Tutorial para utilizar o ECLIPSE para programar para micro-controladores

Autor: Vinicius Kursancew

Para mais informações sobre este tutorial mande email para
viniciusalexandre@gmail.com

Versão: Apr 21, 2005

Introdução

O Eclipse é uma poderosa ferramenta multi-plataforma para programadores. O objetivo deste tutorial é explicar como se configura e trabalha com esta interface quando se deseja compilar programas para micro-controladores utilizando o conhecido compilador SDCC.

O principal alvo é permitir a criação de programas para 8051, mas este tutorial tenta ser o mais genérico possível, sendo que fica fácil estende-lo para outro alvo do SDCC.

Considero o SDCC um compilador bom, apesar de apresentar alguns pequenos bugs ainda (até o presente dia 5 bugs para o compilador de 8051) e em alguns casos não gerar código tão otimizado quanto o KEIL. Para projetos acadêmicos ele é um excelente compilador pois não tem a limitação de 2KB que a versão gratuita do KEIL impõe.

Obtendo o software

ECLIPSE

Você deve pegar o eclipse no site http://www.eclipse.org/ - clique na seção de downloads e baixe o eclipse para seu sistema operacional.

SDCC

Você deve obter o SDCC no site http://sdcc.sourceforge.net/

Pré requisitos

Ter um ambiente de JAVA 2 (mínimo J2SE 2.4.2) instalado em seu sistema

Instalação

SDCC

Caso você use linux, baixe o pacote para sua distribuição e siga os métodos normais de instalação. Você deve ter o make instalado.

Caso você use o windows você deve pegar o arquivo de instalação sdcc-x.x.x-setup.exe. Quando o download terminar abra o executável e siga os passos da instalação. Quando perguntar o tipo selecione FULL e de continuar. Diga não na hora de reiniciar o computador.

Para usuários de windows ainda é necessário pegar alguns utilitários de Unix para windows, acesse http://unxutils.sourceforge.net/ e feça download do UnixUtils. Descompacte o arquivo zip que você obteve num diretório temporário. Neste diretório temporário acesse a pasta USR/LOCAL/WBIN e copie tudo para a pasta BIN no diretório que você instalou o SDCC (normalmente c:\Arquivos de Programas\sdcc ou c:\sdcc).

Reinicie o computador. Para testar abra um prompt do MSDOS e execute o comando MAKE, você deve obter uma mensagem do tipo "*** No targets specified and no makefile found.". Depois execute o comando SDCC, você deve obter uma lista de opções de argumentos.

Nota sobre a instalação do windows: os seus arquivos do projeto devem estar no drive em que o SDCC foi instalado

