



FUNDAÇÃO EDSON QUEIROZ
UNIVERSIDADE DE FORTALEZA
ENSINANDO E APRENDENDO

Aula 04

Ambientes de Programação

Microcontroladores PIC18 – Programação em C



Prof. Ítalo Jáder Loiola Batista

Universidade de Fortaleza - UNIFOR

Centro de Ciências Tecnológicas - CCT

E-mail: italoloiola@unifor.br

Jan/2011

Assembly versus C

- Linguagem C
 - Facilidade na construção de aplicações de grande complexidade;
 - Uma gama de funções nativas ao compilador;
 - Portabilidade da linguagem C;
 - Reduz o tempo de desenvolvimento;
 - Capacidade de introduzir no programa-fonte sub-rotinas, funções e bibliotecas escritas em linguagem *Assembly*;
- *Assembly*
 - Duas vantagens podem justificar a sua utilização:
 - Um bloco escrito em assembly contém um **menor número** de instruções, senão, conseqüentemente, executado **mais rapidamente** do que em C que executa a mesma tarefa;

Obs.: Um código escrito em C é convertido em Assembly;

Então, Para que um projetista **possa aproveitar** todas a vantagens do C, é necessário tenha pelo menos uma noção da Linguagem *Assembly*.

Ambientes de Programação

- ❑ Compilador:
 - MPLAB C18 Lite v3.34;
- ❑ Ambiente Integrado de Desenvolvimento (IDE):
 - MPLAB IDE v8.63;
- ❑ Simulador:
 - Proteus 7.4 (ISIS);
- ❑ Gravador:
 - Tiny Bootloader v1.98;
- ❑ Comunicação Serial:
 - DockLight v1.9;

1. Compilador MPLAB® C18

- Desenvolvido e distribuído pela *MicrochipTechnology*;
- Disponível nas **versões**:
 - Full Version:
 - Vendida por um preço considerado baixo se comparado com compiladores de outros fabricantes;
 - Possibilidade de otimização de código
 - Consiste na utilização de técnicas inteligentes de análise do código;
 - Objetivo de reduzir o número de instruções utilizadas na construção de um bloco de código.
 - Por consequência, diminui a quantidade de memória;
 - Student Edition:
 - Distribuída gratuitamente;
 - A otimização de código só está disponível nos primeiros 60 dias após a instalação.

1. Compilador MPLAB® C18

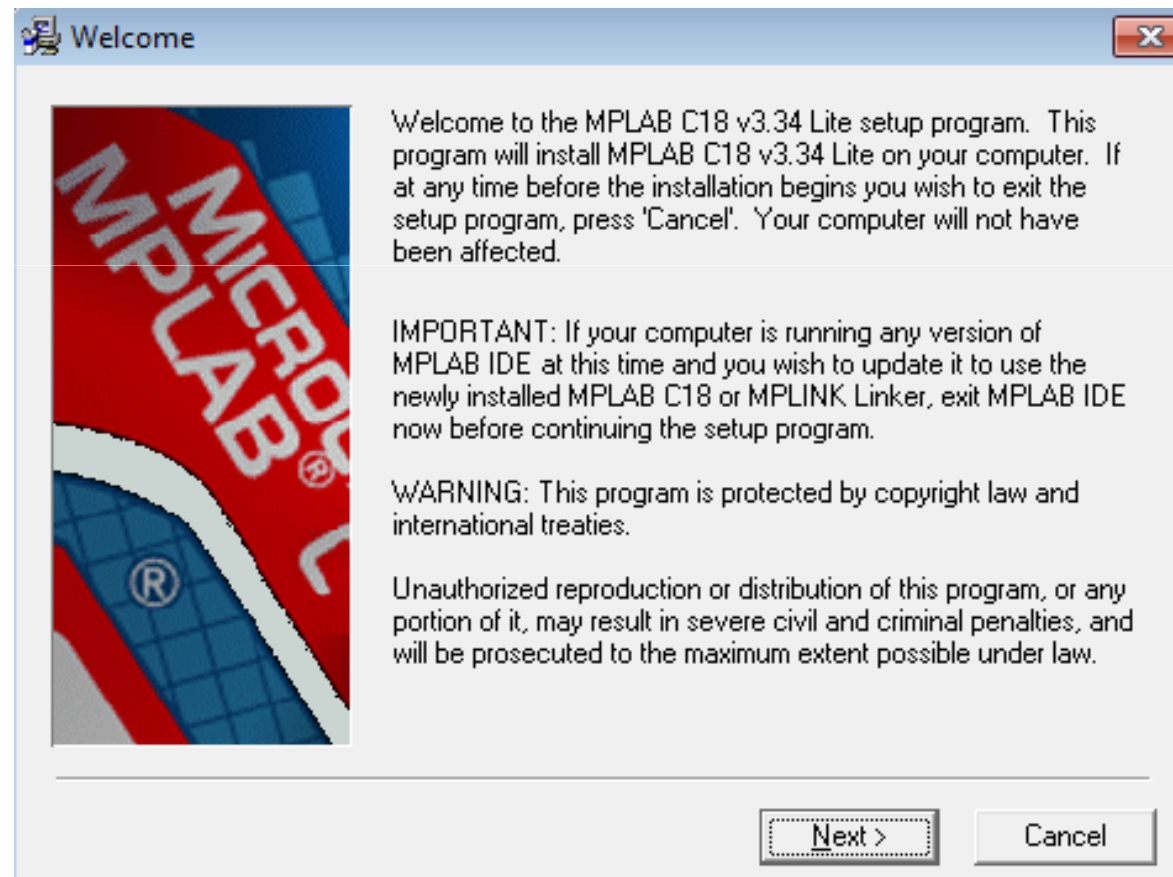
- Tipos de Dados

Type	Size	Minimum	Maximum
<code>char^(1,2)</code>	8 bits	-128	127
<code>signed char</code>	8 bits	-128	127
<code>unsigned char</code>	8 bits	0	255
<code>int</code>	16 bits	-32,768	32,767
<code>unsigned int</code>	16 bits	0	65,535
<code>short</code>	16 bits	-32,768	32,767
<code>unsigned short</code>	16 bits	0	65,535
<code>short long</code>	24 bits	-8,388,608	8,388,607
<code>unsigned short long</code>	24 bits	0	16,777,215
<code>long</code>	32 bits	-2,147,483,648	2,147,483,647
<code>unsigned long</code>	32 bits	0	4,294,967,295

Type	Size	Minimum Exponent	Maximum Exponent	Minimum Normalized	Maximum Normalized
<code>float</code>	32 bits	-126	128	$2^{-126} \approx 1.17549435e - 38$	$2^{128} * (2-2^{-15}) \approx 6.80564693e + 38$
<code>double</code>	32 bits	-126	128	$2^{-126} \approx 1.17549435e - 38$	$2^{128} * (2-2^{-15}) \approx 6.80564693e + 38$

