3,W
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB1,WREG
                BTFSS           AARGB3,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK15#v(i)
```

```
UADD15#v(i)     ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK15#v(i)      RLCF            AARGB3, F

                variable i = i + 1

                endw

                BTFSC           AARGB3,LSB
                GOTO            UOK15
                MOVFP           BARGB1,WREG
                ADDWF           REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK15

                endm

;*****************************************************************************************
;*****************************************************************************************
;
;       24/24 Bit Division Macros
;
SDIV2424        macro
;
;       Max Timing:     7+11+22*15+8 = 356 clks
;
;       Min Timing:     7+11+22*14+3 = 329 clks
;
;       PM: 7+11+22*19+8 = 444          DM: 9
;
                variable i

                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD44#v(i)
```

```
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK44#v(i)

SADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK44#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            SADD448
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK448

SADD448         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK448          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD44#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK44#v(i)

SADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK44#v(i)      RLCF            AARGB1, F
```

```
                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            SADD4416
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK4416

SADD4416        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK4416         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            SADD44#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            SOK44#v(i)

SADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

SOK44#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            SOK44
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
```

```
SOK44

                endm


UDIV2424  macro
;
;       restore = 20/25 clks,  nonrestore = 14/17 clks
;
;       Max Timing: 16*20+1+8*25 = 521 clks
;
;       Min Timing: 16*14+1+8*17 = 361 clks
;
;       PM:  16*20+1+8*25 = 521         DM: 10
;
                variable      i

                variable i = 0

                while i < 8

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK44#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                variable i = D'8'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                BTFSC           _C
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
```

```
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                BCF             _C

UOK44#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                CLRF            TEMP, F

                variable i = D'16'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                BTFSC           _C
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F
                BCF             _C

UOK44#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw


                endm

NDIV2424        macro
;
;       Max Timing:     13+23*18+8 = 435 clks
;
;       Min Timing: 13+23*17+3 = 407 clks
;
;       PM: 13+23*24+8 = 573            DM: 10
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
```

```
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            TEMP,W
                SUBWFB          TEMP, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK44#v(i)

NADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK44#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            NADD448
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK448

NADD448         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
```

```
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK448          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK44#v(i)

NADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK44#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            NADD4416
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK4416

NADD4416        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK4416         RLCF            AARGB2, F
```

```
                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                RLCF            TEMP, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                CLRF            WREG, F
                SUBWFB          TEMP, F
                GOTO            NOK44#v(i)

NADD44#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                CLRF            WREG, F
                ADDWFC          TEMP, F

NOK44#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            NOK44
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

NOK44
                endm


UDIV2423        macro
;
;       Max Timing:     11+23*15+8 = 364 clks
;
;       Min Timing:     11+23*14+3 = 336 clks
;
;       PM: 11+23*19+8 = 456                    DM: 9
;
                variable i

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                variable i = D'1'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK43#v(i)

UADD43#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK43#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD438
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK438

UADD438         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK438          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
```

```
                BTFSS           AARGB1,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK43#v(i)

UADD43#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK43#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD4316
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK4316

UADD4316        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK4316         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK43#v(i)

UADD43#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
```

```
                ADDWFC          REMB0, F

UOK43#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw

                BTFSC           AARGB2,LSB
                GOTO            UOK43
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK43

                endm


UDIV2323        macro
;
;       Max Timing:     7+11+22*15+8 = 356 clks
;
;       Min Timing:     7+11+22*14+3 = 329 clks
;
;       PM: 7+11+22*19+8 = 444          DM: 9
;
                variable i

                MOVFP           BARGB2,WREG
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                RLCF            AARGB0, F

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
                RLCF            AARGB0, F

                variable i = D'2'

                while i < D'8'

                RLCF            AARGB0,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD33#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0, F
                GOTO            UOK33#v(i)

UADD33#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK33#v(i)      RLCF            AARGB0, F

                variable i = i + 1

                endw

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB0,LSB
                GOTO            UADD338
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK338

UADD338         ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK338          RLCF            AARGB1, F

                variable i = D'9'

                while i < D'16'

                RLCF            AARGB1,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD33#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK33#v(i)

UADD33#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK33#v(i)      RLCF            AARGB1, F

                variable i = i + 1

                endw
```

```
                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB1,LSB
                GOTO            UADD3316
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK3316

