

# Como criar um novo projeto usando o Processor Expert.

## 1º Atribuição de Ambiente

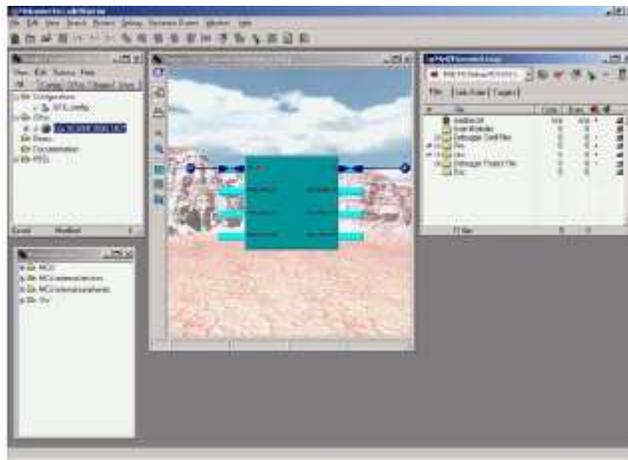
- Você deve ter o CodeWarrior para 68HC08 2.1 Edição Especial instalada em seu computador.
- Você necessitará do Motorola 68HC908QT para poder executar o projeto final.

## 2º Projeto a ser criado

- Nós vamos criar passo a passo esse projeto pelo qual será abordado como usar um dos periféricos no microcontrolador, neste caso será o conversor ADC.. A aplicação irá usar um potenciometro e um LED de cor vermelha. Para valores analógicos de 0 a 86 o LED ficará apagado, acima de 170 o LED ficará ligado, e entre 86 e 170 ficará alternando.

## 3º Criando um novo projeto

- No menu do CodeWarrior, **Arquivo Selecionado > Novo**
- Certifique-se que a tabulação 'Projeto' está ativa, Selecione **HC08 Imutável**
- Entre um nome para o projeto : '**MyQT4project**'
- Mude o diretório se você desejar (Localização, Set...)
- Pressione **OK**. Uma nova janela irá abrir para você selecionar o dispositivo e a linguagem
- Selecione o seu MCU e a linguagem, que no caso será **C with Processor Expert**.

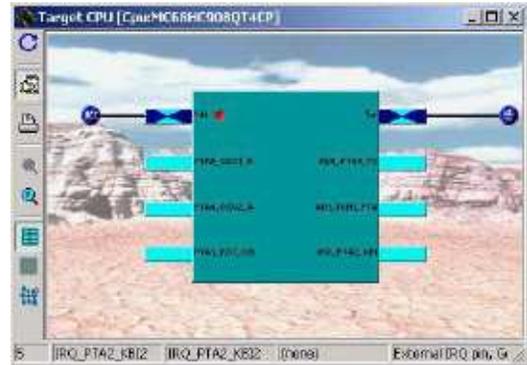


- Pressione **OK**.

Um novo projeto é criado usando o modelo 68HC908QT1 a ser desenvolvido pelo Processor Expert.

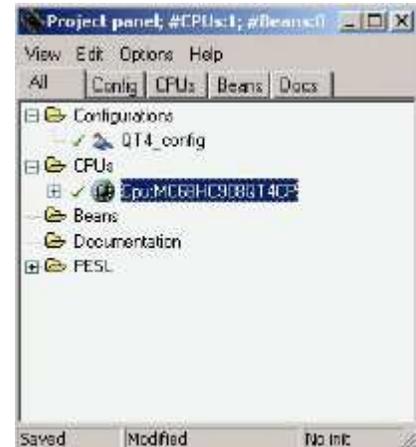
### 3º Target CPU(CPU ALVO)

A 'CPU Alvo' no centro da janela mostra o processador selecionado para o projeto. No dispositivo, nós vemos os diferentes periféricos do MCU, tal como: CPU, Timer, A/D conversor. Módulos com um ícone anexado para eles são módulos utilizados pela aplicação. Os pinos que são indicados para funções externas estão indicados por uma linha e um ícone, símbolo da função anexada (CPU e Porta A).



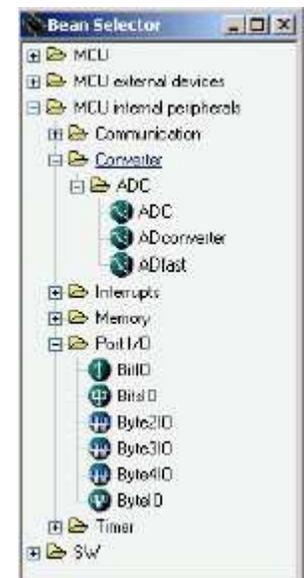
### 3.2'Project Panel' window

O 'Project Panel' Window mostra os ícones que terão sido criados para esta aplicação. Pressionando-se o símbolo [+] próximo a um ícone mostra uma lista de métodos e(ou) eventos relacionados ao MCU. Um tick verde indica que o método ou evento especificado está selecionado e uma cruz vermelha que o código não foi gerado.