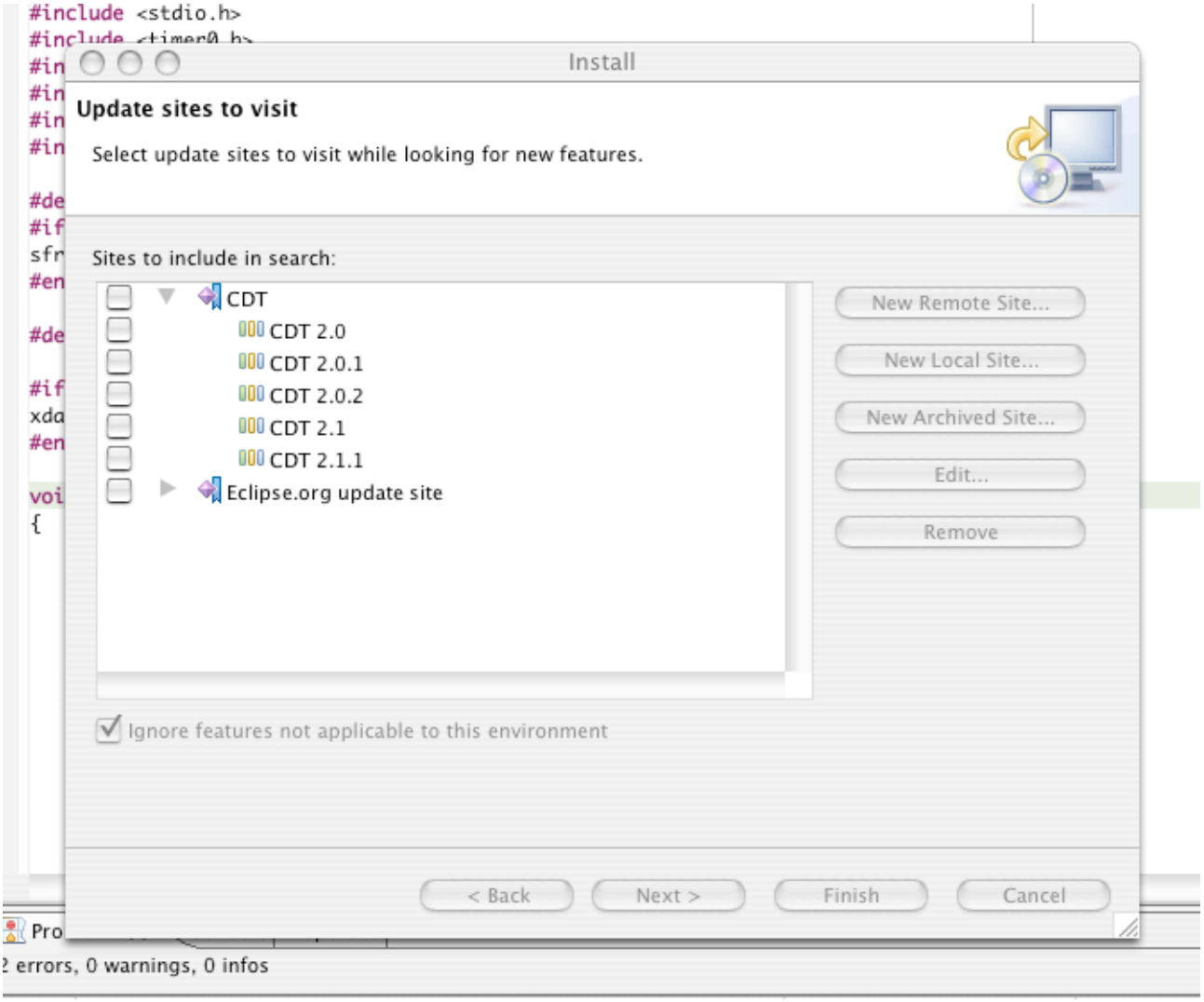
ECLIPSE

O Eclipse não tem nenhum tipo de programa de instalação. Simplesmente extraia o arquivo zip que você obteve em um diretório que você coloca seus programas e o Eclipse está instalado.

Configuração

Quando você iniciar o Eclipse pela primeira vez, será perguntado onde deverá ser mantida sua “workspace” (área de trabalho). Escolha um diretório. Neste diretório o Eclipse guardará todos os seus projetos e suas configurações.

Agora precisamos instalar os arquivos para suportar a linguagem C. Você precisa estra conectado à internet. Vá ao menu **help->software updates->find and install...** uma nova janela abrirá, selecione **Search for new features to install** e clique **Next**. Na proxima tela clique em **New remote site** e adicione http://update.eclipse.org/tools/cdt/releases/new/ à lista, ponha o nome CDT. Então selecione a versão mais recente do CDT e clique **next**.



Na próxima tela o Eclipse perguntará quais funções (features) você deseja instalar. Selecione todas e de Next, aguarde ele concluir a instalação e de Finish. Reinicie o eclipse.

Quando você abrir o eclipse novamente vá ao menu **Window->Open Perspective->Other...**, selecione C/C++ e de OK.