UADD3316        ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK3316         RLCF            AARGB2, F


                variable i = D'17'

                while i < D'24'

                RLCF            AARGB2,W
                RLCF            REMB2, F
                RLCF            REMB1, F
                RLCF            REMB0, F
                MOVFP           BARGB2,WREG
                BTFSS           AARGB2,LSB
                GOTO            UADD33#v(i)
                SUBWF           REMB2, F
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1, F
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0, F
                GOTO            UOK33#v(i)

UADD33#v(i)     ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F

UOK33#v(i)      RLCF            AARGB2, F

                variable i = i + 1

                endw


                BTFSC           AARGB2,LSB
                GOTO            UOK33
                MOVFP           BARGB2,WREG
                ADDWF           REMB2, F
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1, F
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0, F
UOK33
```

```
                endm

;*****************************************************************************************
;*****************************************************************************************
;
;       32/16 Bit Signed Fixed Point Divide 32/16 -> 32.16
;
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD3216S
;
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     24+379+5 = 408 clks             A > 0, B > 0
;                       27+379+19 = 425 clks            A > 0, B < 0
;                       31+379+19 = 429 clks            A < 0, B > 0
;                       34+379+5 = 418 clks             A < 0, B < 0
;                                  10 clks              A = 0
;
;       Min Timing:     24+349+5 = 378 clks             A > 0, B > 0
;                       27+349+19 = 395 clks            A > 0, B < 0
;                       31+349+19 = 399 clks            A < 0, B > 0
;                       34+349+5 = 388 clks             A < 0, B < 0
;
;       PM: 34+439+18+46 = 537            DM: 10
;
FXD3216S        CLRF            SIGN,F
                CLRF            REMB0,F                 ; clear partial remainder
                CLRF            REMB1,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                IORWF           AARGB3,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W                ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set go & negate BARG
                GOTO            CA3216S

                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB1, F
                ADDWFC          BARGB0, F

CA3216S         BTFSS           AARGB0,MSB              ; if MSB set go & negate AARGa
                GOTO            C3216SX

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F
```

```
C3216SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C3216SX1

C3216S          SDIV3216

                BTFSC           TEMPB3,LSB          ; test exception flag
                GOTO            C3216SX4

C3216SOK        BTFSS           SIGN,MSB            ; negate
                RETLW           0x00

                COMF            AARGB3, F
                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB3, F
                ADDWFC          AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00


C3216SX1        BTFSS           BARGB0,MSB          ; test BARG exception
                GOTO            C3216SX3
                BTFSC           AARGB0,MSB          ; test AARG exception
                GOTO            C3216SX2
                MOVPF           AARGB2,REMB0
                MOVPF           AARGB3,REMB1
                BCF             REMB0,MSB
                RLCF            AARGB2,F
                RLCF            AARGB1,F
                RLCF            AARGB0,F
                MOVFP           AARGB0,AARGB2
                MOVFP           AARGB1,AARGB3
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                GOTO            C3216SOK
C3216SX2        CLRF            AARGB3,F            ; quotient = 1, remainder = 0
                INCF            AARGB3,F
                CLRF            AARGB2,F
                CLRF            AARGB1,F
                CLRF            AARGB0,F
                RETLW           0x00

C3216SX3        COMF            AARGB0,F            ; numerator = 0x7FFFFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                COMF            AARGB3,F
                INCF            TEMPB3,F
                GOTO            C3216S

C3216SX4        INCF            REMB1,F             ; increment remainder and test for
                CLRF            WREG,F
                ADDWFC          REMB0,F
                MOVFP           BARGB1,WREG         ; overflow
                CPFSEQ          REMB1
```

```
                GOTO            C3216SOK
                MOVFP           BARGB0,WREG           ; overflow
                CPFSEQ          REMB0
                GOTO                 C3216SOK
                CLRF                 REMB0,W        ; if remainder overflow, clear
                CLRF                 REMB1,W
                INCF                 AARGB3,F       ; remainder, increment quotient and
                ADDWFC               AARGB2,F
                ADDWFC               AARGB1,F       ; test for overflow exception
                ADDWFC               AARGB0,F
                BTFSS                AARGB0,MSB
                GOTO                 C3216SOK
                BSF                  FPFLAGS,NAN
                RETLW                0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       32/16 Bit Unsigned Fixed Point Divide 32/16 -> 32.16
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD3216U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1AARGB2,AARGB3
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+481+2 = 485 clks
;
;       Min Timing:     2+450+2 = 459 clks
;
;       PM: 2+605+1 = 608               DM: 9
;
FXD3216U        CLRF            REMB0, F
                CLRF            REMB1, F

