

## COMENTARIO TÉCNICO

# *Buceando en el HC908.....*

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)

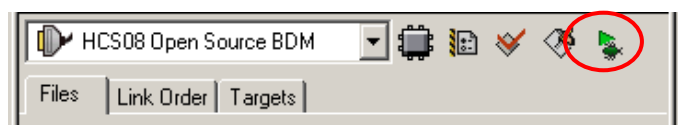
## *“Practicando con el CodeWarrior 5.1/6.x y el sistema R(S)\_POD.....”*

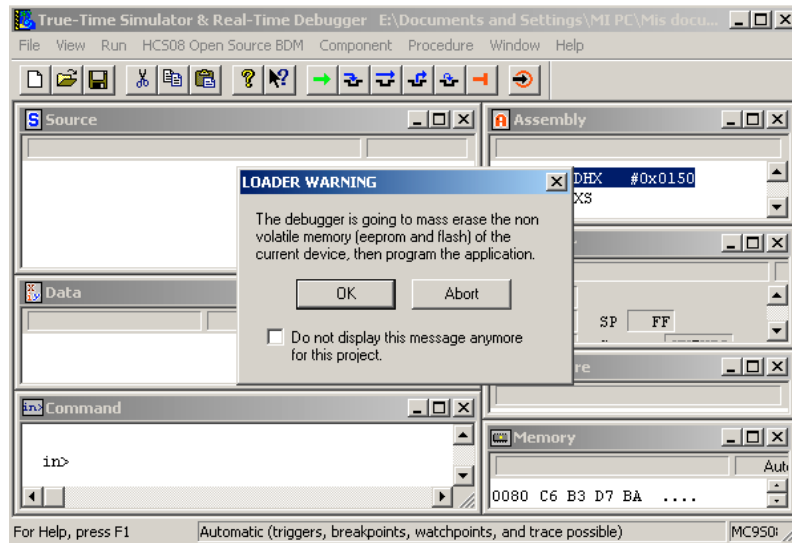
### **3era. y última Parte.**

En los dos artículos anteriores, habíamos realizado los pasos para la generación de código en forma asistida por el aplicativo “*Processor Expert*” que dispone el sistema *codeWarrior 5.1/6.x*, y también procedimos a agregar nuestras propias líneas de código para invertir en forma periódica el puerto “PTB0” del PORTB cada vez que se ingrese a la rutina de interrupción correspondiente al Timer Overflow del TPM.

Ahora, nos ocupa la prueba del código generado, y nada mejor para ello que efectuar una **Emulación en Tiempo Real** por medio de la herramienta *R(S)\_POD*.