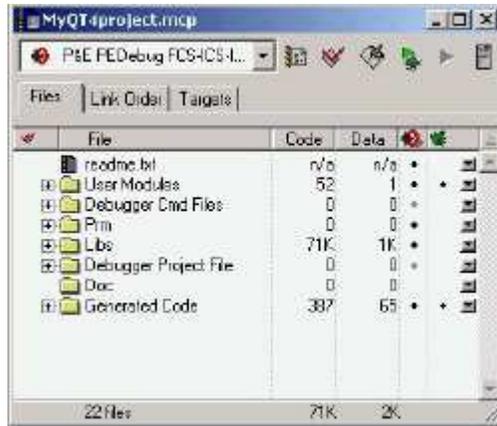
### 3.3'Bean Selector' window

A janela 'Bean selector' oferece ao desenvolvedor uma lista de segmentos (bean) para construir o projeto. Os segmentos disponíveis dependem da versão do CodeWarrior instalado. A versão Padrão e Edições Profissionais oferecem uma série mais vasta de hardware e segmentos de software que a Edição Especial. No exemplo, o conversor A/D e o segmento de portas I/O estão expandidos. Eles contêm os dois segmentos que necessitamos utilizar no projeto :ADfast e BitIO. Desdobre a árvore para fazê-los visíveis!



### 3.4 'Project Manager' window

O '*Project Manager*' é uma janela do CodeWarrior padronizada mostrando os arquivos a serem compilados.



### 4º Adicionando Bean ao projeto

Nós vamos adicionar dois segmentos (bean) ao projeto para construir nossa aplicação: um ao conversor A/D recebendo informações analógicas e um ao I/O comandando o LED.

#### 4.1 Criando o A/D bean

- Na janela do bean selector, de um clique duplo na '**ADfast**' bean. Isto abre a janela 'bean inspector' que será utilizada para montar os parâmetros para esta função. (ignorar a mensagem de erro que aparece quando você abrir)

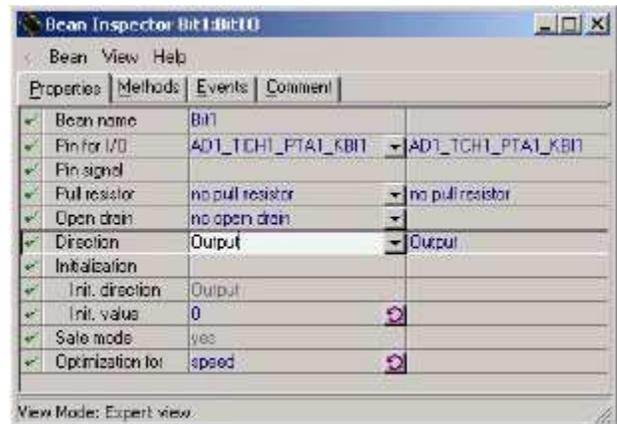
Selecione a aba **Properties** (propriedades):

- Para **desabilitar** ou **habilitar** algum serviço / evento (Service / Event) pressione a figura da flecha encurvada localizada a esquerda do mesmo. (Se você não estiver vendo a tela estendida, vá para o menu 'VIEW' e selecione 'visão Perita').

-Para selecionar o(s) pino(s) A/D do canal 0 (somente um canal selecionado) pressione a seta para selecionar **PTA5\_OSC1\_AD3\_KBI5**.

- Pressione o (...) para selecionar o tempo de conversão (conversion time) e no valor inicial requerido para conversão (Init.Value-Requested) entre com o valor **21.25** (certifique-se que a unidade está  $\mu$ S); pressione em OK para fechar o diálogo. (Ou clique duplo no valor listado.)

- Verifique se o 'low speed' e 'slow speed' estão desabilitados, se não, pressione na flecha encurvada.



Agora selecione **Methods** para o próximo passo.



Nesta janela você habilitará as funções a serem criadas. Neste caso você deverá habilitar conforme a figura mostrada a cima, ou seja, você deverá habilitar as funções MEASURE e a GETVALUE. As demais deverão ser desabilitadas.

