	Rabbit 4/5000 con Dynamic C 10.54 o superior	Nota de Aplicación
		CoAN-012
	Actualización Remota de Firmware	Publicado: 00/00/0000
		Página 1 de 7

Revisión	Fecha	Comentario	Autor
0	19/08/2009		Ulises Bigliati

ÍNDICE

Aclaraciones previas	1
Sobre esta nota:.....	1
Terminología y abreviaturas:	1
Fuera de alcance	1
Objetivos	1
Introducción	2
Requerimientos de hardware y software	2
El medio de almacenamiento temporal	2
El método de transferencia	3
Mecanismo de actualización	3
Incorporando el upgrade remoto en nuestra aplicación	4
El proyecto propuesto	4
El hardware	4
El software	4
Módulos del proyecto	5
Módulo upld.lib	5
Módulo iweb.lib	5
Módulo CoAN-012.c	5
Prueba del proyecto	6
Consideraciones finales	7

Aclaraciones previas

Sobre esta nota:

El presente trabajo está realizado sobre la base de la nota de aplicación AN 421 (*Application note AN421*) que puede encontrarse en el sitio web del fabricante de los productos a los cuales hace referencia, es decir, Rabbit Semiconductors. A la fecha de la realización puede encontrarse un enlace a la nota original en: <http://www.rabbit.com/support/appNotes.shtml>.

Terminología y abreviaturas:


- Firmware:** denominamos bajo este término al archivo binario compilado con Dynamic C, como así también al software que esté en ejecución en el módulo Rabbit.
- Upgrade Remoto:** Con este término nos referimos al proceso de grabación del firmware en un módulo Rabbit sin la utilización del cable de programación.
- DC:** Utilizamos “DC” como abreviatura de Dynamic C. En particular, en esta nota siempre que se hable de Dynamic C o DC, se estará haciendo referencia a Dynamic C versión 10.54 o superior.

Fuera de alcance

No es objeto de esta nota explicar detalles de la utilización de las funciones de librería, ya que esto puede obtenerse de la documentación de Rabbit (AN421), programas de demostración provistos con DC, etc.

Objetivos

La intención de este trabajo es brindar una aproximación a los diferentes aspectos de implementación de la funcionalidad de actualización remota de firmware.

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
		CoAN-012
	Actualización Remota de Firmware	Publicado: 00/00/0000
		Página 2 de 7

Introducción

En esta ocasión presentamos la implementación de una funcionalidad frecuentemente considerada de utilidad por los desarrolladores de sistemas dedicados.

La actualización remota de firmware otorga ventajas en diferentes aspectos y si bien las herramientas de desarrollo de Rabbit ofrecían desde hace algún tiempo algunos mecanismos (complicados, por cierto) para realizar esta tarea, es recién a partir del Dynamic C 10.54 y de la concepción de una arquitectura de hardware adecuada que esta funcionalidad se encuentra disponible para ser incorporada de forma sencilla mediante una API¹ (application programming interface) en las aplicaciones del usuario.

Requerimientos de hardware y software

Para la implementación del upgrade remoto se requiere Dynamic C 10.54 o superior, y el hardware debe reunir las siguientes condiciones:

- Procesador Rabbit 4000 o superior
- Firmware corriendo desde fast SRAM
- Disponibilidad de almacenamiento masivo: NAND, serial flash, mini SD card.

Los módulos Rabbit que reúnen esas características son los siguientes:

RCM4200, RCM43xx, RCM4400W, RCM54xxW, RCM5600W, BL4S1xx, BL4S200, BL5S220.