Makefile

Para gerenciar a compilação de nossos projetos dentro do eclipse nós trabalharemos com um programa chamado MAKE. Este programa lê regras de compilação de um arquivo chamado Makefile, que fica dentro do diretório do projeto. A estrutura do Makefile é simples, você deve especificar um alvo, suas dependências e o comando de compilação. Normalmente no início do Makefile são declaradas umas variáveis para facilitar a vida, como por exemplo o compilador a ser usado e as opções passadas ao compilador.

Existe um alvo especial, chamado **all**, que é o alvo que vai ser construído caso não seja passado nenhum alvo como o primeiro argumento do programa MAKE. Quando MAKE vai construir um alvo ele procura pelos arquivos listados como dependências. Quando ele não acha um arquivo ele vai procurar por um alvo cujo o nome seja o próprio nome do arquivo e vai executar este alvo.

No nosso caso os alvos serão ou arquivos IHX (são a mesma coisa que os HEX do keil) ou arquivos REL, que são os arquivos objeto, que depois são unidos e convertidos em um IHX. As dependências são arquivos C, H, IHX e REL.

Variáveis são declaradas no estilo: VARIAVEL=VALOR (proibido espaço entre o nome e o =) e são acessadas utilizando o cifrão e parenteses: $(VARIAVEL)

Pode-se colocar comentários no arquivo utilizando o #.

Abaixo um exemplo genérico de um Makefile que gera um arquivo IHX a partir de dois arquivos de fonte C.

```
CC=sdcc
CFLAGS=--iram-size 256 --model-small
#alvo generico, normalmente aponta para o principal como dependencia
all: programa.ihx

#alvo principal, dependencias listadas após o : (dois pontos)
#note o TAB que é obrigatório nas regras deste alvo
programa.ihx: programa.c modulo_lcd.rel
	$(CC) $(CFLAGS) programa.c modulo_lcd.rel

#como construir o arquivo modulo_lcd.rel
modulo_lcd.rel: modulo_lcd.c modulo_lcd.h
	$(CC) $(CFLAGS) -c modulo_lcd.h
```

Argumentos e peculiaridades do SDCC

Existem alguns argumentos do SDCC que são relevantes se comentar. Estes argumentos normalmente são colocados na variável CFLAGS no Makefile e sua utilização varia de acordo com o projeto.

Argumento	Descrição
-D	Define uma macro, ex -Dmacro
-I	Inclui o diretorio no caminho de busca de includes, ex -I/incs
-l	Adiciona uma biblioteca ao processo de link, ex -lx.lib
--model-large	Diz ao compilador para utilizar a ram externa para variáveis
--model-small	Diz ao compilador par utiliazar a ram interna para variáveis
-c	Compile apenas (nao gera o IHX). Usado para gerar os arquivos .rel de cada módulo para depois uni-los em um IHX
--iram-size	Diz o tamanho da ram interna (em numero decimal) do seu micro-controlador, ex --iram-size 256
--xram-size	Diz o tamanho da ram externa (em numero decimal) do seu micro-controlador, ex --xram-size 64000

Para obter mais argumentos do SDCC execute o SDCC sem nenhum argumento.

TAMANHO DE VARIÁVEIS

Tipo	Tamanho(bytes)
char	1
int	2
short	2
float	4
long	4

PECULIARIDADES PARA PROGRAMAÇÃO

Se voçê deseja forçar o local onde uma variável será armazenada use as palavras-chave idata e xdata, para ram interna e externa, respectivamente. Ex.

```
idata int foo;
idata char bar;
```

Para dizer que uma função é para ser executada como rotina de interrupção use a palavra-chave interrupt seguido do número da interrupção de acordo com a tabela. Ex.

```
void contador_timer() interrupt 1 { ++counter; }
```

Interrupção 8051	Número do SDCC
Externa 0 (INT0)	0
Timer 0	1
Externa 1 (INT1)	2
Timer 1	3
Serial	4
Timer 2 (apenas 8052)	5