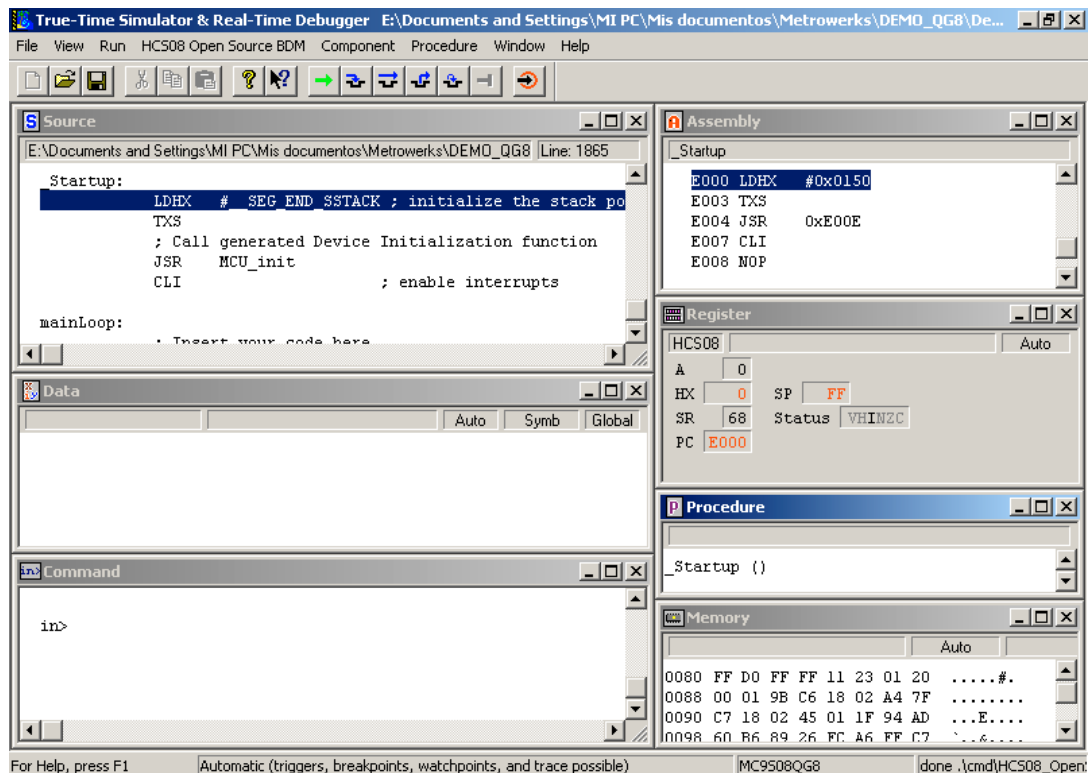
Para llevar a cabo esta tarea, primero deberemos verificar que se haya elegido la opción “**HCS08 Open Source BDM**” en el cuadro de selección de herramientas y que se encuentre conectado el sistema *R(S)\_POD* al puerto USB como se detalló en artículos anteriores. Presionar el switch **SW1 con retención** y deberá encenderse el **LED LD3 (MCU\_PWR)**. Hacer click en la fecha “**verde**” (**Debug**) junto al cuadro de selección de herramientas como se puede ver en la siguiente figura.....



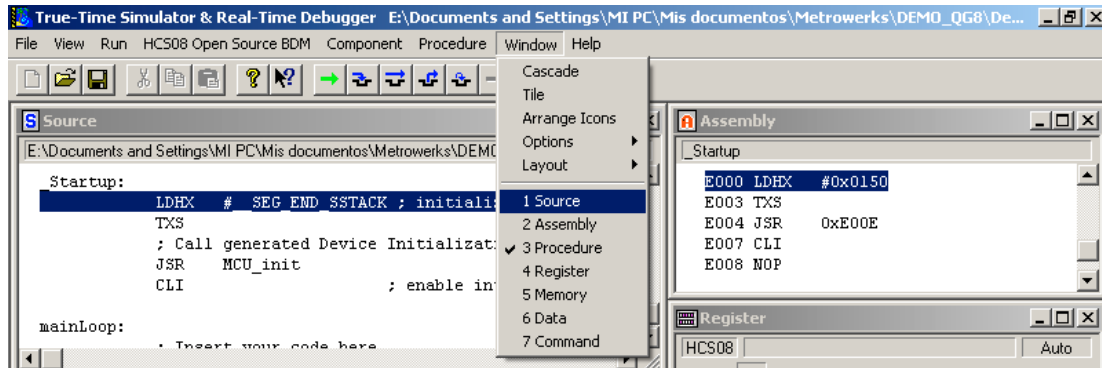


Se abrirán múltiples ventanas, y en la última nos indicará que la memoria FLASH del MCU será **TOTALMENTE** borrada (**mass erase**), para luego programar la aplicación a depurar en la memoria del MCU de la placa de “**Demo**” (**DemoQG8**).

