

COMENTARIO TÉCNICO

# *Buceando en los MCUs Freescale.....*



Por Ing. Daniel Di Lella  
Dedicated Field Application Engineer  
[www.edudevices.com.ar](http://www.edudevices.com.ar)  
[dilella@arnet.com.ar](mailto:dilella@arnet.com.ar)



[www.edudevices.com.ar](http://www.edudevices.com.ar)

## *“Serie Flexis”.... ..... “Como migrar de 8 a 32 Bits sin traumas”*

**Segunda Entrega.**



¡Hola Amigos! .... En el artículo anterior, habíamos visto las características y las ventajas de la “Serie Flexis” de 8 y 32 Bits para hacernos la vida más fácil cuando nuestro proyecto crece y crece y se nos hace algo difícil saber hasta donde.

También quedó clara la facilidad de migración que permite este serie, evidentemente nacida para ello, con productos pin a pin compatibles entre los mundos de 8 y 32 bits, con periféricos idénticos y herramientas de desarrollo de hardware y software unificadas.

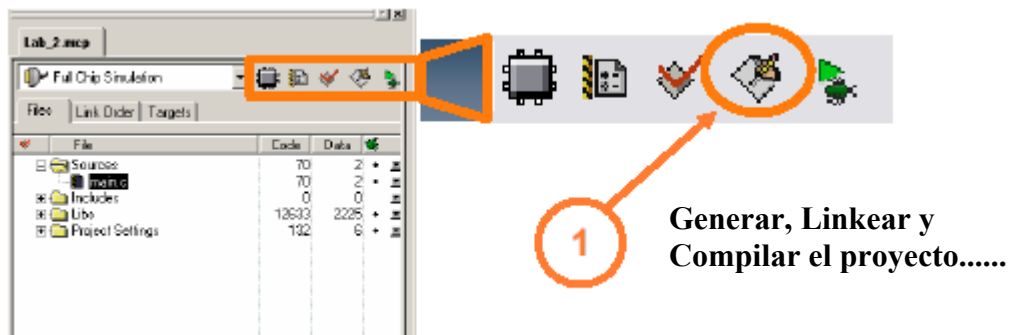
Pero..... Todo ello es cierto si tenemos en cuenta algunos puntos a la hora de implementar nuestro sistema pensando en la migración, ya que de no tenerlos en cuenta, nuestro proyecto puede ser “tortuoso” de migrar del mundo de los 8 bits al de los 32.

### ***Errores comunes cuando migramos desde los 8 Bits de los S08 a los 32 Bits de los ColdFire V1.***

Para lograr la mejor comprensión de los consejos de “portabilidad” y como usarlos, se nos hace necesario tomar un ejemplo específico de migración desde los 8 bits de los S08 a los 32 Bits de los ColdFire V1. Desarrollaremos entonces, un ejemplo basados en el MC9S08QE128 (8 Bits) y el objetivo será migrar el proyecto al MCF51QE128 (32 Bits). Estos dispositivos en particular, son los primeros de la Serie Flexis y nos servirán de referencia para la migración de miembros futuros de dicha familia.

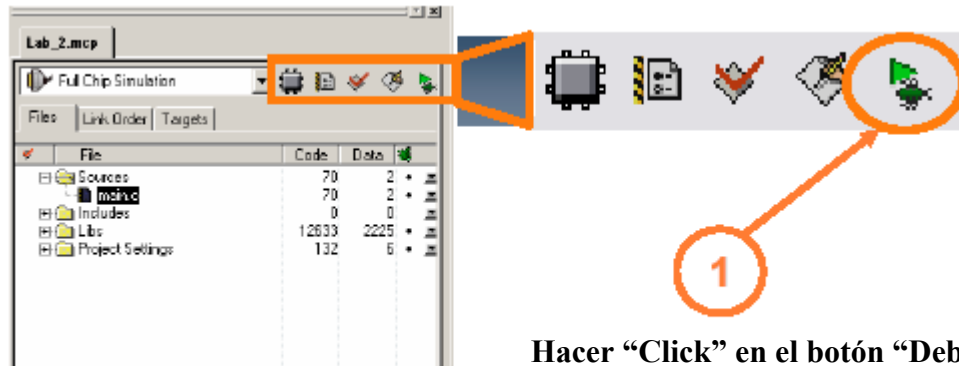
Para trabajar en este ejemplo, el lector puede descargar del sitio web de Freescale Semiconductor ([www.freescale.com](http://www.freescale.com)) el archivo comprimido “AN3465SW.ZIP” que contiene una serie de “Laboratorios” o programas de ejemplo para llevar a cabo junto con el sistema **Codewarrior 6.x** y la herramienta de demostración “**DEMOQE128**”. En este ejemplo concreto, utilizaremos el archivo “**Lab\_2.mcp**” ubicado en la carpeta Controller Continuum\Lab\_2.