Compilador MPLAB® C18 - Instalação

- 1º Passo: Tela de boas-vindas



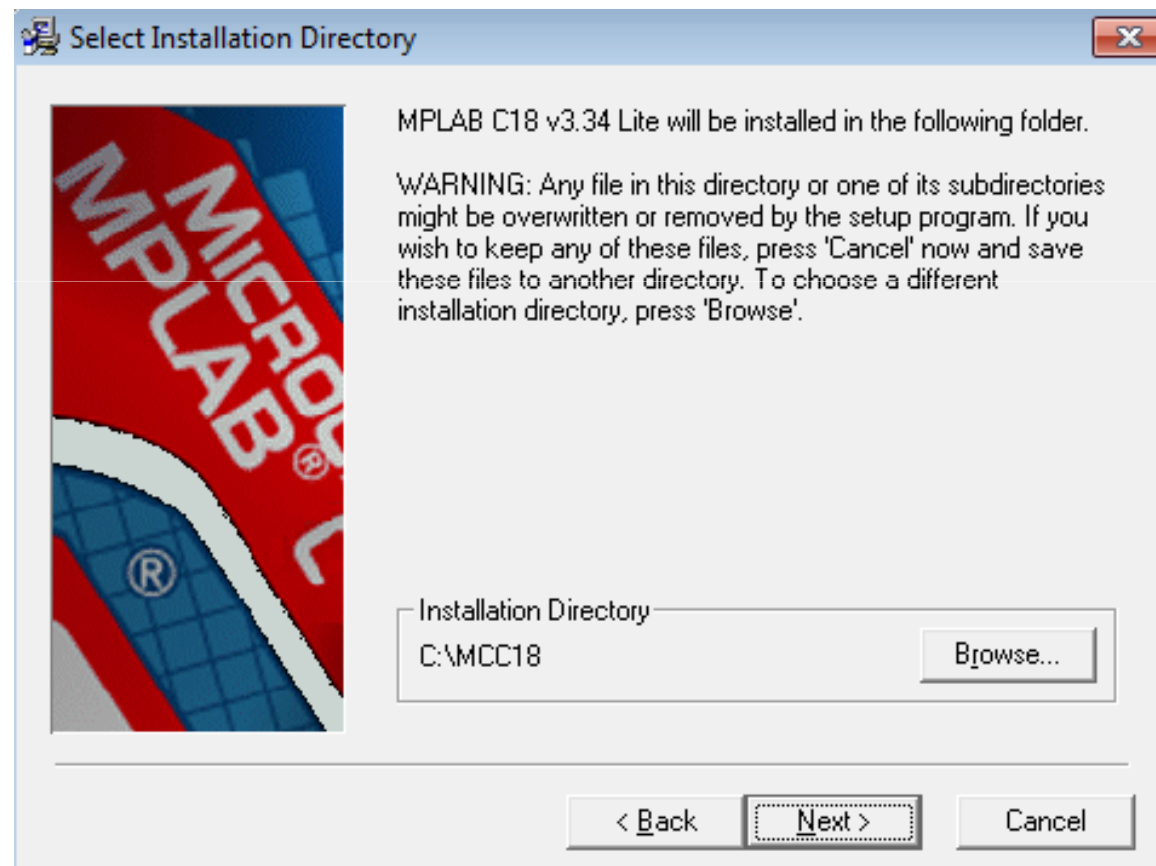
Compilador MPLAB® C18 - Instalação

- 2º Passo: Termo de licença



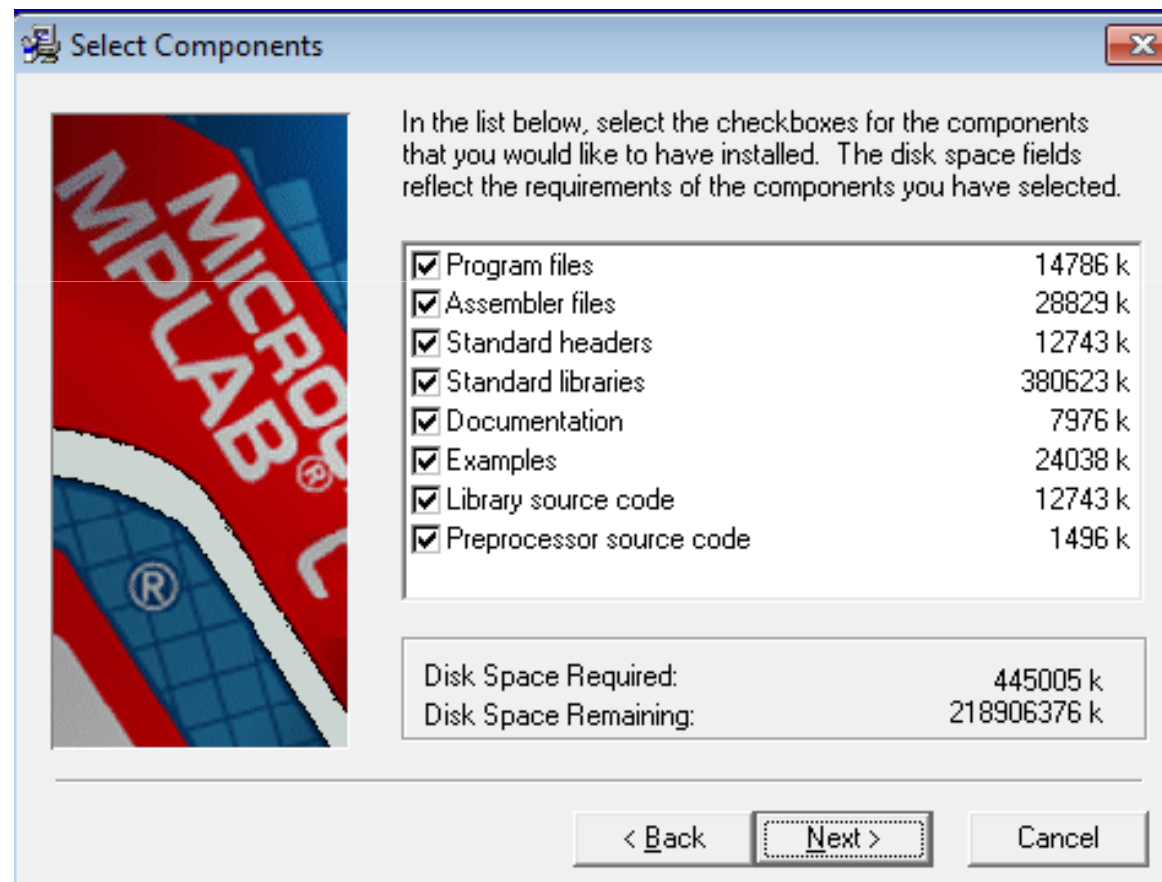
Compilador MPLAB® C18 - Instalação

- 3º Passo: Diretório de instalação



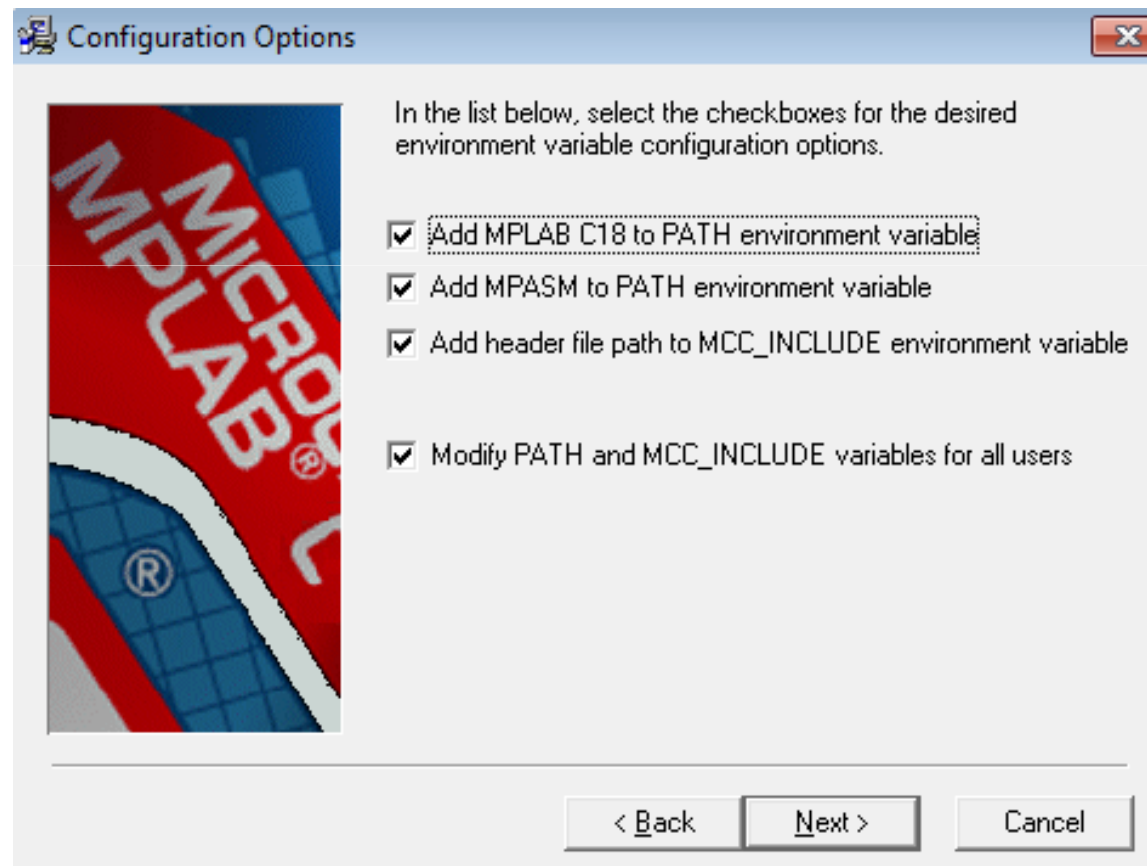
Compilador MPLAB® C18 - Instalação

- 4º Passo: Seleção dos componentes a serem instalados



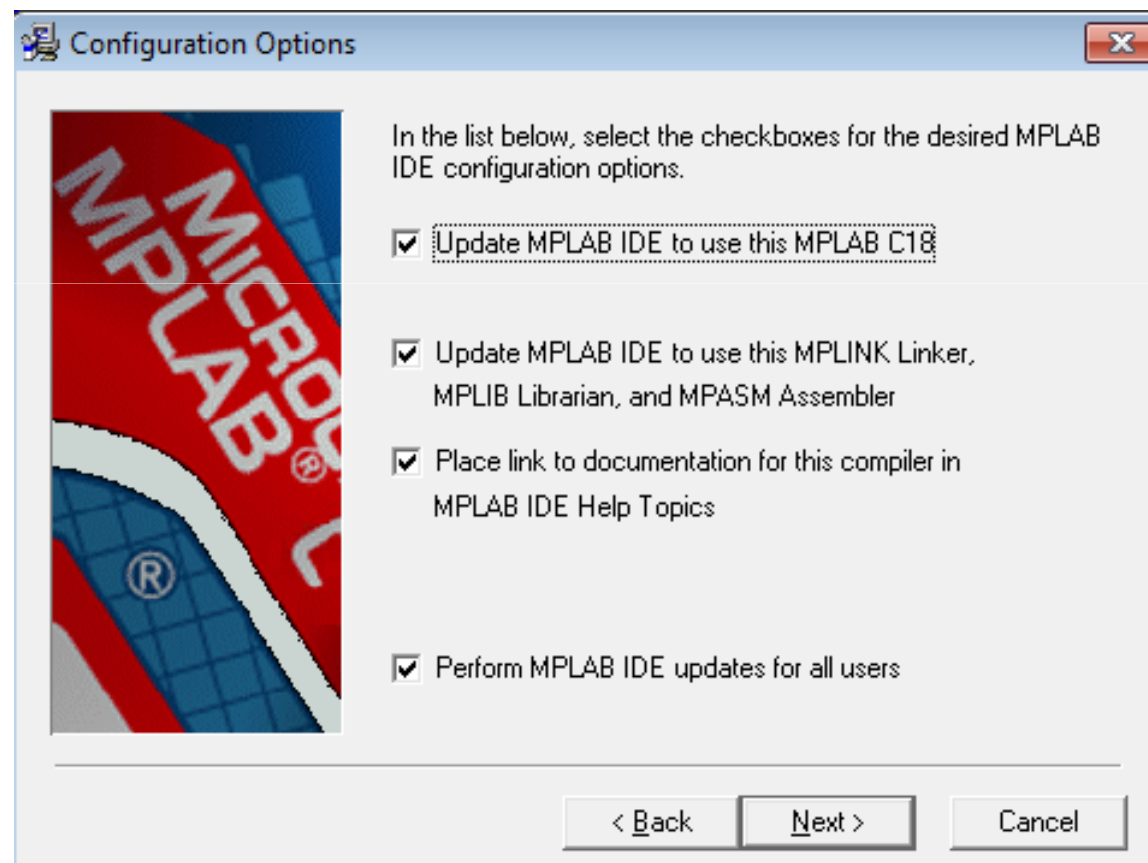
Compilador MPLAB® C18 - Instalação

- 5º Passo: Opção para as variáveis do ambiente



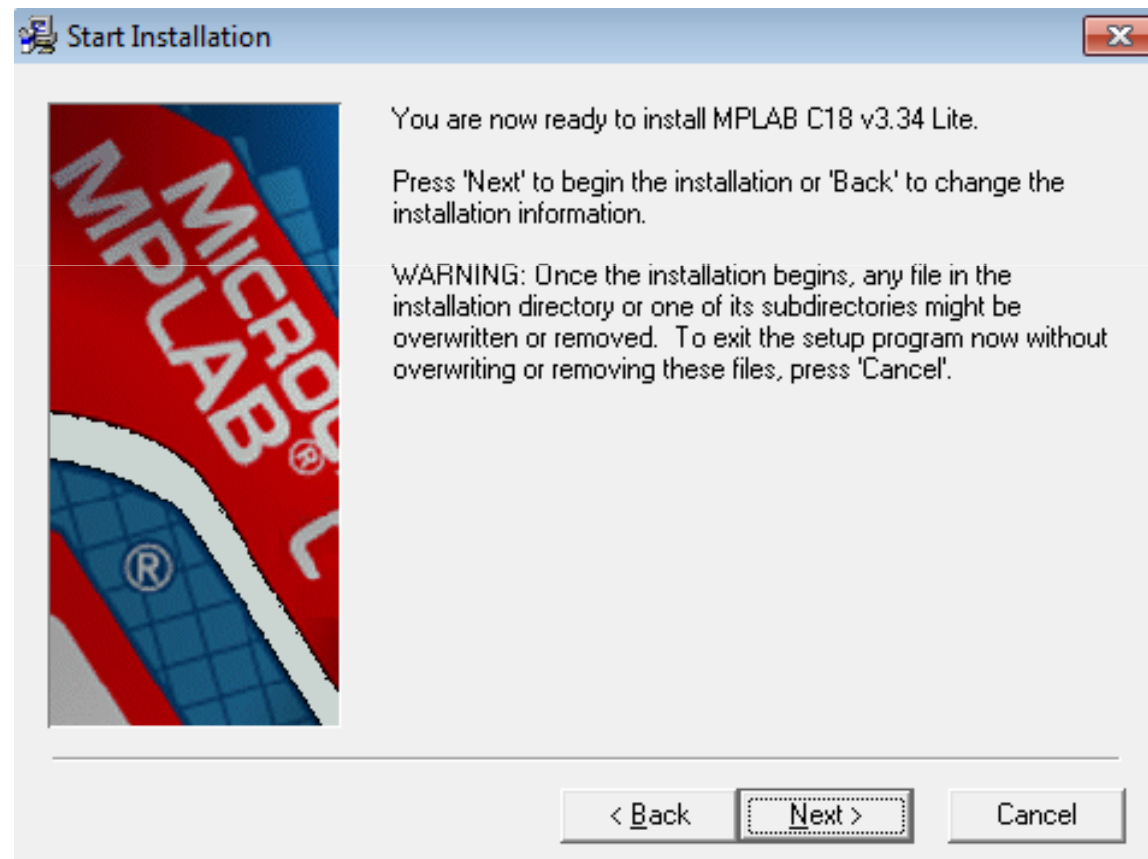
Compilador MPLAB® C18 - Instalação

- 6º Passo: Configurações para o MPLAB IDE



