

COMENTARIO TÉCNICO

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



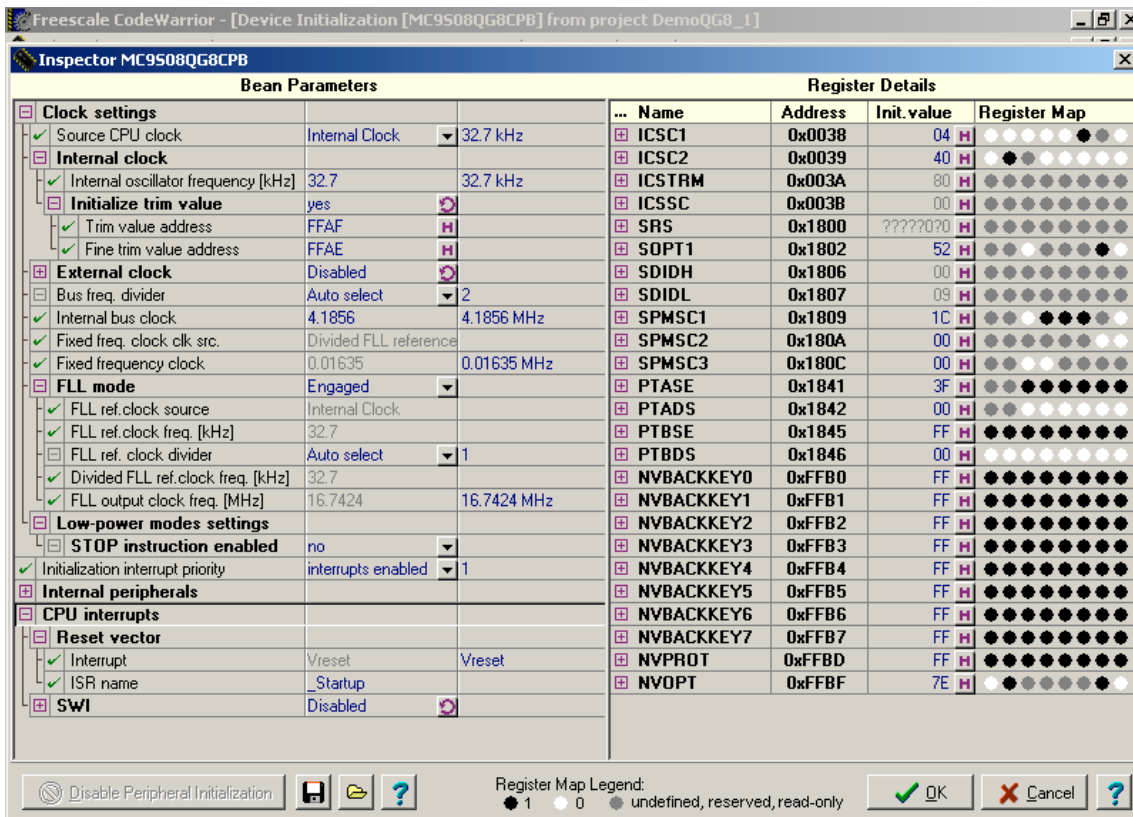
Por Ing. Daniel Di Lella  
Dedicated Distributor Field Application Engineer  
For Freescale Semiconductor Products & Technical Consult  
Dto. Técnico Electrocomponentes S.A.  
[fae@electrocomponentes.com](mailto:fae@electrocomponentes.com)  
[dilella@arnet.com.ar](mailto:dilella@arnet.com.ar)

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

### **2da. Parte.**

Continuando con el proyecto que empezamos en el artículo anterior, y en donde habíamos comenzado con la etapa de configuración del CPU (como se puede ver en la figura), proseguiremos con los módulos de Timer y puertos A y B.