El medio de almacenamiento temporal


La utilidad de actualización de firmware proporcionada por DC, posee la flexibilidad de permitir la elección del medio de almacenamiento del nuevo firmware en función del soporte de hardware disponible en el módulo Rabbit, y de las posibilidades del sistema pudiendo optar por:

- Sistema de archivos FAT sobre:
 - Flash de datos Serial
 - Flash de booteo serial
 - NAND Flash
 - Mini SD card
- Almacenamiento directo (sin sistema de archivos) en:
 - Flash de datos serial
 - Flash de booteo serial

A continuación se vinculan en una tabla los módulos Rabbit con los correspondientes medios de almacenamiento que están disponibles en cada caso:

Módulo base	Medios de almacenamiento
RCM4200	FAT, flash de datos serial
RCM4300	FAT (SD card), flash de booteo serial
RCM4400W	FAT, flash de datos serial
RCM5400W	FAT, flash de datos serial
RCM5600W	Almacenamiento directo en flash de booteo serial
BL4S100 Series	FAT, flash de booteo serial
BL4S200 (RCM4310)	FAT, flash de booteo serial
BL5S220 (RCM5400W)	FAT, flash de datos serial

¹ En la nota de aplicación del fabricante AN421, se encuentra la documentación completa de las funciones que integran esta API que es llamada "Remote Program Update Library", y que está compuesta por los módulos board_update.lib, y firmware_info.lib. Ambos módulos pueden hallarse en `\Lib\Rabbit4000\RemoteProgramUpdate\`

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
	Actualización Remota de Firmware	CoAN-012
		Publicado: 00/00/0000
		Página 3 de 7

El método de transferencia

El método de transferencia del nuevo firmware (archivo .bin compilado con DC) puede (y debe) seleccionarse según los requerimientos de la aplicación, siendo posible su implementación mediante:

- HTTP Server (`upload_firmware.c`)
- HTTP Client (`download_firmware.c`)
- FTP Client (`download_firmware.c`)
- TFTP Client (`tftp_get_firmware.c`)

Además de los métodos mencionados, el Dynamic C incluye un programa de demostración (`bootchk.c`) que le permite a la aplicación del usuario verificar la existencia del firmware en el sistema de archivos de una SD card.

Como puede verse, existen varias maneras de realizar las actualizaciones de firmware, los ejemplos provistos por Rabbit solo muestran los usos mas comunes, otros métodos podrían implementarse, según la necesidad del usuario, por ejemplo, a través de un puerto serie.

Mecanismo de actualización

Este mecanismo está basa en que los módulos que lo soportan poseen una arquitectura que incorpora las siguientes partes:


- a) Memoria de datos SRAM**
Datos de run-time y NVRAM.
- b) Memoria RAM de ejecución rápida.**
Contiene el código ejecutable, el programa corre desde allí.
- c) Memoria Flash de programa.**
Almacenamiento del código ejecutable, constantes, objetos web, etc. Puede ser serial, el programa no se ejecuta desde allí.
- d) Memoria de almacenamiento masivo (Serial Flash / SD card / Nand Flash)**
Datalogging, objetos web, etc.

De la enumeración anterior se desprende lo siguiente:

- 1) El procesador ejecuta su programa desde **(b)** y de pronto, el usuario interviene enviando un nuevo firmware al módulo (por el método que sea).
- 2) Entonces el nuevo firmware se recibe y se coloca en **(d)** donde cómodamente se verifica su integridad antes de considerarlo válido.
- 3) A continuación, si el paso anterior fue exitoso, se puede sobrescribir **(c)** con el nuevo firmware recientemente almacenado en **(d)**. Esto puede hacerse tranquilamente(¿?), porque el procesador ejecuta el programa desde **(b)**, y no hay ningún problema para regrabar **(c)**.
- 4) Una vez sobrescrito el firmware original en **(c)** con el recientemente recibido y leído desde **(d)**, se puede reiniciar el sistema ejecutando desde este punto, el nuevo firmware.

El último firmware transmitido puede quedar en **(d)** por si hiciera falta nuevamente, si se quiere, a modo de un discutiblemente útil back-up, ya que es posible reutilizarlo.

En el punto (3) aparecieron unos signos de interrogación al momento de indicar que tranquilamente se podría regrabar la flash. Los cuestionamientos surgen, porque no habría ningún problema con la regrabación, siempre y cuando no suceda ningún imprevisto (léase fallo de alimentación, etc.), si esto sucede justamente en algún instante dentro del proceso de regrabación del firmware, simplemente el sistema no arrancará más, y lamentablemente para volverlo a la vida habrá que reunir al desafortunado módulo con las herramientas de grabación tradicionales (DC, cable, RFU). Es decir, lo que considerabamos una ventaja, se volvería un problema bastante importante.

