

COMENTARIO TÉCNICO

Buceando en los MCUs Freescale.....



Por Ing. Daniel Di Lella
Dedicated Field Application Engineer

www.edudevices.com.ar

dilella@arnet.com.ar



www.edudevices.com.ar

“El ajuste de fábrica del Oscilador Interno de los HC9S08/V1 ColdFire y el R(S)_POD”

¡¡Hola amigos!! Siguiendo con la serie de artículos técnicos relacionados a la amplia familia de MCUs de Freescale Semiconductor, en esta oportunidad, les traigo la solución a un problema que en más de una ocasión me la han consultado y creo que merece ser aclarada.

Como ustedes saben, las familias *HC9S08*, *Serie Flexis HC9S08 / V1 ColdFire*, poseen múltiples opciones de osciladores, entre ellas, la opción de utilizar el **Oscilador Interno** (ICS – “Internal Clock Source” o ICG – “Internal Clock Generator”) que disponen y que tiene muy buena precisión (+/- 2%) si está correctamente ajustado. Todos los MCUs de las familias anteriormente citadas vienen “ajustados de fábrica” por medio de 2 bytes grabados en posiciones reservadas de la memoria FLASH (NVFTRIM y NVICSTRIM) que permiten ajustar la frecuencia de trabajo del oscilador interno simplemente “copiando” estos valores de fábrica a sendos registros en memoria RAM (ICSSC e ICSTRIM) dentro del código de programa del usuario (en las primeras líneas de este).

Si estos valores no pudieran copiarse a estos registros, el Oscilador Interno funcionaría con una precisión de +/- 25% en todo el rango de temperatura, algo no muy adecuado para aplicaciones que necesiten de buena precisión en los tiempos como pueden ser las comunicaciones Seriales Asíncronas (SCI) o demoras de tiempo críticas.

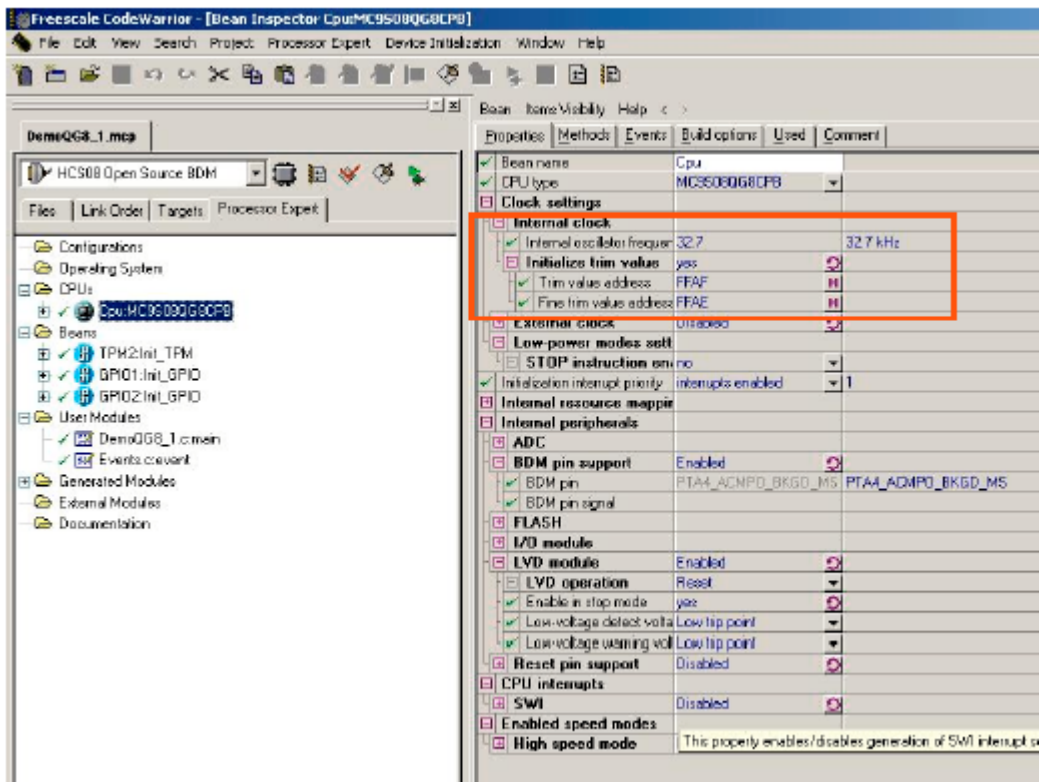
Los usuarios del sistema de desarrollo “*R(S)_POD*” se encuentran con el inconveniente de que la herramienta no posee el ajuste automático del oscilador interno (Auto – Trimming) como si lo hace el sistema USBMULTILINKBDME **pero a expensas de un precio de la herramienta mucho mayor que la de fabricación nacional.**