Procedamos entonces a generar el proyecto, compilar el código para el mismo con el **MC9S08QE128** como MCU y notaremos que **no se generan errores** según se puede observar en la **Figura 6**.



**Figura 6.- Generación y compilación del proyecto “Lab\_2.mcp”.**

Luego, procederemos a hacer “Click” en el botón “Debug” (flecha Verde), según se muestra en la **Figura 7.-** y así el proyecto estará listo para “correr” con el MCU MC9S08QE128.



Hacer “Click” en el botón “Debug” para comenzar con la emulación del proyecto.

**Figura 7.- Emulación del proyecto.**

El proyecto “Lab\_2.mcp” comprende el uso de los módulos de KBI y de RTI de la siguiente forma:

- El RTC se habilita para hacer “parpadear” el LED1 cada 1 segundo e incrementar una variable global.
- Se configura el módulo de KBI para que cada vez que se presiona el botón pulsador de la placa DEMOQE128 el LED2 invierte su estado.
- El MCU estará en estado STOP hasta que ocurra una interrupción.

Nótese que, cuando trabajamos con el S08QE128, el proyecto compila y corre perfectamente. En este ejemplo tenemos código que está utilizando GPIO, KBI y RTI, mientras que se usa la instrucción STOP para minimizar el consumo de energía. Ahora procederemos a cambiar el “core” de S08 a ColdFire V1, siguiendo los pasos explicados en el artículo anterior y que se detallan a continuación:

1. Hacer “Click” en el ícono “MCU Change Wizard”, según **Figura 8.-**
2. Seleccione el dispositivo MCF51QE128.
3. Seleccione al modo de conexión “P&E Multilink/Cyclone Pro”.
4. Acepte la opción (tildar) de crear un backup del archivo.
5. Hacer “Click” en la opción “Finish” para finalizar la conversión. (**Figura 9.-**).

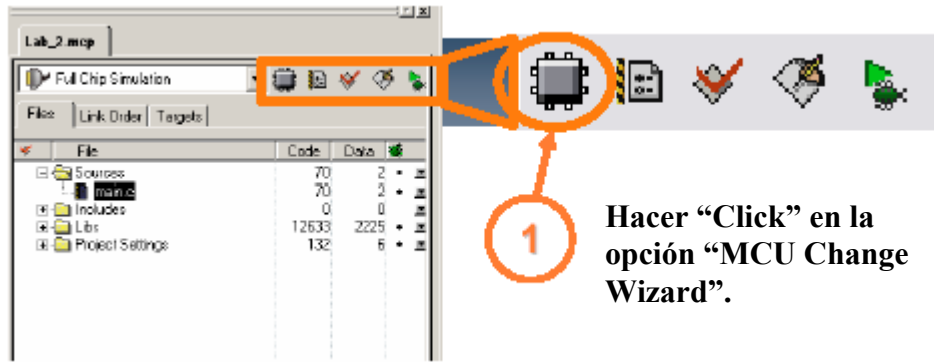


Figura 8.- Migrando de S08 a ColdFire V1 con la función "MCU Change Wizard".

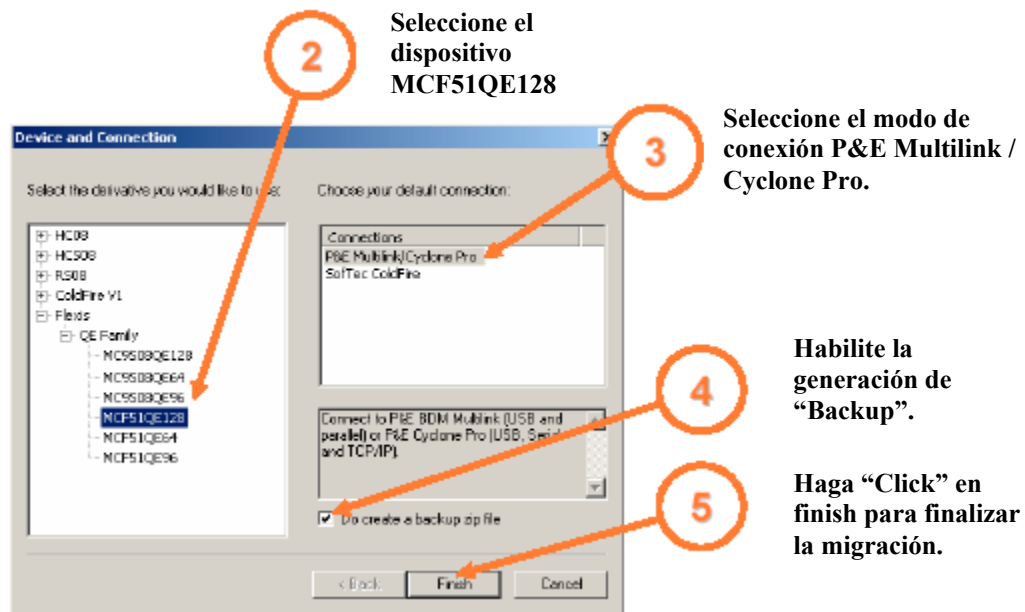


Figura 9.- Otros pasos de la migración.