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
		CoAN-012
	Actualización Remota de Firmware	Publicado: 00/00/0000
		Página 4 de 7

Incorporando el upgrade remoto en nuestra aplicación

Si consideramos los programas de demostración incluidos en el DC junto con la documentación provista en la nota de aplicación AN421, no parece que quedara mucho mas que agregar. Todo está muy bien documentado y ejemplificado y las demos de Rabbit, tal como debe ser, se enfocan en exponer las funcionalidad en sí mismas de manera puntual.

Es en este contexto, es que proponemos generar un módulo de software fácilmente reutilizable por cualquiera de nuestros proyectos, mediante el cual podamos agregarle rápidamente esta funcionalidad al programa de usuario que lo necesite sin impactar en el resto del programa.

Para esto combinamos un par de ejemplos² provistos con DC. Uno de ellos funcionará a modo de programa principal, y el otro será una de las demos que implementan el upgrade remoto, en esta ocasión, utilizando como método de transmisión el *upload HTTP*.

A este último programa lo transformaremos en un módulo de software independiente que podrá ser incorporado al programa principal como una library mas.

El proyecto propuesto

El hardware

El proyecto de software de Dynamic C está basado en un RCM5400W. Este módulo posee una memoria flash serial de 1Mb, donde podremos depositar el firmware transmitido, y además de reunir el resto de los requerimientos descriptos anteriormente, presenta conectividad ethernet inalámbrica. Utilizaremos este módulo conjuntamente con su placa de prototipos correspondiente.

El software

Tal como anticipamos, nuestro software es la fusión de dos programas de demostración que vienen incluidos en el DC. Uno de ellos se denomina "*browseled*" (*browseled.c*) y lo utilizamos como programa principal, es decir, será el programa de usuario que contemplará la posibilidad de ser actualizado remotamente. En este caso, la funcionalidad del programa en sí mismo es poco importante, de todas maneras, comentaremos que lo que hace es simplemente virtualizar los leds DS2, DS3, y los switches S2 y S3 presentes en la placa de prototipos, en una interfaz web. Por otra parte, permite ver el código del programa principal en otra página web.

El otro programa, implementa el mecanismo que queremos probar y es llamado "*upload_firmware*" (*upload_firmware.c*). Este último programa, será convertido en un módulo independiente que al ser incluido en este o en cualquier otro programa, permitirá la actualización en forma remota, en este caso, vía el HTTP upload.

En virtud de lo expuesto, la estructura que adquiere este proyecto en cuanto a archivos se refiere, es la que sigue:

```


myProjects__CoAN-012
|__CoAN-012.c
|__CoAN-012.dcp
|__lib.dir
|__lib
|   |__upld.lib
|   |__iweb.lib
|__iweb
|   |__showsrc.shtml
|   |__browseled.zhtml
|   |__upload.zhtml
|   |__3
|   |__

```

² Los programas seleccionados para ese proyecto y sus correspondientes rutas de ubicación son :

- Samples/RemoteProgramUpdate/**upload_firmware.c**
- Samples/RCM5400W/TCPIP/**browseled.c**

³ El directorio "iweb" también contiene las imágenes (.gif)

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
		CoAN-012
	Actualización Remota de Firmware	Publicado: 00/00/0000
		Página 5 de 7

Nota: Resaltamos en negrita los módulos que son de mayor interés pues se trata de los reutilizables que permiten agregar la funcionalidad a cualquier proyecto que los incorpore.

Módulos del proyecto

Dado que el código fuente del proyecto se encuentra debidamente comentado haremos solamente un breve repaso de los módulos que lo componen.