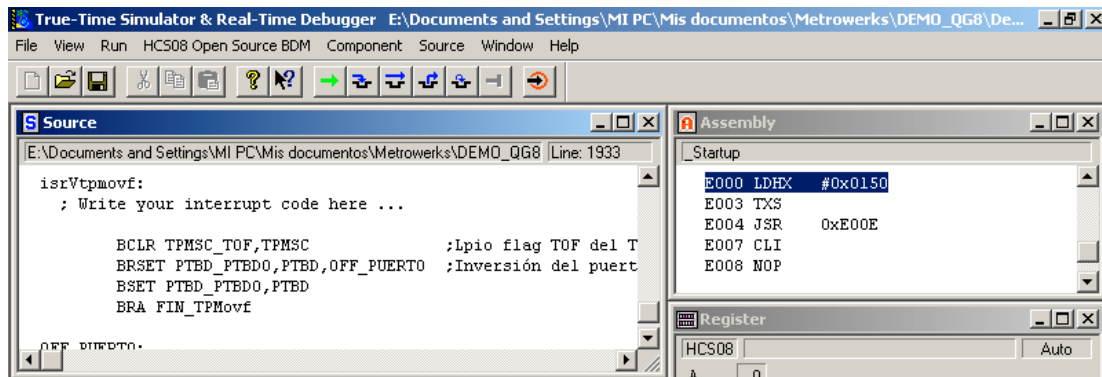
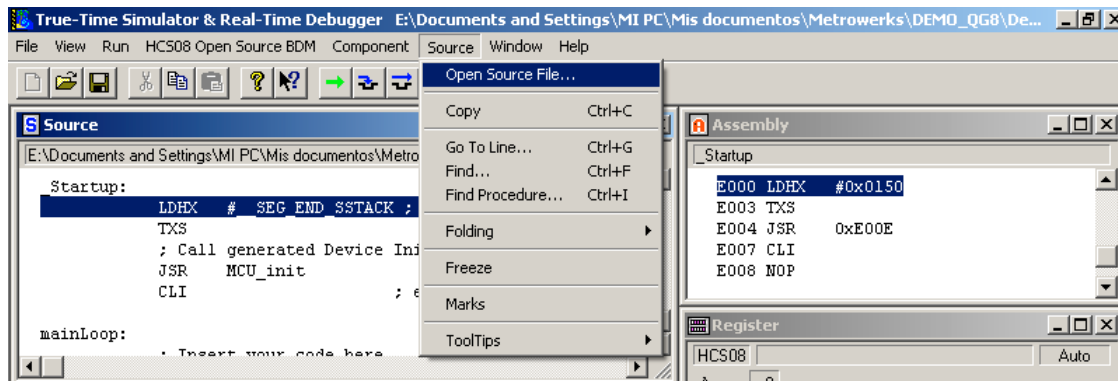
Al hacer click en “**O.K.**”, se presentará la siguiente ventana de Debug.....



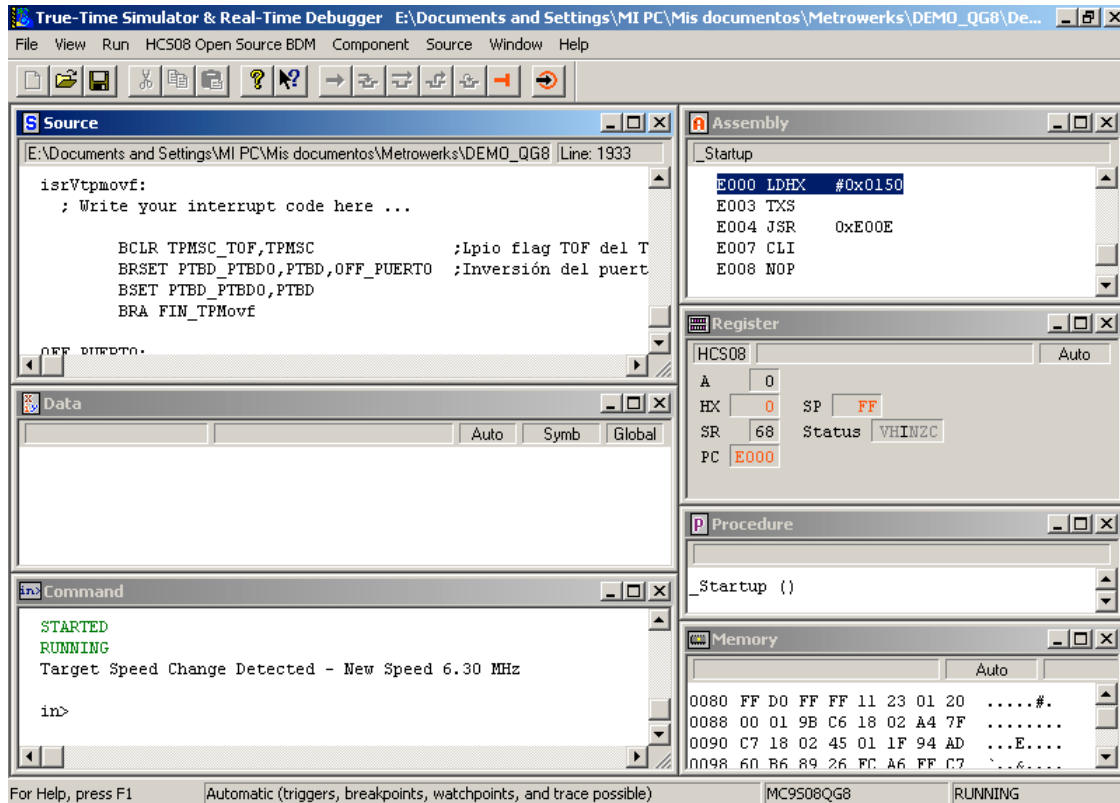
Como se quiere observar el código implementado para el ejemplo, y en especial las líneas agregadas para invertir el estado del puerto **PTB0**, se deberá seleccionar la opción **“Window”** y al desplegarse la ventana de opciones, se hará click en **“Source”** para activar esta posibilidad.....