Para compensar este inconveniente, el usuario puede utilizar los valores contenidos en los registros NVFTRIM y NVICSTRIM ubicados en la zona de memoria FLASH y grabados durante el ajuste automático que se hace de fábrica del oscilador interno de los HC9S08.

Este ajuste de fábrica asegura una precisión en la frecuencia de +/- 2% y ello es más que suficiente para la mayoría de los grabados en FLASH, el usuario deberá evitar el “**borrado en masa**” (**mass erase**) aplicaciones que utilizan este tipo de oscilador. Para conservar los valores de fábrica de la memoria FLASH, por lo que esta serie de indicaciones solo serán válidas para aquellos dispositivos que se encuentren “virgenes” de fábrica, una situación más que común durante la etapa de producción de equipos y sistemas.

Pasos a seguir:

- 1) Cuando se utiliza la aplicación “*Processor Expert*”, configurar el bean “CPU” para habilitar la función “**initialize trim value**” que permite utilizar los valores almacenados en posiciones de flash y copiarlos a los registros de RAM. En nuestro ejemplo, el MC9S08QG8 dispone de dichos valores en las posiciones de flash \$FFAE y \$FFAF.



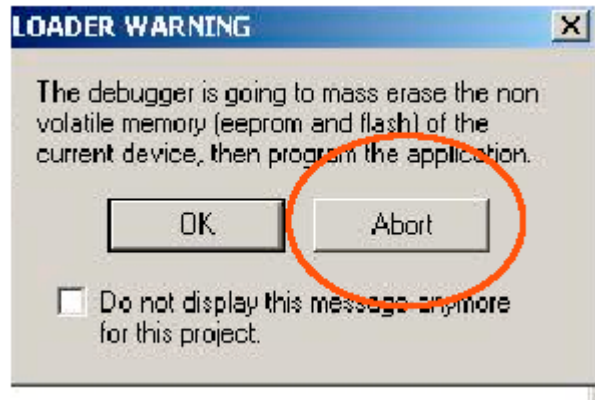
- 2) Una vez generado el código con el “Processor Expert”, en el archivo “MCUint.asm” se generarán en forma automática las líneas de código que permiten copiar los valores almacenados en las posiciones de flash grabadas de fábrica, en los registros en RAM ICSTRIM e ICSSC, como se puede ver en la siguiente figura.....

```

AND    #0CF
STA    SPMSC3
; System clock initialization
; ICSTRM: Initialize internal clock trim from a non volatile memory
LDA    $FFAF
STA    ICSTRM
; ICSSC: Initialize internal clock trim from a non volatile memory
LDA    $FFAE
ORA    #$FE
STA    ICSSC
; ICSC1: CLKS=0,RDIV=0,IREFS=1,IRCLKEN=0,IREFSTEN=0
MOV    #$04,ICSC1 ; Initialization of the ICS control register 1