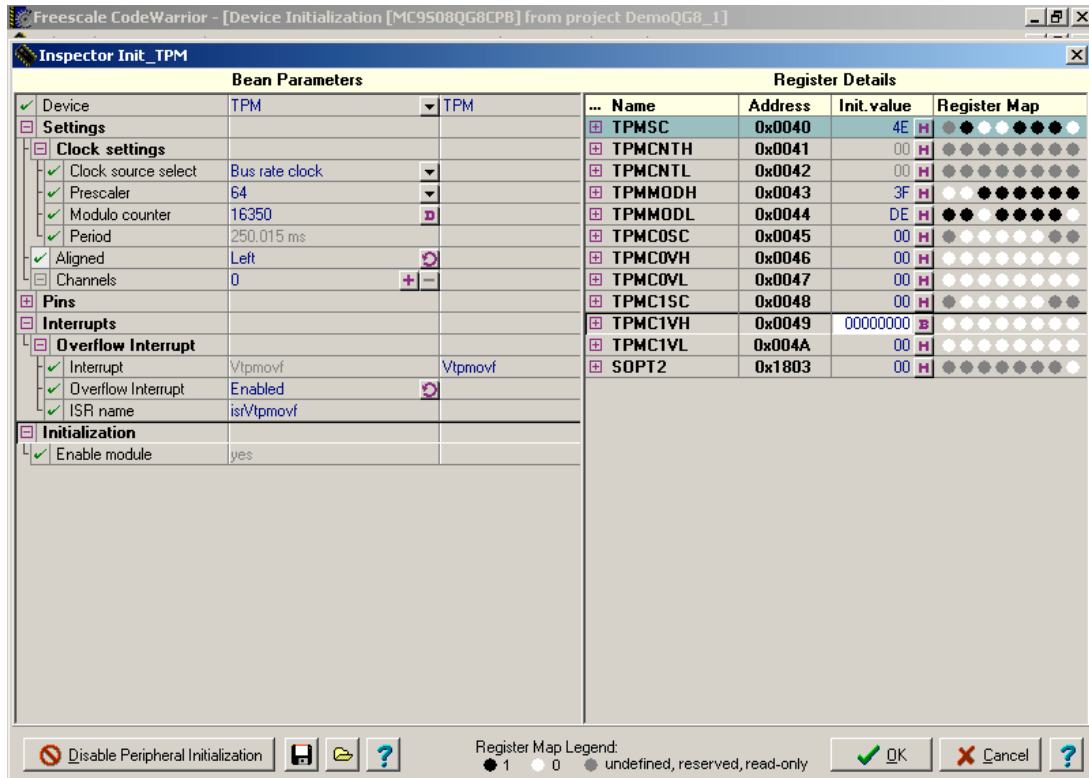
## Pantalla CPU:



11) Una vez configurados los distintos puntos del cuadro de configuración del módulo de CPU, hacer click en el icono “O.K” para confirmar los cambios y el sistema volverá a la pantalla principal. Ahora será el turno de la configuración del **módulo de TIMER** haciendo click sobre el cuadro con el nombre “TPM”.

12) Se configurará el mismo con los siguientes parámetros:

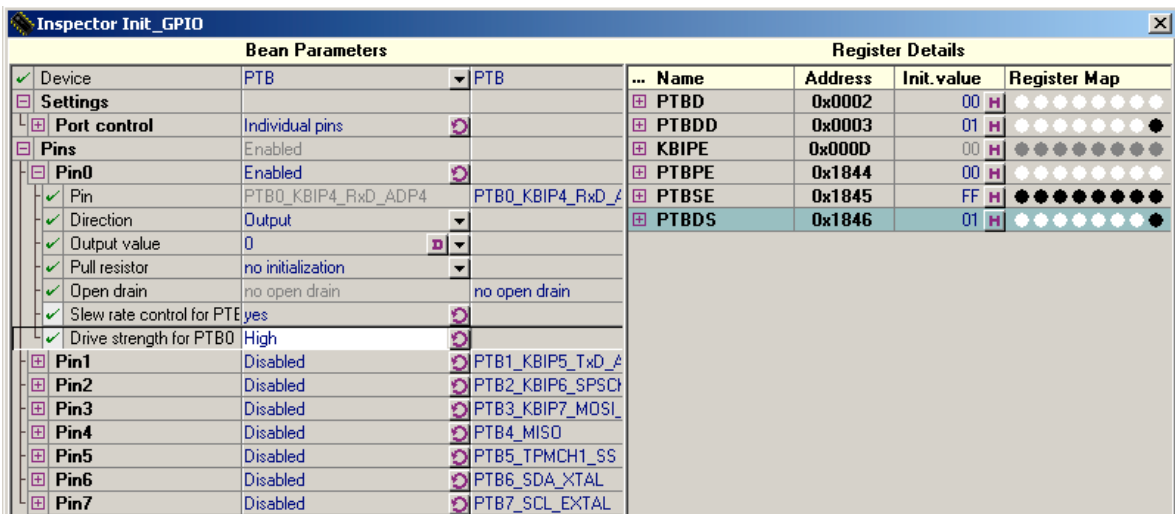
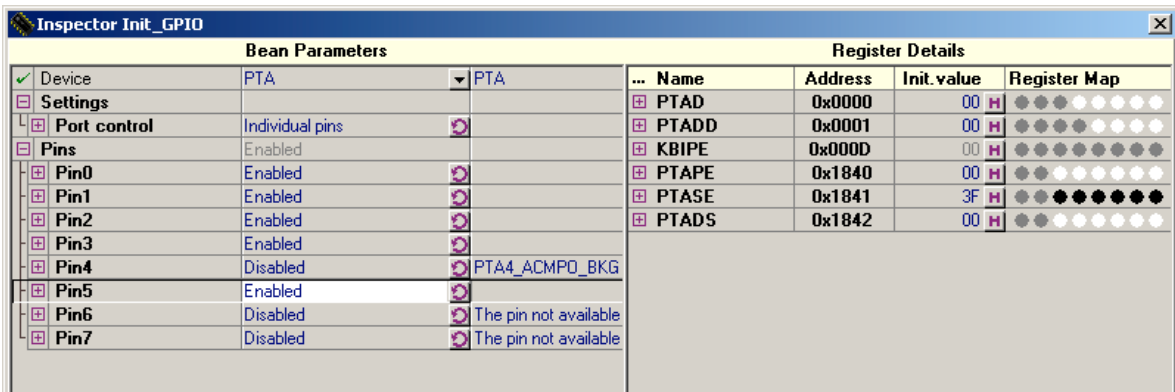
- Selección de fuente de Clock → Bus rate Clock.
- Prescaler → 64.
- Módulo Counter → 16350 cuentas.
- Período → 250 mSeg.
- Alineado → izquierda.
- Pines → No habilitados.
- Interrupción por “Overflow” → habilitada → “isrVtpmovf” (etiqueta de la **Interrupción**).
- Inicialización → Módulo habilitado.