MISTURANDO ASSEMBLER E C

Para inserir código assembler no meio de suas rotinas em C use a palavras-chave _asm e _endasm. Ex:

```
char rotina_mista(char k) {
     k; /* move k para o acumulador */
     _asm
          add a,#3
     _endasm;
     return k; /* retorna k+3 */
}
```

REGISTRADORES ESPECIAIS E PORTS

Para permitir que o compilador entenda os registradores para o 8051 ou 8052 deve-se adicionar como include 8051.h ou 8052.h, respectivamente. Os registradores especiais tem seu nome padrão assim como os bits acessíveis diretamente, ex para atribuir um valor a P1, use

P1 = 128;

Para atribuir um valor ao bit 0 da P1 utilize

P1_0 = 1;

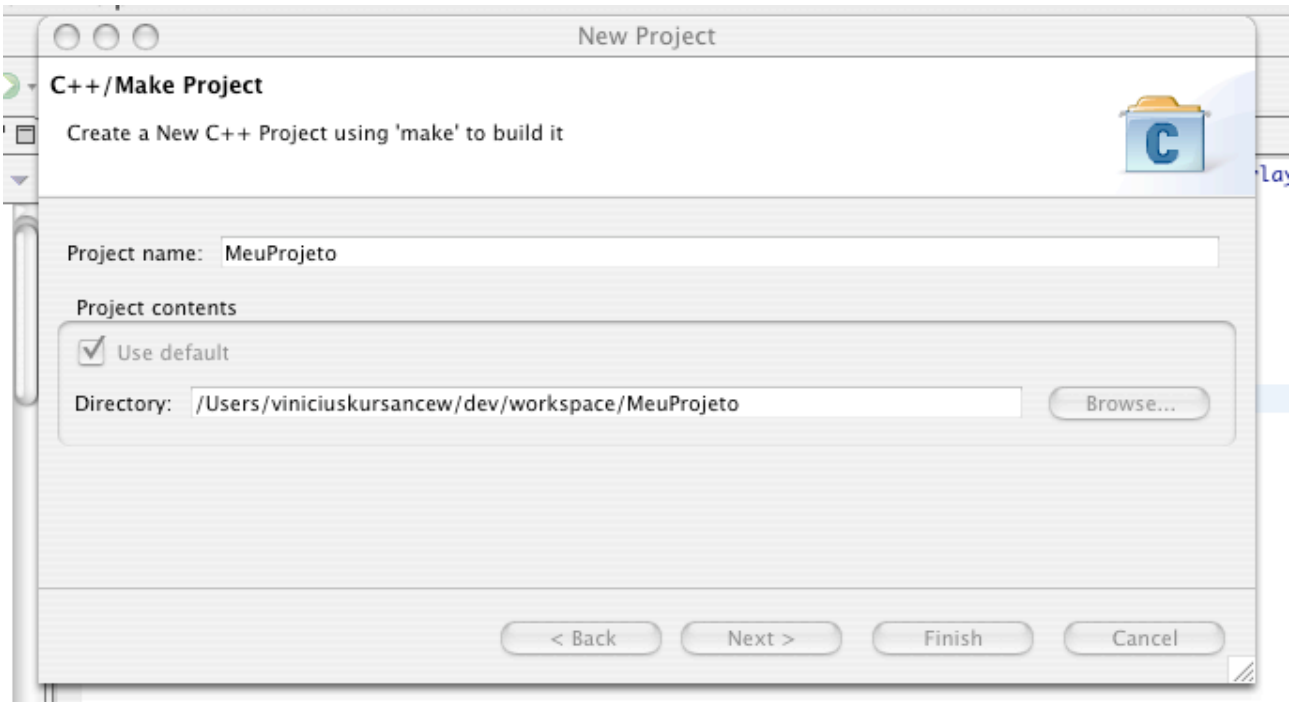
Veja os arquivos 8051.h e 8052.h para uma lista com todos os registradores. Estes arquivos situam-se na pasta includes dentro do diretório de instalação do SDCC (no windows normalmente em c:\arquivos de programas\sdcc\includes e no linux em localizado normalmente na pasta /usr/share/sdcc/includes).