```

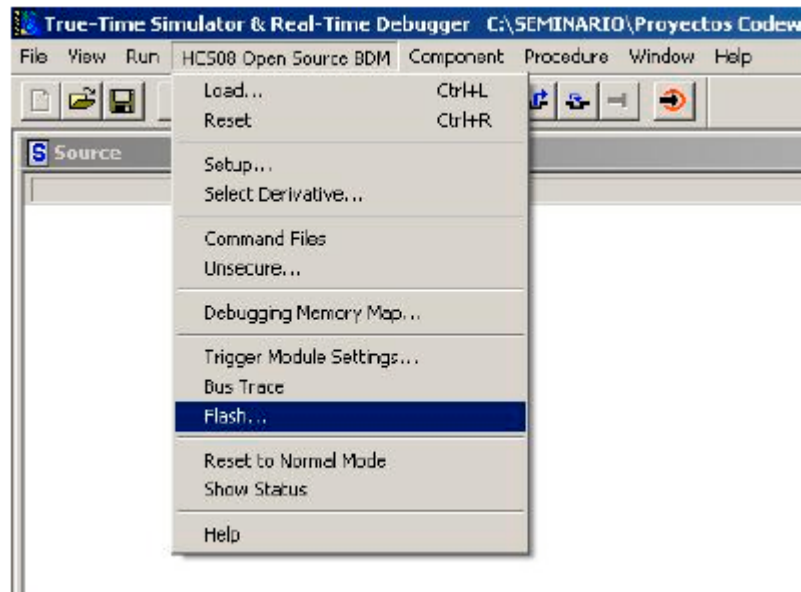
- 3) Una vez compilado el código, se está en condiciones de grabar la memoria flash de programa del MCU y/o proceder a la Emulación en Tiempo Real por medio del “**Debugger**” (**Hiwave**) del entorno CodeWarrior. Como se ha visto anteriormente, ello se consigue al hacer “click” en el ícono con la “**flecha verde**” de la barra de herramienta del editor. Al abrirse el entorno de Debugging, el mismo preguntará por la opción de “**Borrado en masa**” (**mass erase**) de la memoria FLASH, proceder a “**ABORTAR**” el borrado en masa para conservar los valores de fábrica, según se muestra en la figura.



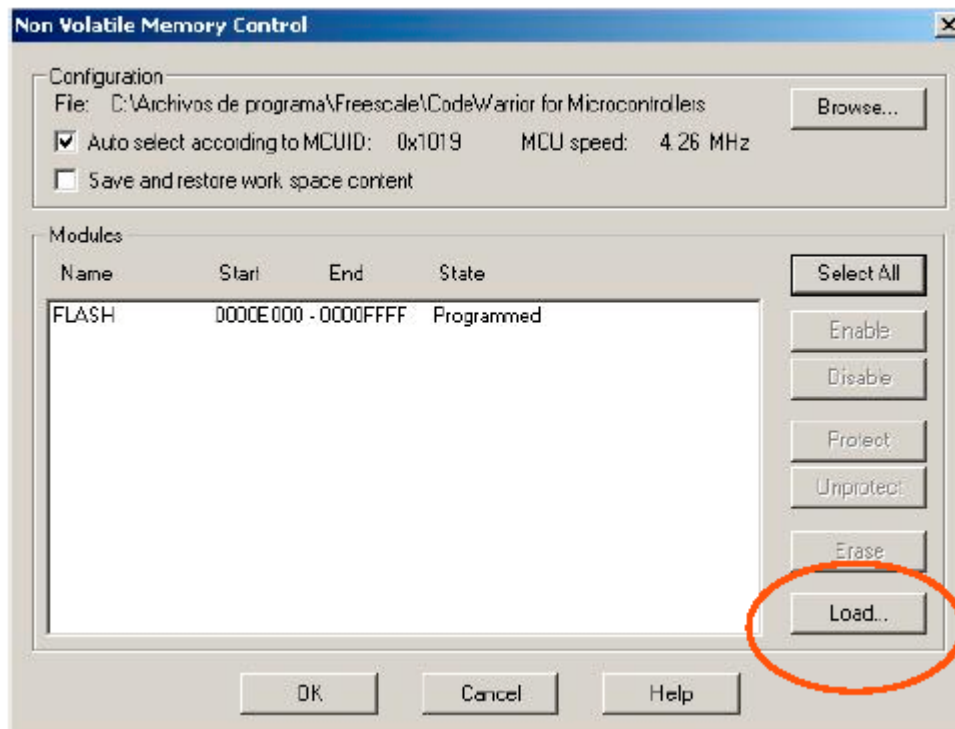
- 4) Como resultado de esta acción el sistema **nos indicará un estado de error de carga del programa en la memoria FLASH del MCU** (el mismo no se ha cargado aún), hacer click en “aceptar” para continuar.....