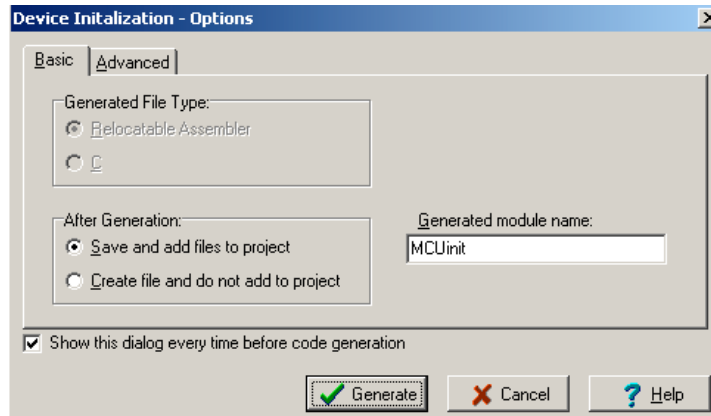
13) Al igual que en el módulo anterior, se procederá a hacer click en el botón “O.K” para confirmar la configuración de este y volver a la pantalla principal. Luego será el turno de los módulos de los puertos A y B (PTA / PTB) que los configuraremos de la siguiente manera:

PTA (PORT A) → PTA0 / PTA3 → Habilitados → Input  
 → PTA4 → No habilitado  
 → PTA5 → Habilitado → Input  
 → PTA6 / PTA7 → No Habilitados.

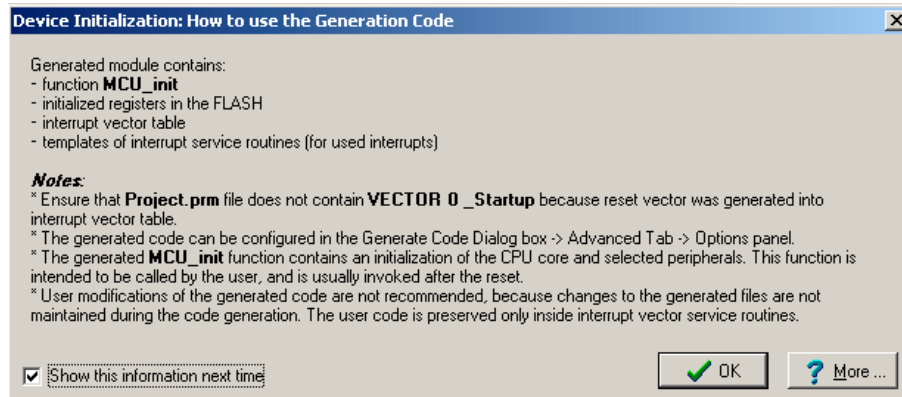
PTB (PORT B) → PTB0 → Habilitado → OUTPUT (control de corriente HIGH).  
 → Valor de arranque → “0”.  
 → PTB1 / PTB7 → No Habilitados



- 14) Si luego se presiona el botón **“Generate Code”**, el generador de código del Processor Expert generará código y nos mostrará una ventana explicando los pasos a seguir para incorporarlo efectivamente al resto del programa.



Se abrirá una pantalla previa que nos dará distintas alternativas de configuraciones (no tocar, solo para expertos!!) y nos dirá que el código generado llevará el nombre del módulo “MCUint” y se deberá proceder a presionar el botón “Generate” .....