Integrando tudo

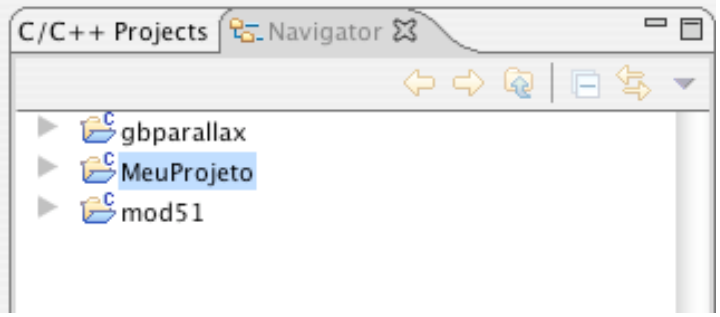
Agora vamos ver como realmente integraremos todos estes programas (make, sdcc e eclipse)

CRIANDO UM PROJETO

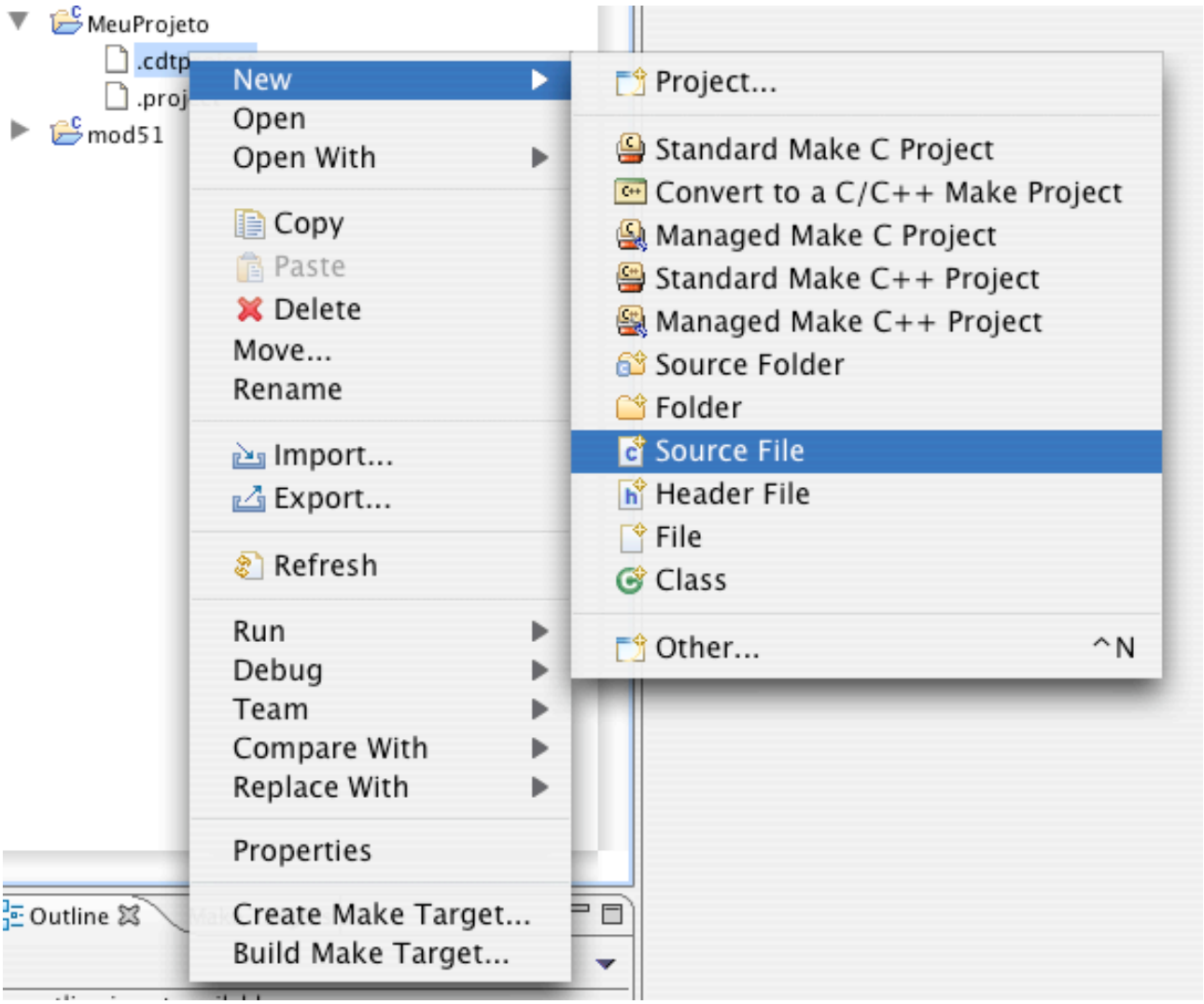
Vá no menu **File->New->Standard Make C Project...** e você verá uma tela para dar nome para seu projeto. De um nome a ele e clique Finish.