Al activarse la opción **“Source”**, hacer click en **“Open Source File”** para luego elegir el archivo **“MCUint.dbg”**, según se muestra en las figuras.....



Si se quiere hacer “**Correr**” la aplicación en **Tiempo Real**, sin colocar un “Break Point” (punto de parada), simplemente hacer click en el icono con la **flecha “verde”** y la aplicación comenzará a correr en tiempo real sin detenerse (simultáneamente desaparecerán los iconos relacionados con la ejecución de código en distintos modos), y solo quedarán los de **HALT** (detención) y **Reset** (reset general del MCU).



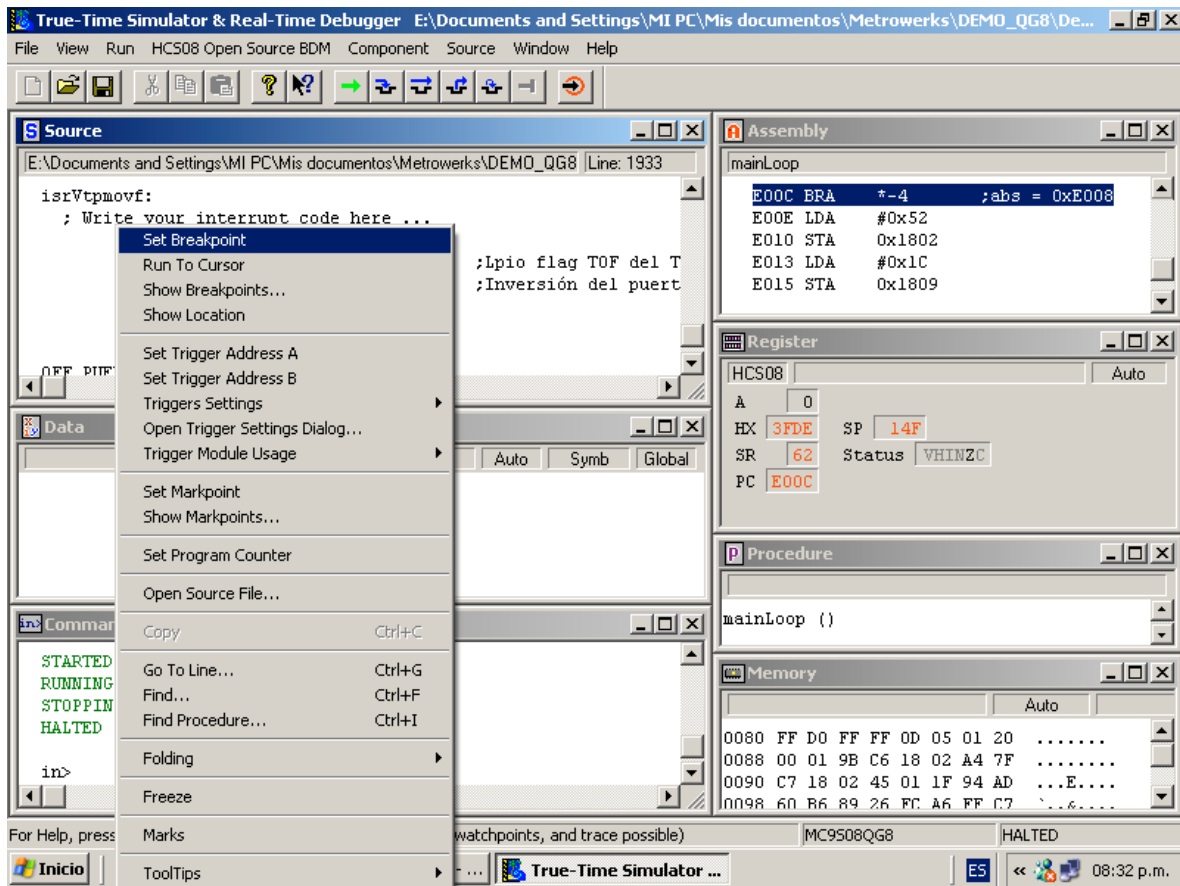
Por el puerto **PTB0** (pin 12 de la placa **DemoQG8**) saldrá una señal cuadrada de 500 ms de período en forma ininterrumpida .....