Otra pantalla nos dará algunas recomendaciones a tener en cuenta y se procederá a hacer click en el icono “O.K” para obtener finalmente el código generado.

15) Hecho esto, se estará en condiciones de introducir nuestras líneas de código en la sub rutina de interrupción por Timer Overflow (**isrVtpmovf**) ubicada dentro del módulo generado MCUint (**MCUint.asm**) para realizar, por ejemplo, un Toggle (inversión de estado) cada vez que atendemos la interrupción propiamente dicha. En este punto podemos poner todas nuestras tareas en forma de llamado a sub rutina que se irán ejecutando una a una cada 250 ms. Las líneas a agregar pueden ser las siguientes...

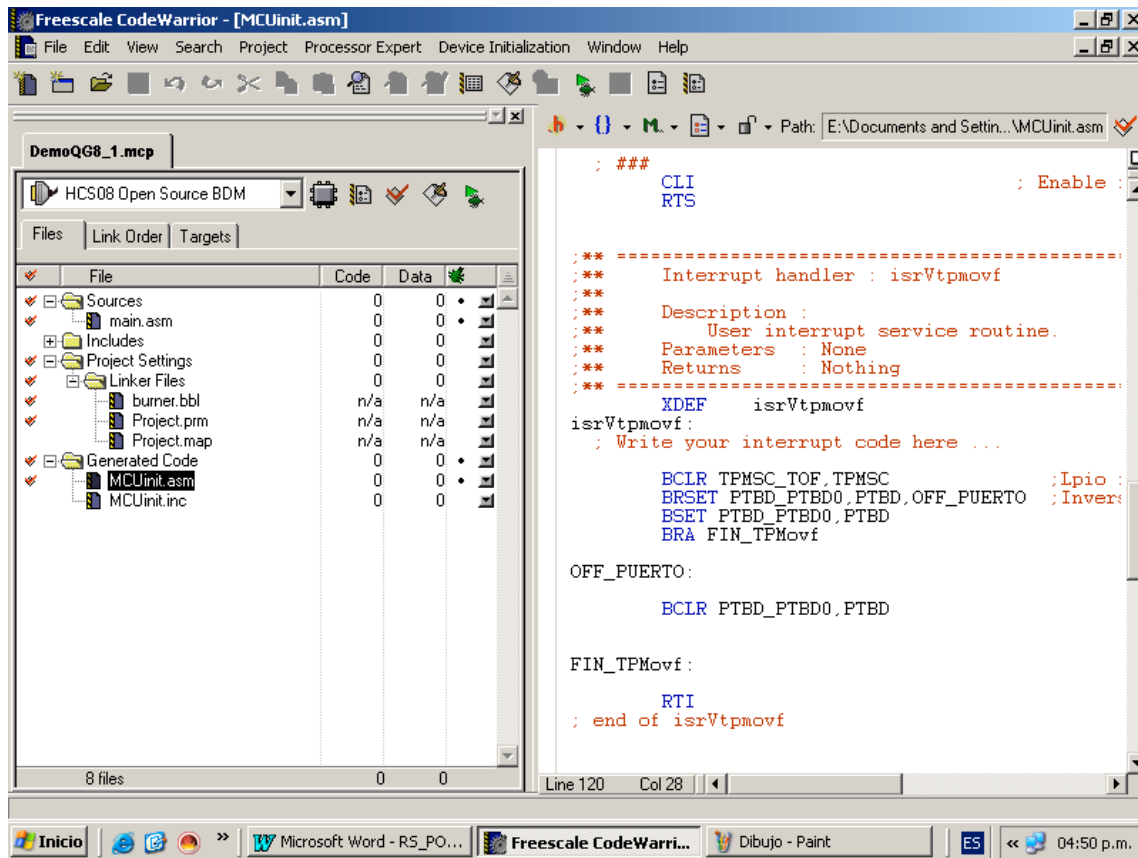
```
BCLR TPMSC_TOF,TPMSC           ;Lpio flag TOF del TIMER
BRSET PTBD_PTBD0,PTBD,OFF_PUERTO ;Inversión del puerto...
BSET PTBD_PTBD0,PTBD
BRA FIN_TPMovf
```

OFF\_PUERTO:

```
BCLR PTBD_PTBD0,PTBD
```

FIN\_TPMovf:

```
RTI
```



- 16) Una vez introducido nuestro código, se deberá compilar haciendo click en el botón **“Make”** en la barra de proyecto o en la barra de herramientas general. Si no se ha cometido algún error de compilación se estará en condiciones de pasar a la etapa de **Emulación en Tiempo Real** del programa de ejemplo.

*Continuará .....*