Você notará que na janela NAVIGATOR na esquerda, o seu projeto aparecerá.

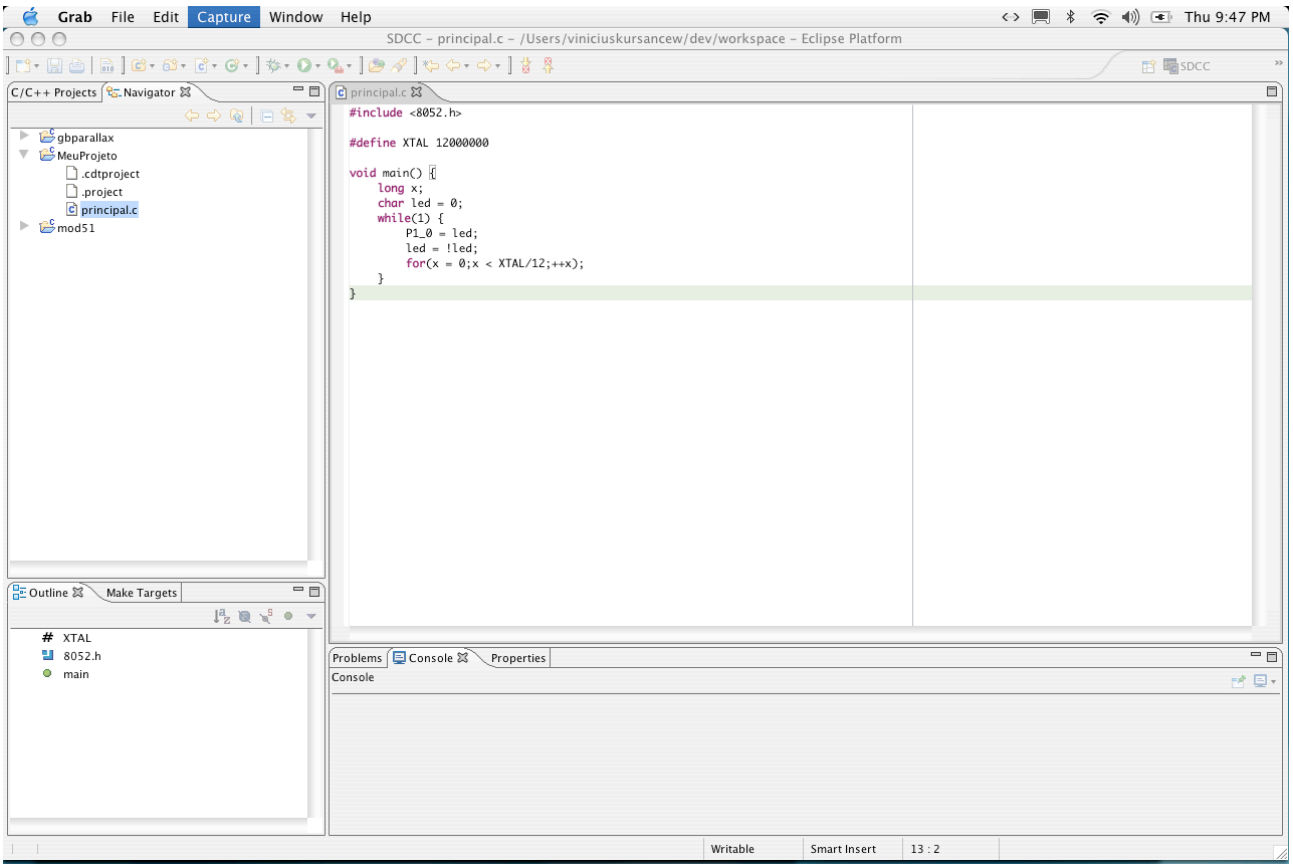


Agora vamos criar nosso arquivo fonte principal. Clique com o botão direito do mouse sobre o projeto e acesse **New->Source File**.



Nomeie o arquivo como principal.c e de Finish. O arquivo abrirá na janela do editor. No screenshot