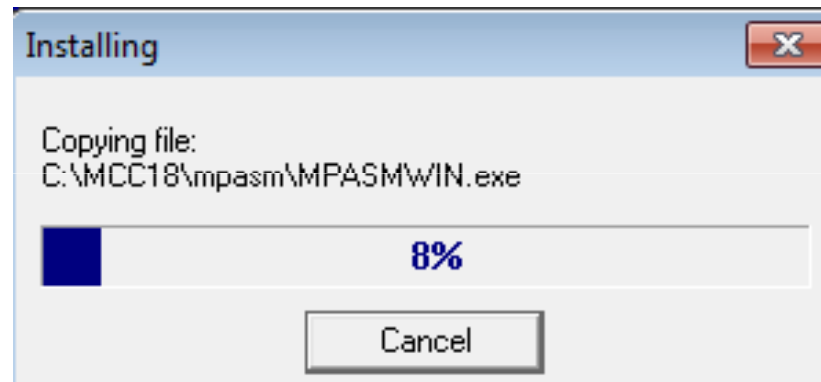
Compilador MPLAB® C18 - Instalação

- 7º Passo: Pronto para a transferência dos arquivos



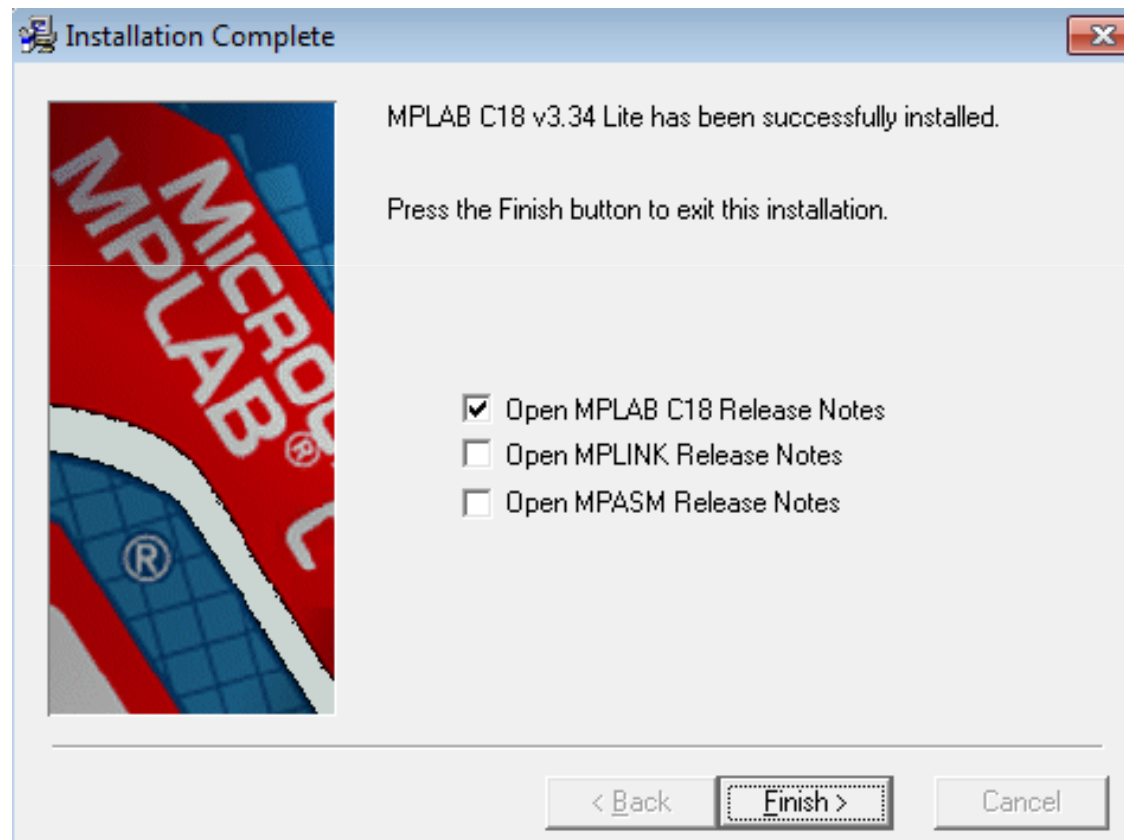
Compilador MPLAB® C18 - Instalação

- 8º Passo: Barra de progresso de instalação



Compilador MPLAB® C18 - Instalação

- 9º Passo: Instalação feita com sucesso

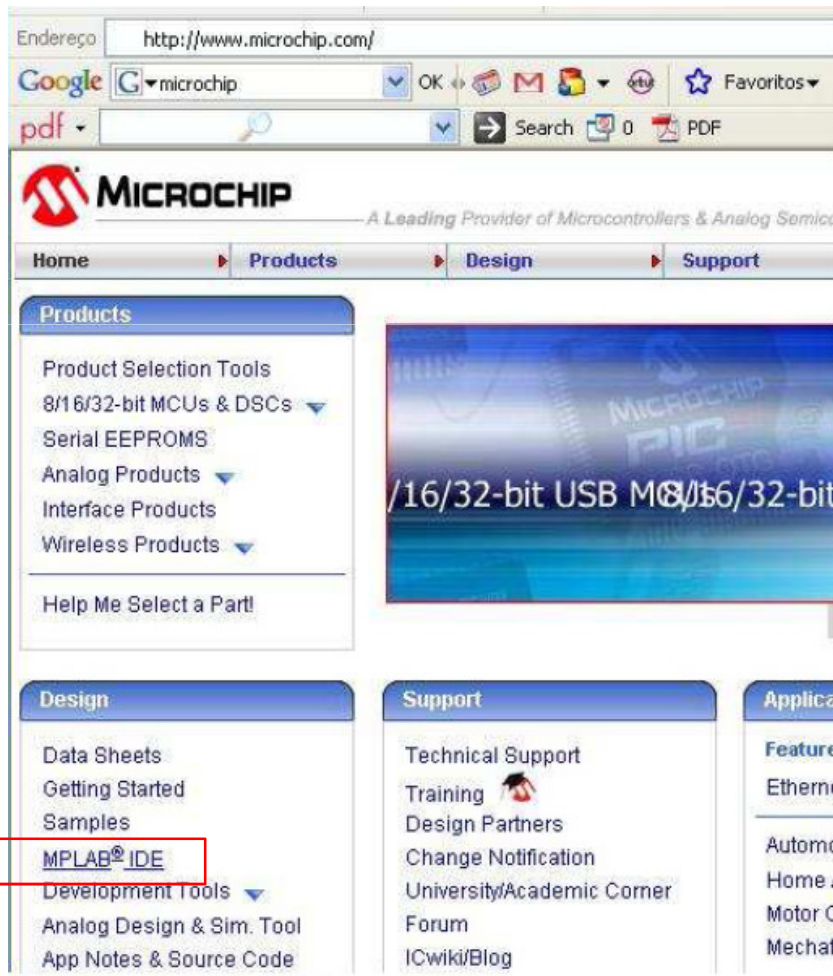








Ambiente Integrado de Desenvolvimento (IDE)

- Permite:
 - Desenvolver programas em:
 - Assembly
 - C
 - Simulação
 - Depuração
 - Microchip MPLAB IDE
 - Disponível em <http://www.microchip.com>

MPLAB IDE

- 1º Passo: Download



MPLAB Software			
Downloads	Associated Files and Release Notes		
MPLAB IDE v8.10 Full Release Zipped Installation	Release Notes for MPLAB IDE v8.10		
MPLAB IDE v6.14 Interim Release	Release Notes for MPLAB IDE v6.14		
Memory Starter Kit Patch for v8.14	Information on v8.14 Memory Kit Patch		
Downloads			
Title	Date Published	Size	DL
Labcenter's Proteus VSM Demo Plug-In for MPLAB IDE	4/27/2008 5:29:08 PM	31411 KB	
MPASM/MPLINK User's Guide	4/27/2008 2:01:16 PM	2628 KB	
MPLAB Assembler, Linker and Utilities for PIC24 MCUs and dsPIC DSCs User's Guide	4/27/2008 5:29:08 PM	3311 KB	
MPLAB IDE User's Guide	4/27/2008 4:08:23 PM	3842 KB	
MPLAB v8.14 Release Note Set	7/17/2008 10:24:23 AM	254 KB	
Patch for MPLAB PM3 with 25AA020A Devices	5/7/2008 2:51:41 PM	1 KB	