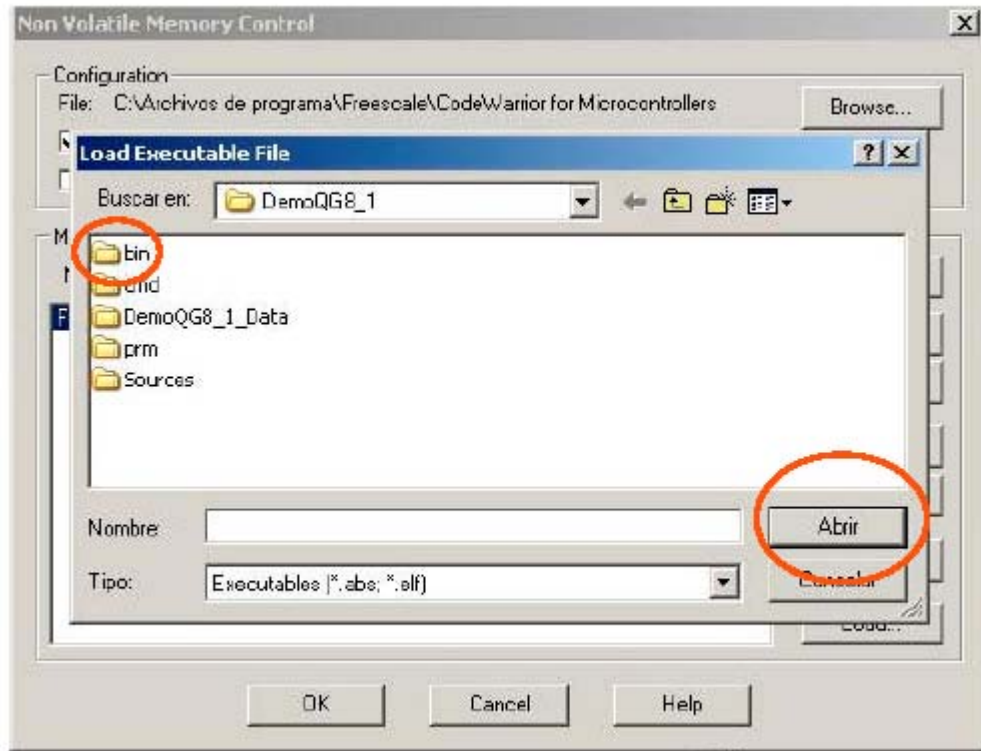
- 5) Hacer Click en la opción “**HCS08 Open Source BDM**” de la barra de herramientas del debugger y elegir la opción “**Flash**” como se ve en la figura.



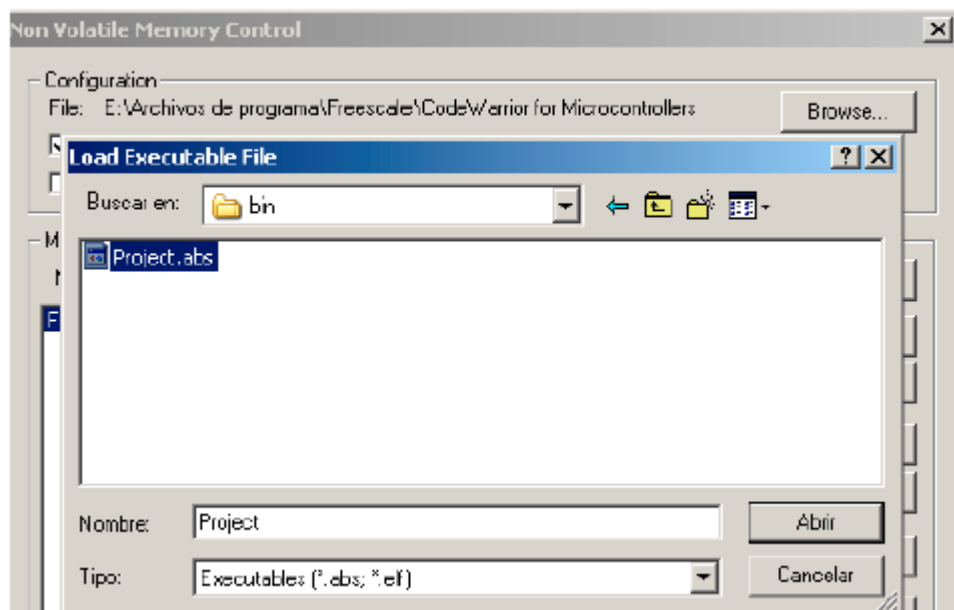
- 6) Una vez dentro de la opción **“Flash”**, hacer **“click”** en el ícono **“Load”** que nos permitirá cargar el programa (en formato **“S19”**) en la memoria FLASH respetando los valores previamente grabados de fábrica.