### Atención:

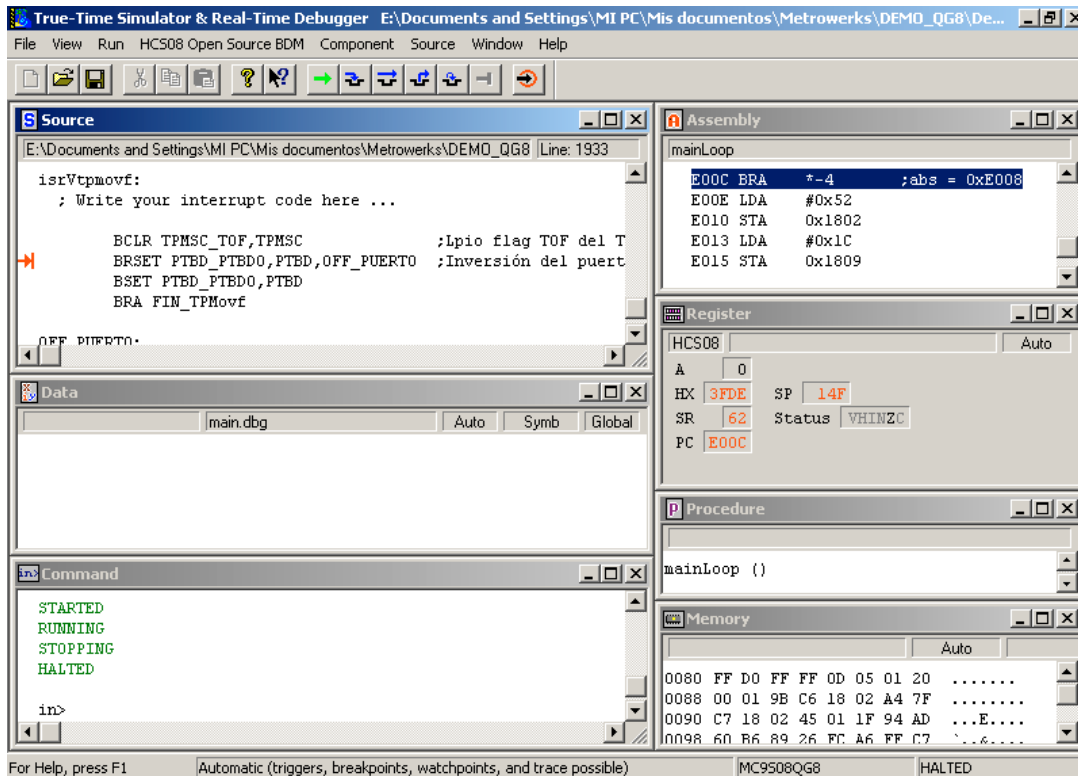
Debido a que el oscilador interno del **MC9S08QG8** se ajusta por medio de unos valores contenidos en la FLASH del mismo (ver sección ICS, Internal Clock Source, en el manual del MCU) y que durante el proceso de borrado de TODA la memoria FLASH los mismos son borrados (con \$FF), la frecuencia final del oscilador puede ser DISTINTA a la originalmente configurada en nuestro programa. Escapa al propósito de este artículo, explicar como solucionar este inconveniente.