2. MPLAB IDE

- 2º Passo: Tela de boas-vindas



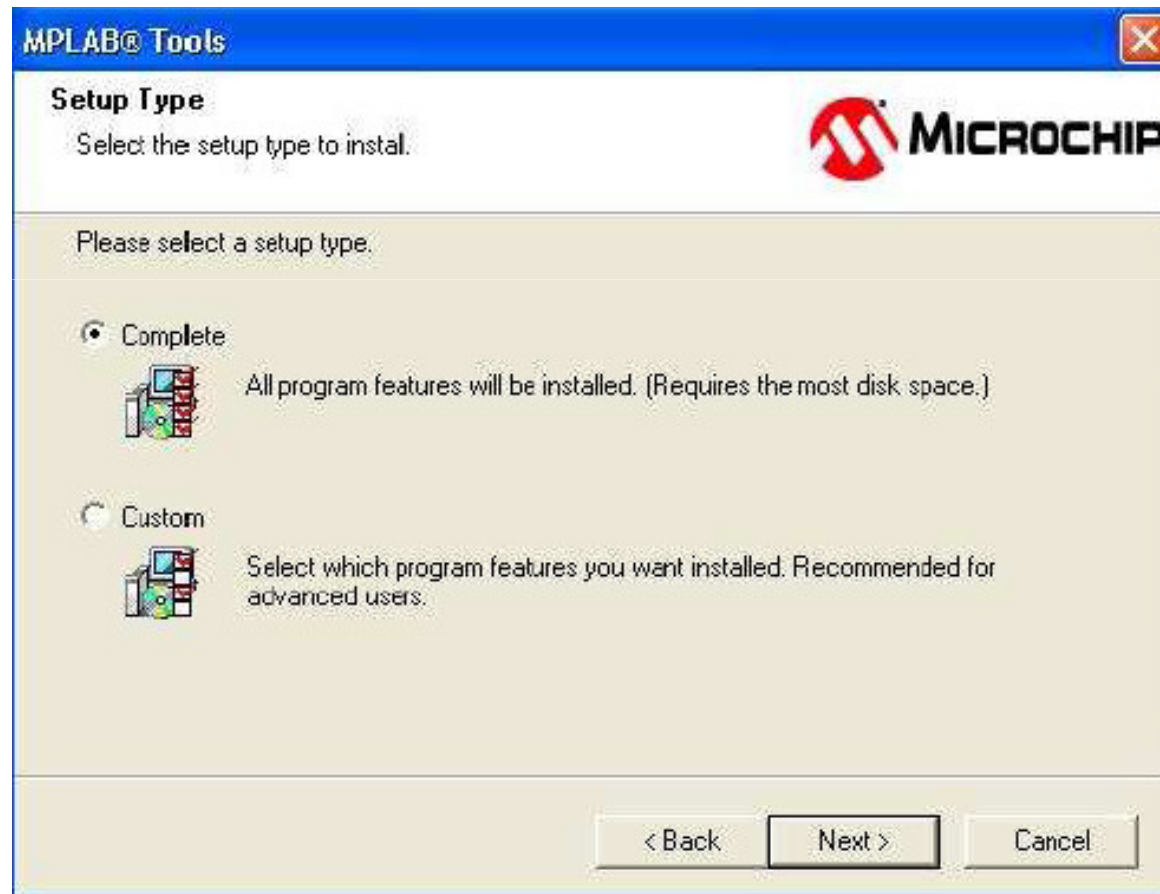
2. MPLAB IDE

- 3º Passo: Termo de licença



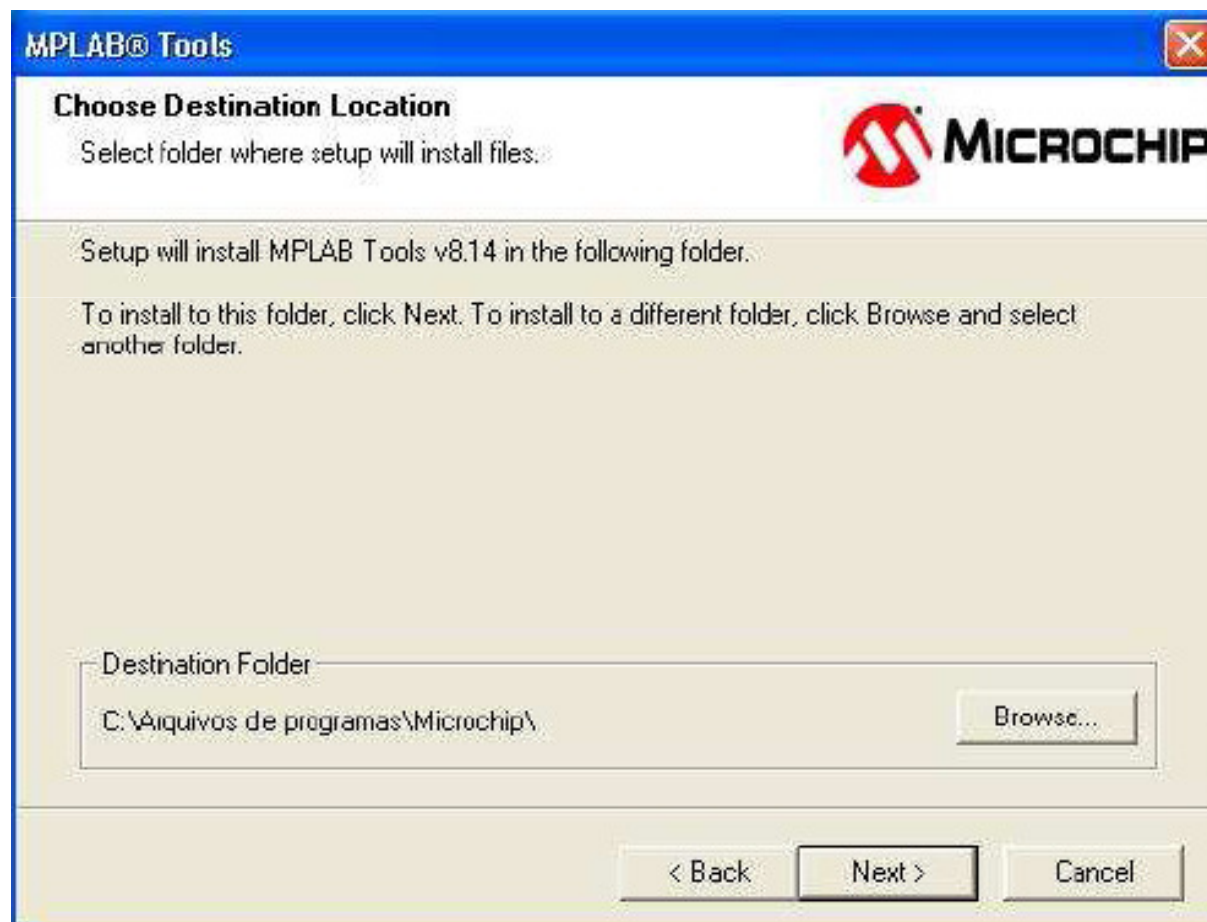
2. MPLAB IDE

- 4º Passo: Tipo de instalação



2. MPLAB IDE

- 5º Passo: Diretório de instalação



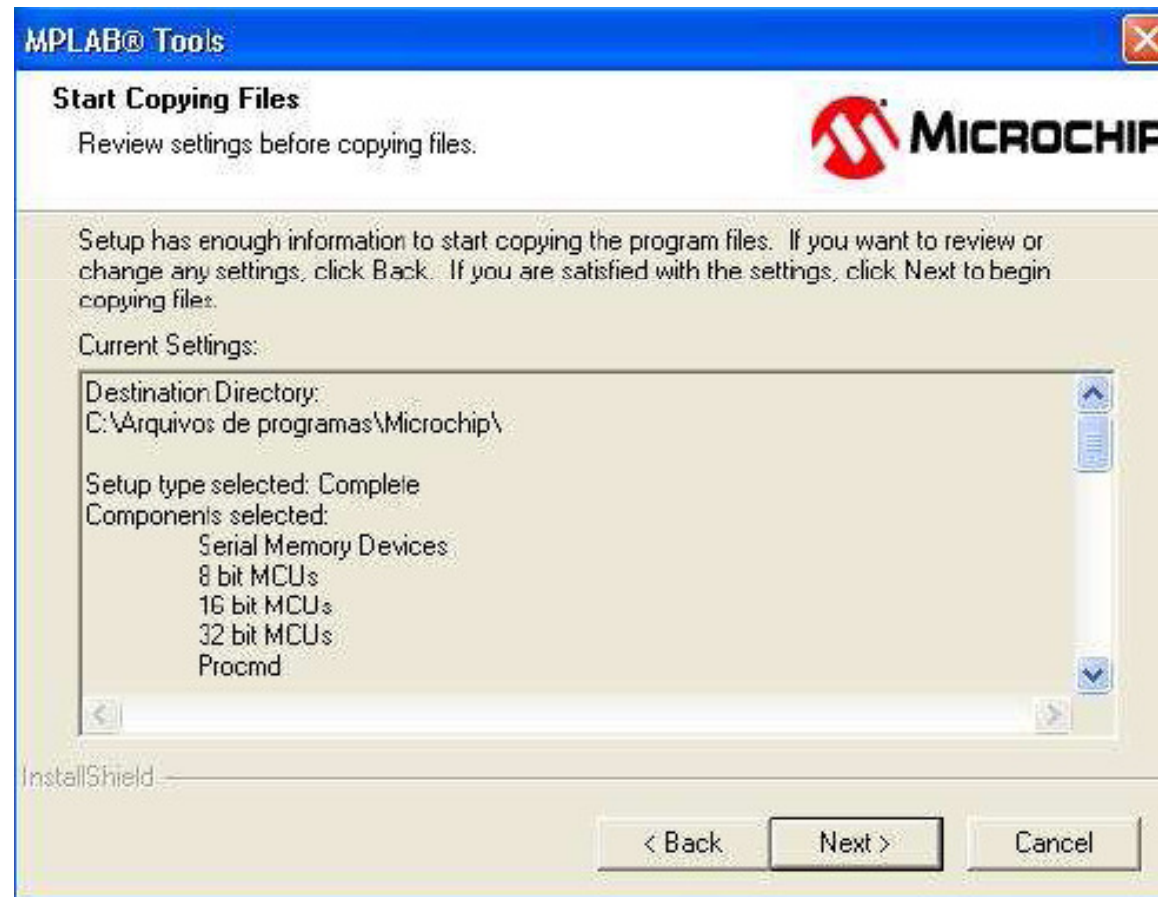
2. MPLAB IDE

- 6º Passo: Aplicação da licença



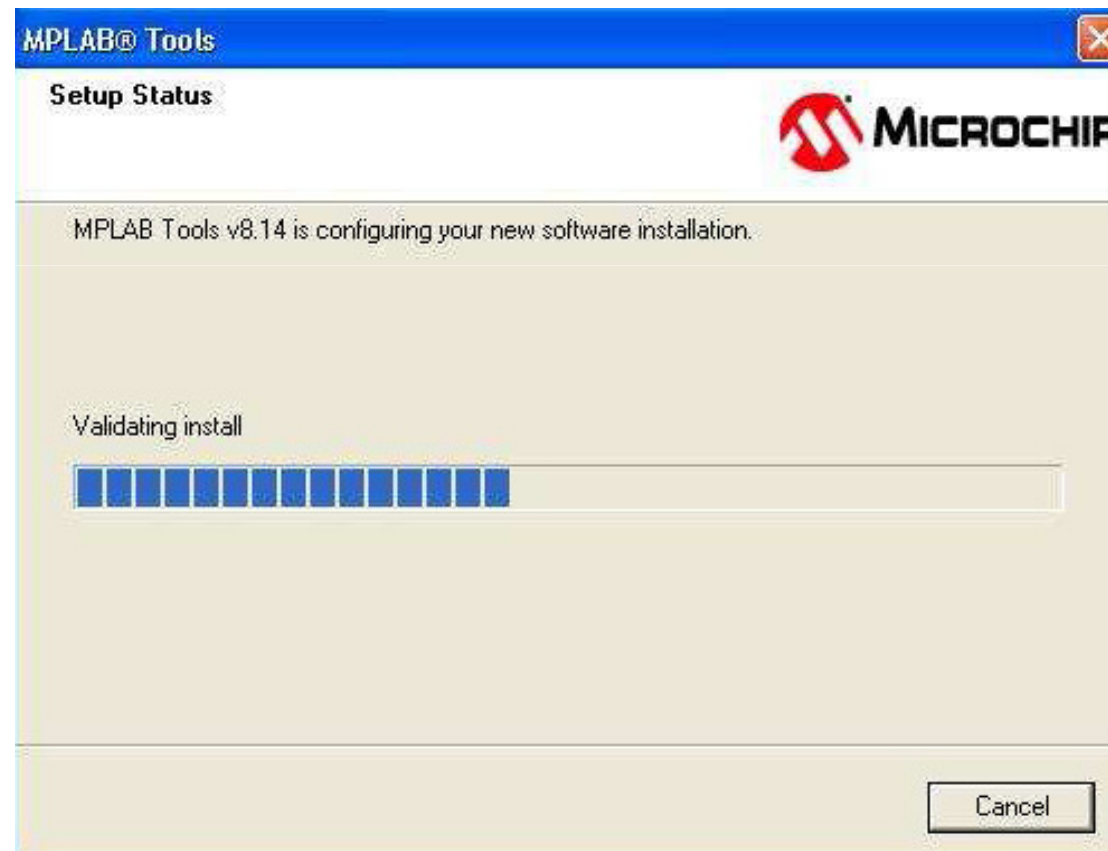
2. MPLAB IDE

- 7º Passo: Início da cópia dos arquivos