Módulo upld.lib

Se trata de la pieza de software que implementa el mecanismo de actualización remota, este módulo funciona en conjunto con `upload.zhtml` que es la página web que cumple el rol de interfaz para esta funcionalidad. A esta página web la hemos ubicado dentro del directorio `iweb`, que se encuentra dentro del directorio del proyecto. Cabe mencionar que la ubicación de esta página es referida por la siguiente línea de código que se encuentra en el encabezado del módulo bajo análisis:

```
#ximport "iweb/upload.zhtml"    upload_zhtml
```

Dentro del archivo `upld.lib`, solamente debemos configurar en que bloque de memoria residirá el firmware transmitido descomentando la línea adecuada:

```
// #define BU_TEMP_USE_FAT        // use file on FAT filesystem
// #define BU_TEMP_USE_SBF        // use unused portion of serial boot flash
#define BU_TEMP_USE_SFLASH        // write directly to serial flash
```

Como vemos, nuestra elección fue depositar el firmware directamente en la flash serial de datos, sin utilizar el sistema de archivos. Y esto sería todo lo que habría que hacer aquí.

Módulo iweb.lib

El módulo `iweb.lib` es propio del programa de usuario, es decir, implementa funciones relacionadas con el programa en sí mismo, y en este caso, no es muy relevante lo que suceda aquí adentro. Nos limitaremos a comentar que aquí realizamos la importación de los objetos web de la aplicación, definimos aspectos del control de acceso web, como así también se definen las funciones que controlan los eventos recibidos por la interfaz web de parte del usuario. En definitiva, toda la funcionalidad presentada por el programa de demostración `browseled.c` se encuentra encapsulada en este módulo.

Módulo CoAN-012.c

Finalmente llegamos al programa principal, desde el cual invocamos al resto de los módulos y definimos una serie de parámetros necesarios para que nuestra aplicación funcione como queremos. Entre los mas relevantes figuran los siguientes:

Para que la API del upgrade remoto incluya el código necesario para el método de transferencia que vamos a utilizar:

```
#define USE_HTTP_UPLOAD
```

Invocamos a nuestro módulo "upld.lib"


```
#use "upld.lib"
```

Invocamos la library de Rabbit que implementa la funcionalidad de upgrade remoto:

```
#use "board_update.lib".
```

Invocamos a nuestro módulo "iweb.lib"

```
#use "iweb.lib"
```

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
		CoAN-012
	Actualización Remota de Firmware	Publicado: 00/00/0000
		Página 6 de 7

Adicionalmente podremos ver al principio del programa, los `#define's` correspondientes a la parametrización de la interfaz de red, por supuesto debemos adecuar estos valores a los de nuestra propia red, la descripción de estos aspectos queda fuera del alcance de este trabajo.

Si continuamos viendo el código, llegaremos hasta el `main()` donde simplemente llamamos funciones de inicialización, las siguientes son las que inicializan a los módulos de software descritos más arriba:

```

iweb_init();
upld_setup();

```

Más adelante, se inicializa el servidor web, y a continuación se ingresa en un loop infinito, que simplemente mantiene vivo el servicio web, y se dedica a mantener actualizadas las variables que el programa expone vía web.

Prueba del proyecto

Una vez incorporado el proyecto a su instalación de Dynamic C⁴ se debe realizar una primer grabación del módulo en su memoria flash.

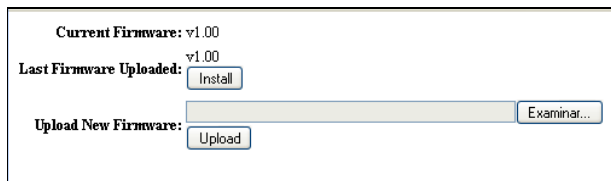
A continuación se puede comprobar el funcionamiento del programa principal, desconectándolo del DC, y una vez que se ha reiniciado, ingresando en un browser la dirección IP del módulo.

Nos pedirá nuestra identificación: (**admin / upload**) y a continuación estaremos frente a la interfaz web del programa. Si presionamos un switch en la placa de prototipos, esto se verá reflejado en la interfaz web y viceversa. Podemos, además, mediante el click en un enlace presente en la página, visualizar el código fuente del programa principal. Pero todo esto no es relevante, ya que lo que nos interesa es la actualización remota, para esto:

Debemos tener preparado algún archivo que contenga el firmware a transferir, para esto, compilamos el proyecto nuevamente⁵, pero no en flash, sino en un archivo, con el siguiente comando del menú del DC: Copile->Compile To .Bin File ->... De esta forma se habrá generado el archivo binario resultante de la compilación del proyecto dentro del directorio de este.