Luego procedemos a detener la ejecución del código de aplicación haciendo click en el icono de **HALT**.....

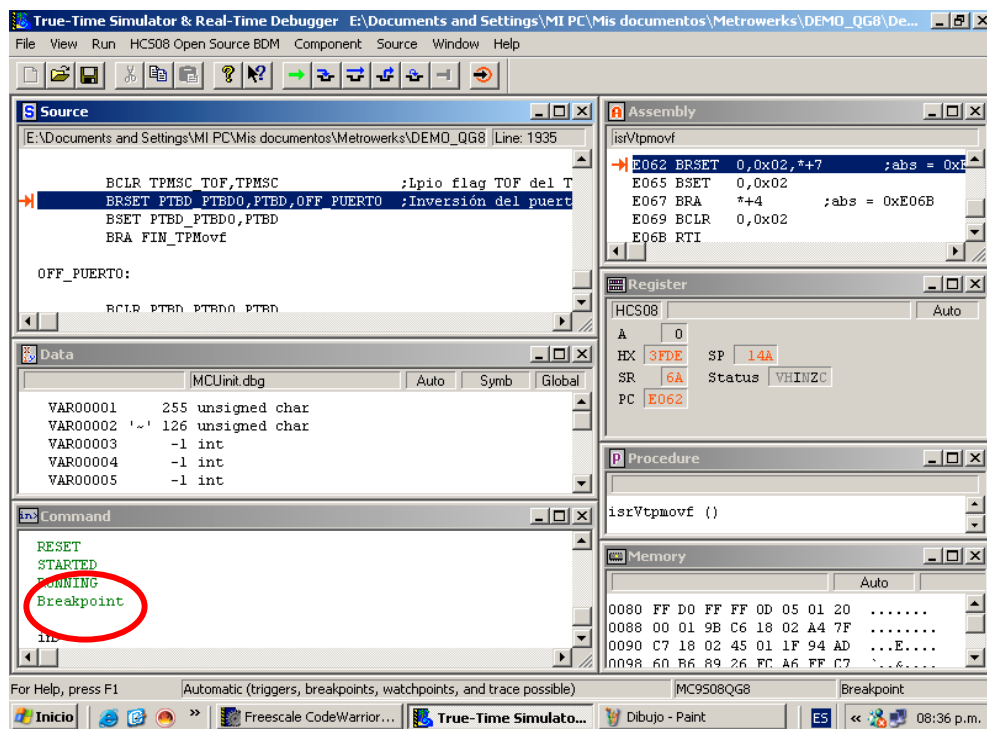
Para colocar un Break Point, en cualquier punto del programa de ejemplo, hacemos click en la línea que nos interesa colocarlo y presionamos el **botón derecho del mouse** y elegimos la opción **“Set Breakpoint”**.



Y se representará el mismo por una **flecha “Roja”** a la izquierda de la línea elegida, según se puede observar en la figura.....



Al hacer Click en la **flecha “verde” (RUN)**, la aplicación **correrá hasta el breakpoint** previamente marcado y se **detendrá en él**, como se muestra en la figura.....



Bueno amigos, hasta aquí vimos como utilizar el entorno CodeWarrior 5.1/6.x y la herramienta de bajo costo R(S)\_POD con un MCU “QG8” de la familia HC9S08.

Sugerimos al lector, ampliar los comandos y funciones disponibles por medio de la opción “**Help**” en el entorno “*CodeWarrior*” o en los numerosos tutoriales disponibles en el sitio del mismo ([www.freescale.com/codewarrior](http://www.freescale.com/codewarrior) ).

*¡Será hasta la próxima! .....*

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