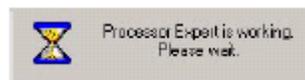
## 4.2 Criando o I/O bean

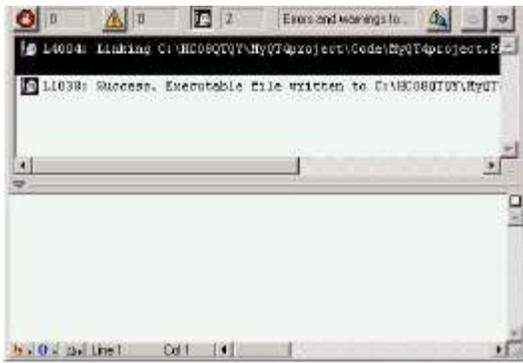
Na janela do **bean selector** dê um duplo clique em BITIO, para poder entrar na janela apresentada na figura acima. Agora basta você configurar os demais itens , primeiro passo será indicar a porta que será usada no projeto, no nosso caso será a porta PTA1. Depois informe a direção da porta que no nosso caso será saída. O próximo passo será indicar o valor inicial , no nosso caso será em lógica 0. Depois será selecionar a otimização que neste caso será alta. Feito isso tudo, agora vamos para METHODS que é usado para indicar quais funções serão criadas pelo Processor Expert.

**Pronto basta configurar conforme a figura acima que estará tudo certo.**

## 5 Como gera o código driver.

**Feito tudo isso conforme foi descrito só resta agora compilar o projeto para ver se está tudo ok. Para compilar basta clicar F7 daí irá aparecer uma figura que vocês podem observar abaixo.**

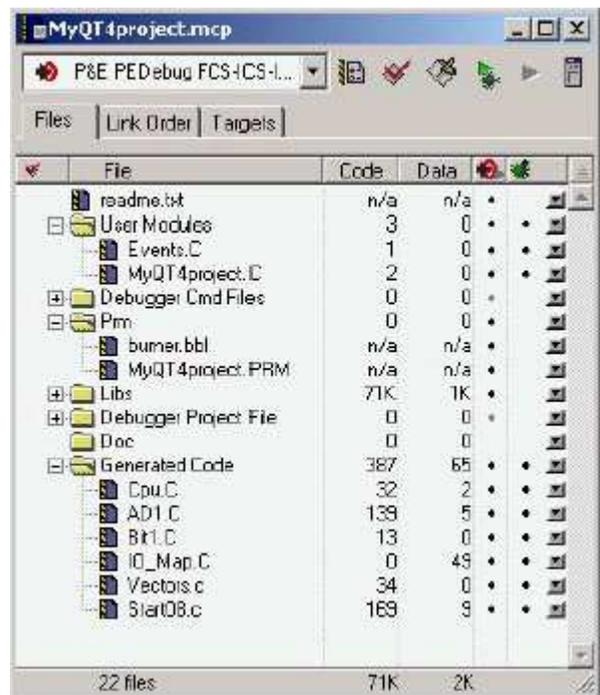




**Se a figura acima aparecer indicando que a compilação foi um sucesso, é sinal que o código fonte foi criado com sucesso pelo Processor Expert.**

## **6 Verificando os arquivos criados**

**Os arquivos criados ficam na pasta *GENERATED CODE*, os arquivos com os nomes *AD1.C* e *BIT.C* são os códigos fontes criados para nossa aplicação.**



**Agora você irá na pasta PRM, esta é pasta em que deverá ficar o código fonte que você irá escrever, que no nosso caso será conforme este abaixo.**

```
static byte myValues[1]; /* Numero de canais */
void main(void) {
byte min = 255/3;
byte max = (255*2)/3;
byte err;
bool s;
int i;
for(;;) {

err = AD1_Measure(TRUE);

err = AD1_GetValue((byte *)myValues);

if (myValues[0] > max) {
s = 1; /* set LED ON */
} else {
if (myValues[0] < min) {
s = 0;
} else {
s = s^1;
}
}
for (i = 0; i < 5000; i++);
Bit1_PutVal(s);
}
}
```

Feito isso, salve o projeto e compile, se tudo estiver certo clique em make depois em debugar circuito. Pronto, está feito o seu projeto utilizando o Processor Expert.

### ***Bibliografia***

***1º Manual do compilador Metrowerk***

***2º Revista Saber Eletrônica.***

***Autor: Wellington Messias Silva de Oliveira***

***Ocupação: Técnico em eletrônica.***

***Experiência: 8051, PIC, AVR, HC08***

***Linguagem: ASM para PIC, C para os demais microcontroladores citados.***