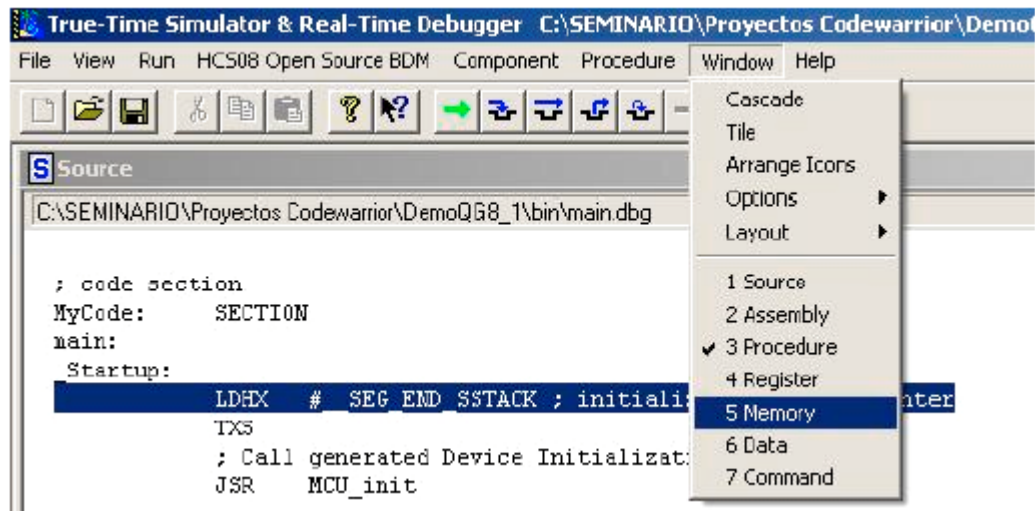
- 7) Al abrirse la ventana **“Load Executable File”**, elegir la carpeta **“Bin”** que contiene el archivo ejecutable a cargar en la memoria flash del MCU y luego hacer **“click”** en **“abrir”** para confirmar la operación.....



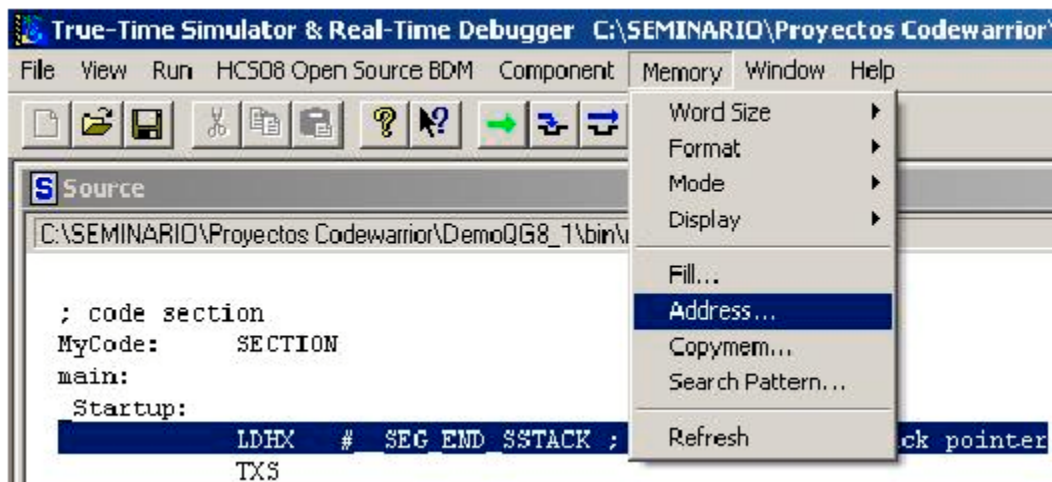
- 8) Dentro de la carpeta **“bin”**, estará el programa ejecutable bajo el nombre **“Project.abs”**, seleccionarlo y luego hacer click en **“abrir”** para confirmarlo....