                NDIV3216

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       32/15 Bit Unsigned Fixed Point Divide 32/15 -> 32.15
;
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD3215U
;
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+386+2 = 390 clks
;
;       Min Timing:     2+355+2 = 359 clks
;
;       PM: 2+448+1 = 451               DM: 8
;
```

```
FXD3215U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV3215

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       31/15 Bit Unsigned Fixed Point Divide 31/15 -> 31.15
;
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;
;       Use:    CALL    FXD3115U
;
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     2+379+2 = 383 clks
;
;       Min Timing:     2+349+2 = 353 clks
;
;       PM: 2+439+1 = 442               DM: 8
;
FXD3115U        CLRF            REMB0, F
                CLRF            REMB1, F

                UDIV3115

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       24/24 Bit Signed Fixed Point Divide 24/24 -> 24.24
;
;       Input:  24 bit signed fixed point dividend in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD2424S
;
;       Output: 24 bit signed fixed point quotient in AARGB0, AARGB1, AARGB2
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     24+356+5  = 385 clks            A > 0, B > 0
;                       29+356+19 = 404 clks            A > 0, B < 0
;                       29+356+19 = 404 clks            A < 0, B > 0
;                       34+356+5  = 395 clks            A < 0, B < 0
;                                   10 clks             A = 0
;
;       Min Timing:     24+329+5  = 358 clks            A > 0, B > 0
;                       29+329+19 = 377 clks            A > 0, B < 0
;                       29+329+19 = 377 clks            A < 0, B > 0
;                       34+329+5  = 368 clks            A < 0, B < 0
;
;       PM: 34+444+18+44 = 540             DM: 11
;
FXD2424S        CLRF            SIGN,F
                CLRF            REMB0,F             ; clear partial remainder
                CLRF            REMB1,F
```

```
                CLRF            REMB2,F
                MOVPF           AARGB0,WREG
                IORWF           AARGB1,W
                IORWF           AARGB2,W
                BTFSC           _Z
                RETLW           0x00

                MOVPF           AARGB0,WREG
                XORWF           BARGB0,W
                BTFSC           WREG,MSB
                COMF            SIGN,F

                CLRF            TEMPB3,W                ; clear exception flag

                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA2424S

                COMF            BARGB2, F
                COMF            BARGB1, F
                COMF            BARGB0, F
                INCF            BARGB2, F
                ADDWFC          BARGB1, F
                ADDWFC          BARGB0, F

CA2424S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C2424SX

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                INCF            AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

C2424SX         MOVPF           AARGB0,WREG
                IORWF           BARGB0,W
                BTFSC           WREG,MSB
                GOTO            C2424SX1

C2424S          SDIV2424

                BTFSC           TEMPB3,LSB              ; test exception flag
                GOTO            C2424SX4

C2424SOK        BTFSS           SIGN,MSB
                RETLW           0x00

                COMF            AARGB2, F
                COMF            AARGB1, F
                COMF            AARGB0, F
                CLRF            WREG, F
                INCF            AARGB2, F
                ADDWFC          AARGB1, F
                ADDWFC          AARGB0, F

                COMF            REMB2, F
                COMF            REMB1, F
                COMF            REMB0, F
                INCF            REMB2, F
                ADDWFC          REMB1, F
                ADDWFC          REMB0, F

                RETLW           0x00

C2424SX1        BTFSS           BARGB0,MSB              ; test BARG exception
                GOTO            C2424SX3
```

```
                BTFSC           AARGB0,MSB              ; test AARG exception
                GOTO            C2424SX2
                MOVPF           AARGB0,REMB0            ; quotient = 0, remainder = AARG
                MOVPF           AARGB1,REMB1
                MOVPF           AARGB2,REMB2
                CLRF            AARGB0,F
                CLRF            AARGB1,F
                CLRF            AARGB2,F
                GOTO            C2424SOK
C2424SX2        CLRF            AARGB0,F                ; quotient = 1, remainder = 0
                CLRF            AARGB1,F
                CLRF            AARGB2,F
                INCF            AARGB2,F
                RETLW           0x00