6. El entorno integrado CodeWarrior 6.x cambiará en forma automática el archivo "derivatives.h" y se muestra ello en la **Figura 10.-**

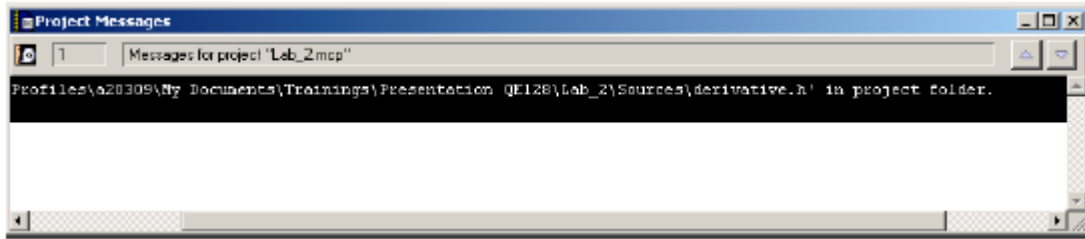


Figura 10.- Mensajes del proyecto.

7. Compilar el Proyecto.
8. El dispositivo ha cambiado y la **Figura 11** muestra los errores de migración.

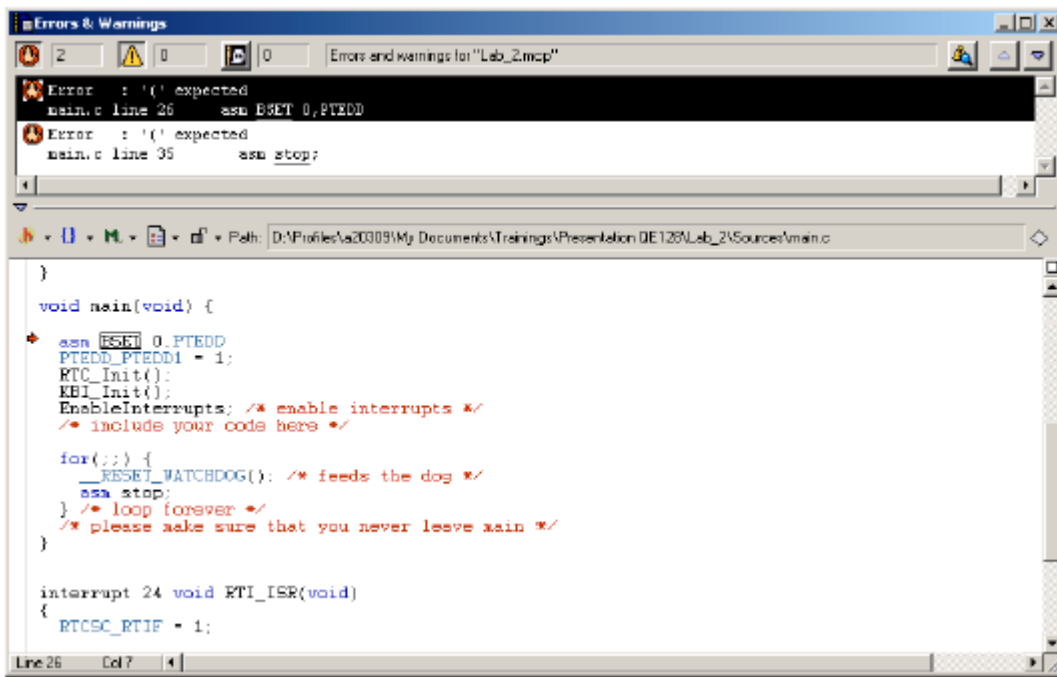


Figura 11.- Errores de Compilación al usar el proyecto con ColdFire V1.

Cuando migramos el software desde los 8 Bits del S08 a los 32 Bits ColdFire V1, las diferencias de las arquitecturas podrían afectar la operación del programa de aplicación. En este ejemplo, las diferencias en la arquitectura lleva al compilador a emitir mensajes de error. Estos errores fueron introducidos intencionalmente para mostrar los puntos a tener en cuenta durante la migración.

En el próximo artículo de “*Buceando....*” veremos uno a uno los errores y como corregirlos, para garantizar que nuestro proyecto tenga una fácil migración de los 8 Bits a los 32 Bits.

*¡Hasta la próxima mis Amigos!*

**Continuará.....**

[www.edudevices.com.ar](http://www.edudevices.com.ar)