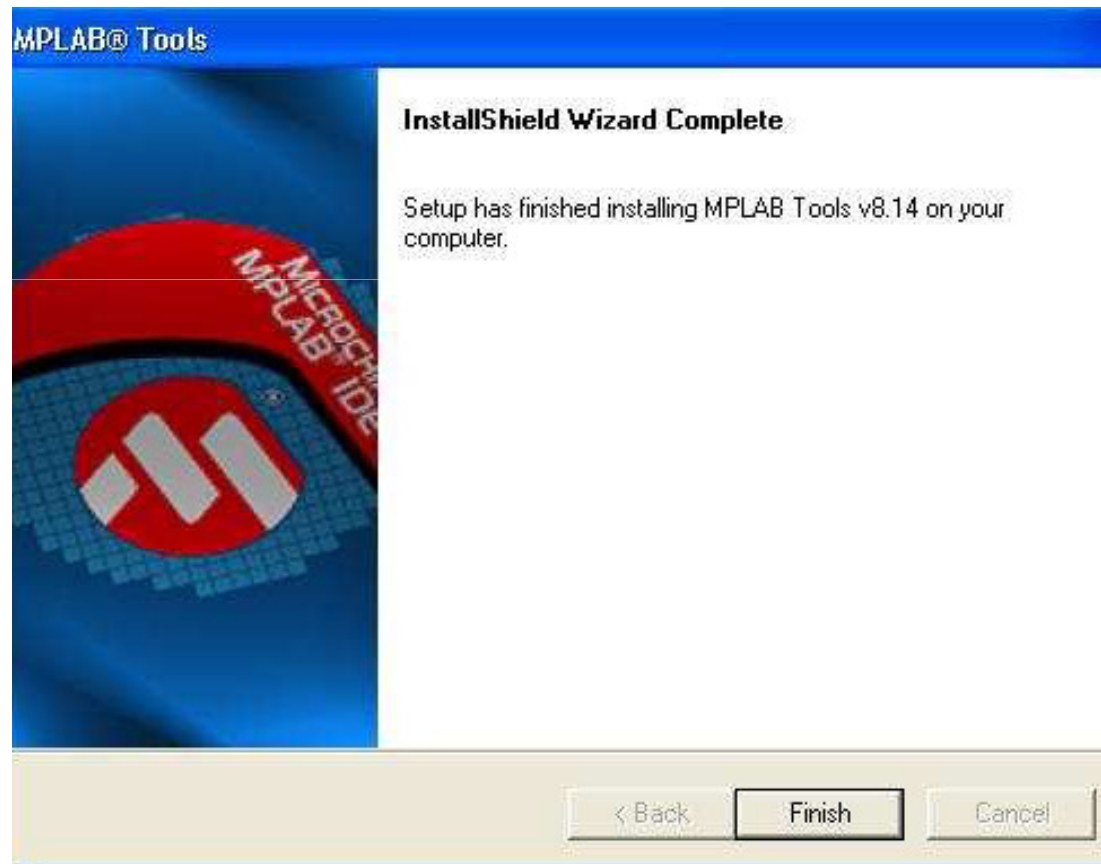
2. MPLAB IDE

- 8º Passo: Barra de progresso de instalação



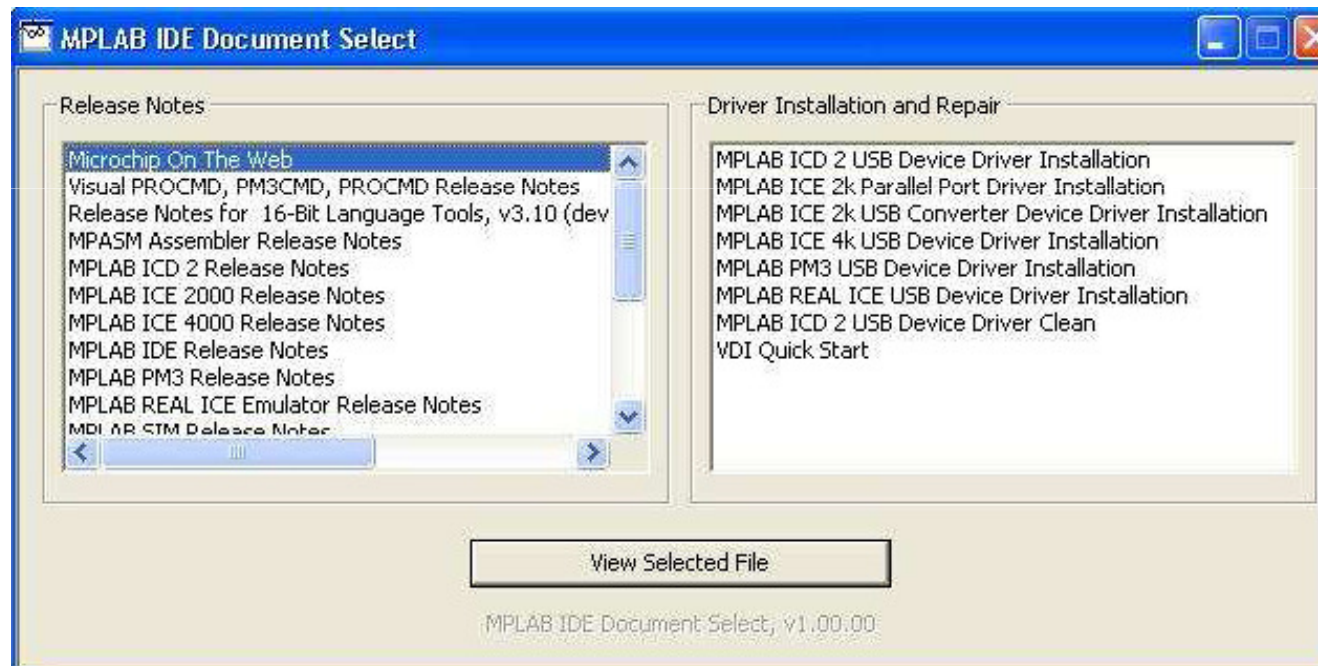
2. MPLAB IDE

- 9º Passo: Instalação feita com sucesso



2. MPLAB IDE

- **10º Passo:** A janela abaixo será aberta para a consulta de arquivos. Clique em “X” para fechá-la:



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

1º Passo: Criação de um **arquivo-fonte** em C

1. Criar uma pasta chamada “fontesC” no diretório raiz;
2. salvar o arquivo “programa1.c”;