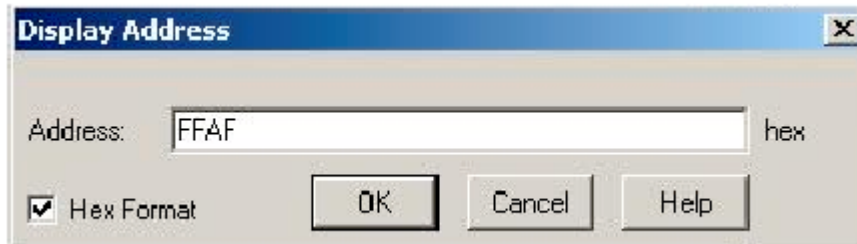
- 9) Hecho lo anterior, el programa ejecutable del usuario se cargará en la memoria FLASH del MCU sin alterar los valores contenidos en las posiciones reservadas para los valores de ajuste de fábrica del oscilador. Como el programa del usuario utilizará estos valores para copiarlos a los registros en RAM del ajuste del oscilador interno, la aplicación correrá con el oscilador ajustado a los valores de fábrica.
- 10) Para comprobar que ello es así, el usuario puede observar los valores grabados de fábrica en la memoria Flash haciendo click en la opción “Windows” de la barra de herramientas del entorno de Debugger y luego hacer click en la opción “Memory”.



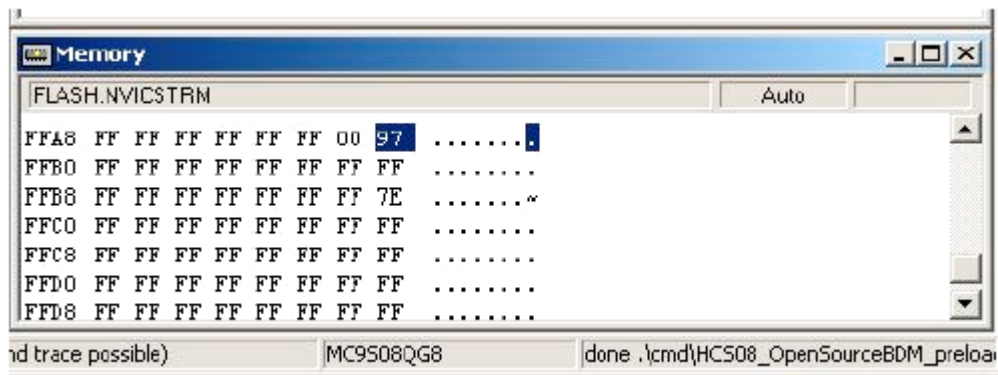
- 11) Una vez hecho ello, aparecerá en la barra de herramientas la nueva opción “Memory” y allí se deberá seleccionar la opción “Address” según se puede ver en la figura....



- 12) Para nuestro ejemplo, donde utilizamos la placa “DemoQG8” que contiene el MCU MC9S08QG8CPE, elegiremos la dirección **\$FFAF** que corresponde a la posición de memoria reservada en flash para el valor de fábrica **NVISCTRIM** y **\$FFAE** para el valor de fábrica **NVFTRIM**.



- 13) Finalmente, en la ventana “Memory” pueden observarse los valores de fábrica en las posiciones de flash (\$FFAE / \$FFAF) para el MC9S08QG8, según se muestra en la figura...



De esta forma el usuario puede aprovechar los valores de fábrica para correr las aplicaciones con una buena precisión del oscilador interno.