C2424SX3        COMF            AARGB0,F                ; numerator = 0x7FFFFF + 1
                COMF            AARGB1,F
                COMF            AARGB2,F
                INCF            TEMPB3,F
                GOTO            C2424S

C2424SX4        INCF            REMB2,F                 ; increment remainder and test for
                CLRF            WREG,F                  ; overflow
                ADDWFC          REMB1,F
                ADDWFC          REMB0,F
                MOVFP           BARGB2,WREG
                CPFSEQ          REMB2
                GOTO            C2424SOK
                MOVFP           BARGB1,WREG
                CPFSEQ          REMB1
                GOTO            C2424SOK
                MOVFP           BARGB0,WREG
                CPFSEQ          REMB0
                GOTO            C2424SOK
                CLRF            REMB0,F                 ; if remainder overflow, clear
                CLRF            REMB1,F                 ; remainder, increment quotient and
                CLRF            REMB2,W
                INCF            AARGB2,F                ; test for overflow exception
                ADDWFC          AARGB1,F
                ADDWFC          AARGB0,F
                BTFSS           AARGB0,MSB
                GOTO            C2424SOK
                BSF             FPFLAGS,NAN
                RETLW           0xFF


;*****************************************************************************************
;*****************************************************************************************
;
;       24/24 Bit Unsigned Fixed Point Divide 24/24 -> 24.24
;
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD2424U
;
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+435+2 = 440 clks
;
;       Min Timing:     3+407+2 = 412 clks
;
```

```
;       PM: 3+573+1 = 577               DM: 10
;
FXD2424U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                NDIV2424

                RETLW           0x00


;*****************************************************************************************
;*****************************************************************************************
;
;       24/23 Bit Unsigned Fixed Point Divide 24/23 -> 24.23
;
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD2423U
;
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+364+2 = 369 clks
;
;       Min Timing:     3+336+2 = 341 clks
;
;       PM: 3+456+1 = 460               DM: 9
;
FXD2423U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                UDIV2423

                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
;
;       23/23 Bit Unsigned Fixed Point Divide 23/23 -> 23.23
;
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;
;       Use:    CALL    FXD2323U
;
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;
;       Result: AARG, REM  <--  AARG / BARG
;
;       Max Timing:     3+356+2 = 361 clks
;
;       Min Timing:     3+329+2 = 334 clks
;
;       PM: 3+444+1 = 448               DM: 9
;
FXD2323U        CLRF            REMB0, F
                CLRF            REMB1, F
                CLRF            REMB2, F

                UDIV2323
```

```
                RETLW           0x00

;*****************************************************************************************
;*****************************************************************************************
```

**Note the following details of the code protection feature on PICmicro® MCUs.**

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable”.
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.



DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV — DNV MSC The Netherlands Accredited by the RvA — ISO 9001 / QS-9000 REGISTERED FIRM

*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 certified.*

# WORLDWIDE SALES AND SERVICE

## AMERICAS

**Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200 Fax: 480-792-7277
Technical Support: 480-792-7627
Web Address: http://www.microchip.com

**Rocky Mountain**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-7456

**Atlanta**
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

**Boston**
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

**Chicago**
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

**Dallas**
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

**Detroit**
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

**Kokomo**
2767 S. Albright Road
Kokomo, Indiana 46902
Tel: 765-864-8360 Fax: 765-864-8387

**Los Angeles**
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

**New York**
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

**San Jose**
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

**Toronto**
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

## ASIA/PACIFIC

**Australia**
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

**China - Beijing**
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

**China - Chengdu**
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-6766200 Fax: 86-28-6766599

**China - Fuzhou**
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

**China - Shanghai**
Microchip Technology Consulting (Shanghai) Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

**China - Shenzhen**
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office
Rm. 1315, 13/F, Shenzhen Kerry Centre,
Renminnan Lu
Shenzhen 518001, China
Tel: 86-755-2350361 Fax: 86-755-2366086

**Hong Kong**
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

**India**
Microchip Technology Inc.
India Liaison Office
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

**Japan**
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

**Korea**
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

**Singapore**
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-334-8870 Fax: 65-334-8850

**Taiwan**
Microchip Technology Taiwan
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

## EUROPE

**Denmark**
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

**France**
Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

**Germany**
Microchip Technology GmbH
Gustav-Heinemann Ring 125
D-81739 Munich, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

**Italy**
Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

**United Kingdom**
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

01/18/02