Technical Note

Flash Corruption with writeUserBlock Function

If a user program uses the library function **writeUserBlock()**, a Dynamic C library error present in versions 7.20-7.25 can cause corruption or incorrect placement of the System ID block and/or User block areas with certain board types that have a version 4 System ID block. These boards were made in production runs from June to late August of 2002. This problem may also occur on customer designed boards with the same Dynamic C and ID block versions.

The System ID block is a reserved area in the top of flash memory that contains important information such as product ID, and on some Ethernet-enabled boards, media access control (MAC) address. The User block is an area of flash reserved for user use and calibration constants. These features are explained in detail in the Rabbit Designer's Handbooks. The information we are most concerned with here is the product ID, the MAC address and calibration constants.

The only difference between a version 4 ID block and earlier versions is that **writeUserBlock()** mirrors the System ID and User blocks in versions of Dynamic C 7.20 and above. This mirroring provides an extra measure of protection with flash writes in case of power failure while writing. **writeUserBlock()** will work correctly with version 4 System ID blocks and versions 7.26 and higher of Dynamic C.

Symptoms

The utility program accompanying this document will test whether a board has a corrupted version 4 System ID block and attempt to recover it. Although the block contains other information, the only two items that can cause compilation or runtime problems if the information is incorrect are the product ID and MAC address.

Product ID

The consequence of not having a correct product ID is that Dynamic C will misidentify the board type and incorrectly set an the internal macro **_BOARD_TYPE_** that is used to conditionally compile code in several libraries. This will usually generate a compiler error when trying to **#use** a board-specific library or compile a board-specific function.

MAC Address

The consequence of not having a valid MAC address is that user programs won't be able to communicate over Ethernet.

Calibration Constants

Incorrect or missing calibration constants in the User block will cause incorrect A/D D/A conversions.

Affected Products

The following boards may experience compiling errors due a corrupted product ID:

- TCP/IP Development Board (MAC address not a problem because it is in EEPROM, not flash)
- OP67x0 (MAC address is in EEPROM also)

The following boards may experience compiling errors due to a corrupted product ID and/or a corrupted MAC address. BL21x0 boards may have corrupted or mislocated calibration constants:

- BL20x0
- BL21x0
- OP68x0
- RCM20x0
- RCM21x0
- RCM30x0
- RCM31x0
- SR91x0 CPU board

Note: x refers to any digit.

Trivial Case: RCM20x0

None of the information in the RCM20x0's System ID block is required for correct operation because there is no MAC address and no board-specific library or code for this product. However, if your compiler version is older than 7.02C and it is not a Premier version of Dynamic C, you will see a compiler error if a correct product ID is not present because the product ID will default to 0x100 (JackRabbit/BL1800 board) and JackRabbit specific libraries will try to **#use** JackRabbit libraries that weren't included in all Dynamic C versions until 7.02. This is easily fixed by commenting out the **#use** statements in **default.h** for the BL1800 libraries. If you upgrade Dynamic C, this will not be an issue.

Resolution

Libraries correcting the error in **writeUserBlock()** were introduced and released in Dynamic C version 7.26. Libraries correcting the **writeUserBlock()** error for versions 7.20, 7.21, and 7.25 accompany this document in a zip file. You should replace the libraries in your Dynamic C version with the like-named libraries for that version. If you have already run a program that uses **writeUserBlock()** with version 7.20, 7.21, or 7.25, and are experiencing any of the described symptoms, you will need to run the diagnostic repair utility and/or write ID block utilities described below to recover or replace the system ID block information and calibration constants if applicable. If your board is BL2xxx type with calibration constants and the diagnostic/repair utility reports that it was unable to find or recover valid calibration constants, you will have to recalibrate your analog I/O using sample programs provided or send it back to Z-World for recalibration.

If a product has an intact MAC address, or the MAC address is in EEPROM, then Ethernet will still work correctly. A simple library change will set the **_BOARD_TYPE_** macro correctly if the product ID is corrupted: **#undef _BOARD_TYPE_** and **#define** it to the desired value in **\LIB\DEFAULT.H**. However, this change will have to be propagated if you upgrade Dynamic C. A more permanent fix is to change the system ID block.

Diagnostic/Repair Utility

The diagnostic/repair utility program accompanying this document, **IDBLOCK_RECOVERY.C**, will do the following.

1. Check the installed System ID block version to ensure that it is version 4; if not it will quit without making changes.

2. If BL21xx board, if needed and if possible, recover calibration constants.

3. If a User block exists, and only if necessary, copy the valid User block data to the appropriate location.

4. If a System ID Block exists, and only if necessary, copy the valid version 4 System ID block data to the appropriate version 2 System ID block location.

5. Change the ID block version from 4 to 2.

Directions for running the program are in the comment block at the top of the program. If the program reports an unrecoverable System ID block, follow the instructions in the next section, "Rewriting the System ID Block." If the program reports that it was unable to recover calibration constants on a BL21xx board follow the instructions in the section, "Recalibrating BL21xx Series Boards."

Rewriting the System ID Block

To rewrite the system ID block, you need to run **WRITE_IDBLOCK.C** provided in the zip file accompanying this document. Directions for running the program are in the comment block at the top of the program. It is important to enter the correct product ID and MAC address if used. This program will create a version 2 System ID block.

Recalibrating BL21xx Series Boards

If the **IDBLOCK_RECOVERY.C** program reports that it cannot find or recover calibrations constants for a BL21xx board, you must recalibrate using

    \SAMPLES\BL2100\ADC\AD_CALIB.C

and

    \SAMPLES\BL2100\DAC\DACAL.C

or return the board to Z-World for recalibration.