abaixo eu já digitei o seguinte programa, que faz com que um LED pisque na PORT 1.0, salve o programa depois de copiar:

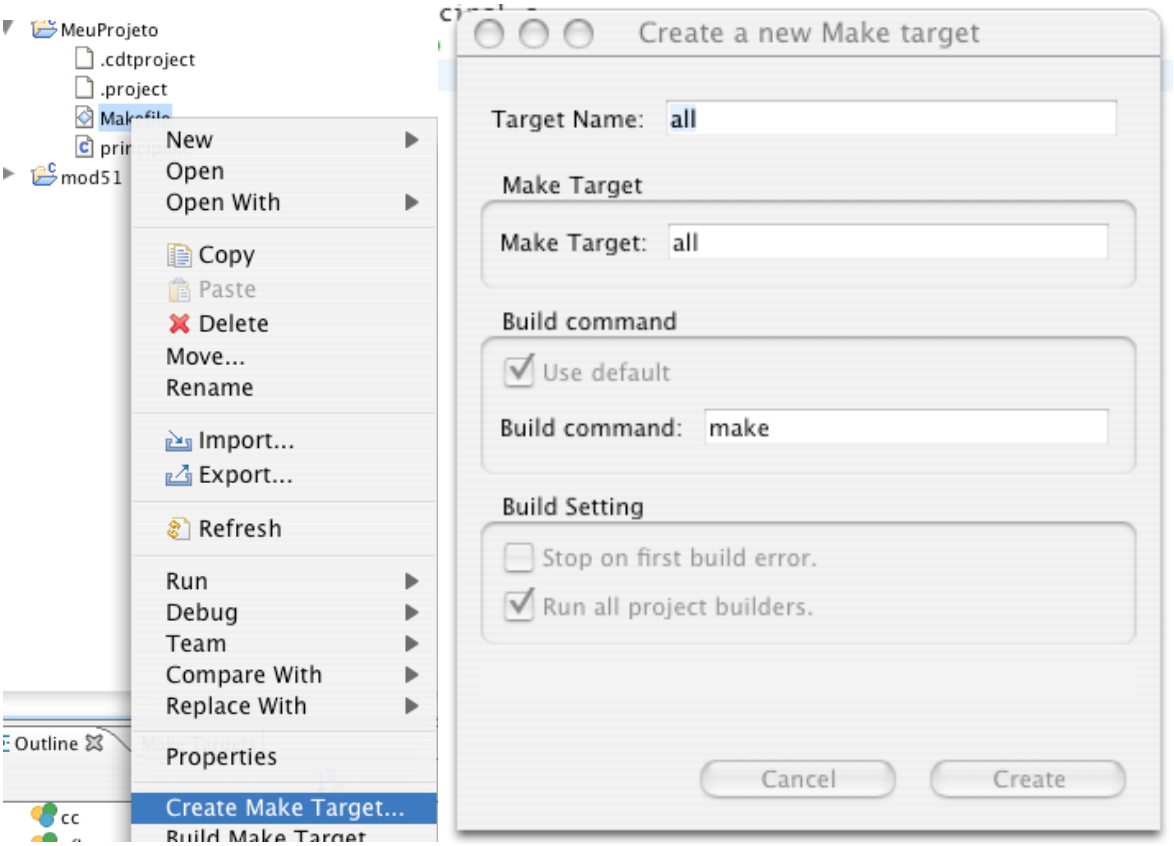
```
#include <8052.h>

#define XTAL 12000000

void main() {
    long x;
    char led = 0;
    while(1) {
        P1_0 = led;
        led = !led;
        for(x = 0;x < XTAL/12;++x);
    }
}
```