Una vez que ya contamos con el archivo .bin tipeamos en la barra del explorador la dirección IP del módulo seguido de `/fw/upload.zhtml` que es la ruta donde mapeamos nuestra página de upload (<http://10.0.0.55/fw/upload.zhtml>).

Si ya nos habíamos registrado con el equipo ingresaremos directamente, de lo contrario nos pedirá la identificación (**admin / upload**). Habremos ingresado entonces a la página que nos permitirá actualizar el firmware, y nos mostrará algo como esto:




Como puede verse, se nos informa la versión del firmware actualmente en uso, y la última versión que se ha transferido. Los números de versión pueden ser manipulados desde la API de programación que nos ofrece el DC⁶ sin embargo, esto no lo hemos hecho en esta demo.

⁴ Para esto, simplemente debe crear un subdirectorio llamado `myProjects` dentro del directorio de instalación del DC y dentro de este directorio debe copiar el directorio del proyecto (CoAN-012). A continuación, puede abrir el proyecto desde el menú: `File->Project->Open` y buscar el archivo `CoAN-012.dcp` dentro del directorio del proyecto.

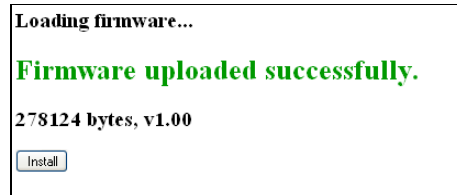
⁵ Si bien, la propia interfaz web de actualización de firmware nos puede mostrar información sobre la versión, podría ser conveniente introducir alguna modificación en el programa para notar fácilmente la diferencia entre el firmware previo y el actual.

⁶ Para esto, existen las macros: `_FIRMWARE_NAME_`, `_FIRMWARE_VERSION_`, `_FIRMWARE_TIMESTAMP_` en la documentación (AN421 o manual de referencia de funciones del DC).

	Rabbit 4/5000 + Dynamic C 10.54 o superior	Nota de Aplicación
	Actualización Remota de Firmware	CoAN-012
		Publicado: 00/00/0000
		Página 7 de 7

Como puede verse, tendremos la oportunidad de instalar el último firmware transferido (que ha quedado almacenado en flash), o de transferir uno nuevo.

Si optamos por enviarle un nuevo firmware al módulo, al culminar la transferencia y si es que esta tuvo éxito, veremos una imagen como la que sigue en la ventana de nuestro explorador web:



Y si además optamos por instalar el firmware en ese mismo momento, perderemos en principio la comunicación con el módulo, la cual podrá ser restablecida una vez que el equipo termine de regrabar su memoria flash, y a continuación se termine de reinicializar.

Cabe mencionar que la comunicación browser-módulo, luego del upgrade, no será restablecida automáticamente en esta demo, para lograr esto, sería necesario forzar al explorador web a que consulte periódicamente la dirección IP del módulo hasta que este vuelva a responder.

Este procedimiento sería tarea exclusiva del cliente web (nuestro browser) y por lo tanto debe programarse dentro de la página de upload, incorporando rutinas en Javascript y probablemente utilizando AJAX para una mayor eficiencia.

Consideraciones finales

La posibilidad de actualizar el firmware en forma remota, es decir, con independencia de las herramientas de desarrollo, mediante protocolos de comunicación estándares como HTTP, FTP, etc., puede ser en la mayoría de los casos, de gran atractivo para un producto final.

A partir de la versión 10.54 del Dynamic C esta funcionalidad puede ser muy fácilmente incorporada a cualquier desarrollo basado en procesadores Rabbit 4000 o superiores que cumplan los requisitos de hardware necesarios.

No obstante, a la hora de evaluar la implementación de este mecanismo, es necesario tomar conciencia de que este esquema no ofrece ninguna garantía de supervivencia ante fallos de alimentación, es decir, que el sistema no cuenta por ejemplo, con un *boot-manager* o *boot-loader* que pueda ofrecer algún procedimiento que garantice que el sistema pueda recuperarse ante una interrupción en el proceso de regrabación del firmware.