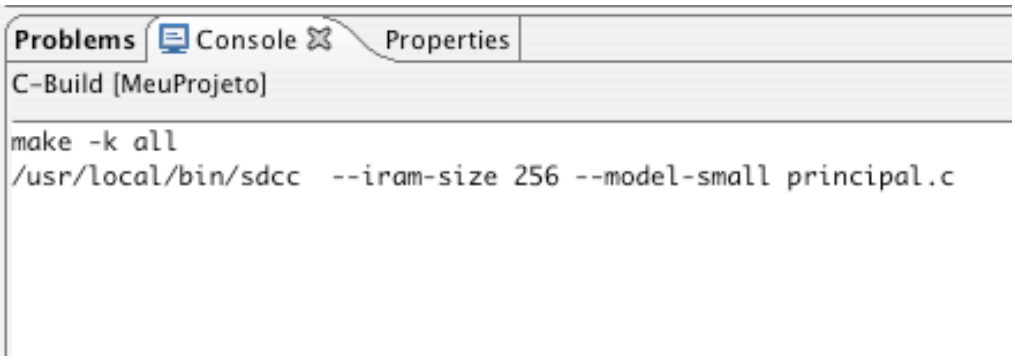
Agora precisamos criar um Makefile. Clique com o terceiro botão em seu projeto e clique em **New->File**. Nomeie o arquivo como Makefile. Entre com o seguinte no Makefile e depois salve:

```
cc=sdcc
cflags=--iram-size 256 --model-small

all: principal.ihx

principal.ihx: principal.c
    $(cc) $(cflags) principal.c
```

Agora vamos adicionar qual alvo a gente deseja compilar. Clique com o botão direito do mouse sobre o projeto e clique em **Create make target...**. Quando a janela abrir ela já vai estar preenchida para construir o alvo **all**. Como é isso que desejamos, apenas clique **Create**. Agora estamos prontos para compilar.



Para compilar clique com o terceiro botão no projeto e clique em **Build Make Target...** Selecione o alvo **all** e clique em **Build**. Caso apareça `make: sdcc: Command not found` na janela de console, no Makefile, em sua variável CC adicione o caminho completo para o SDCC, ex: CC=/usr/local/bin/sdcc

Depois de ter compilado vão aparecer vários arquivos gerados pelo compilador dentro do seu projeto. O principal é o IHX (Intex Hex Format), que é o arquivo que vai ser gravado no micro-controlador. Note que alguns programadores de dispositivo necessitam que você utilize o formato BIN para 8051. Neste caso você deve utilizar a ferramente HEX2BIN. Até a presente data ele pode ser baixado em
http://www.atmel.com/dyn/resources/prod_documents/HEX2BIN.EXE

Encontrado problemas de compilação: para ajudar a achar seus erros de programação exitem duas janelas, **Console** e **Problems**, estas estão situadas abaixo do editor. A janela Console lista todos os passo ocorridos durante a compilação e a janela Problems mostra os erros de uma maneira mais clara, e na maioria das vezes permite que você vá diretamente ao erro dando dois cliques nele.