```

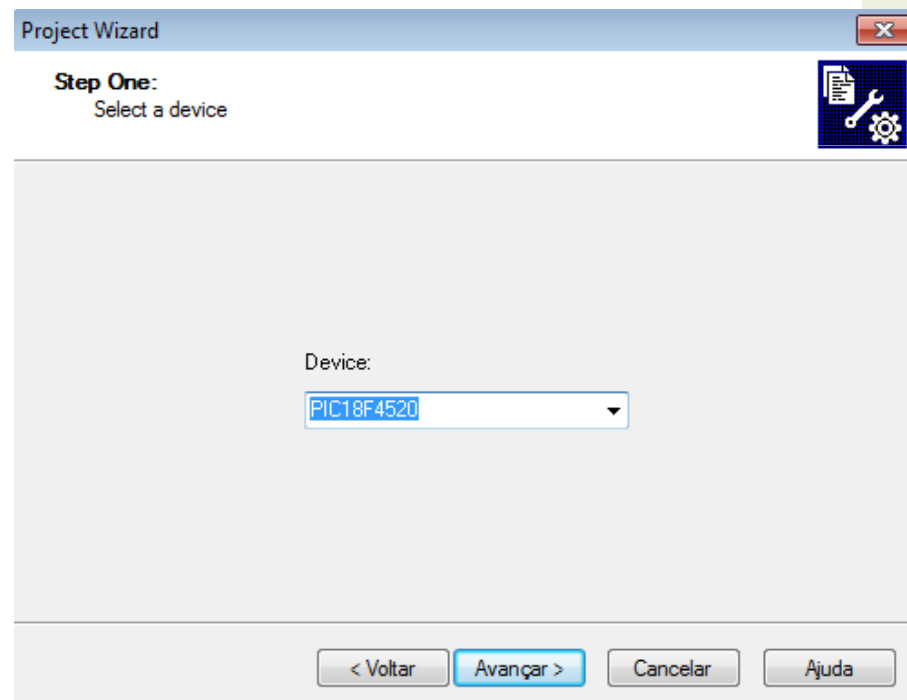
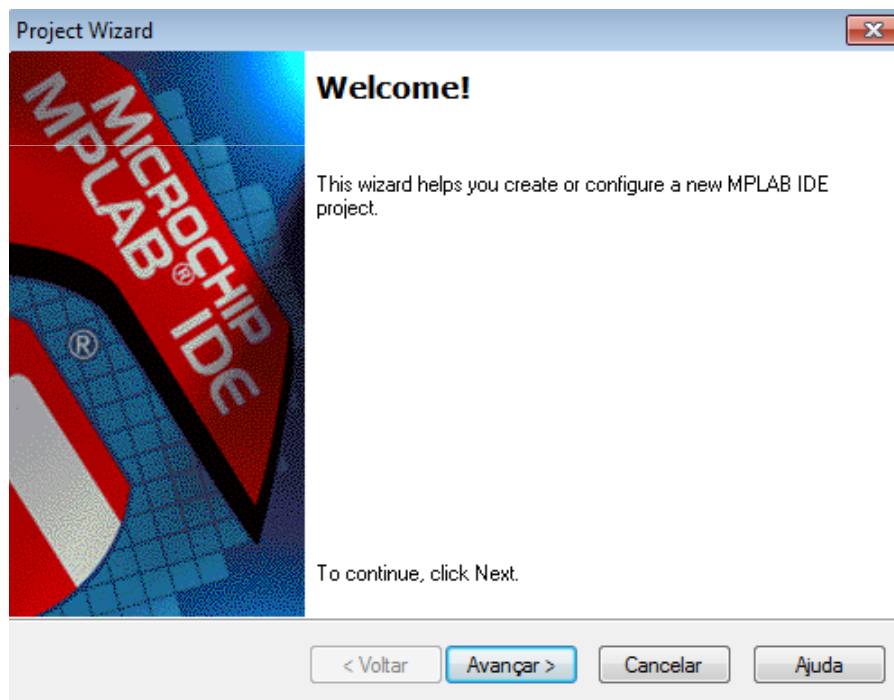
//*****
//*Meu primeiro programa em C.
//*****
#include <p18f4520.h>           //arquivo cabeçalho padrão do PIC18F4520
//*****
void main (void)              //função main
{
    TRISA = 0x02;              //pino RA1 entrada e demais pinos do PORTA saída
    TRISB = 0x00;              //PORTB saída
    TRISC = 0x00;              //PORTC saída
    TRISD = 0x00;              //PORTD saída
    TRISE = 0x00;              //PORTE saída
    ADCON1 = 0x0F;             //configura os pinos dos PORTA e PORTE como digitais
    PORTA = 0;                 //limpa PORTA
    PORTB = 0;                 //limpa PORTB
    PORTC = 0;                 //limpa PORTC
    PORTD = 0;                 //limpa PORTD
    PORTE = 0;                 //limpa PORTE
    while (1);                 //loop infinito
//*****
}

```

2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o Project Wizard

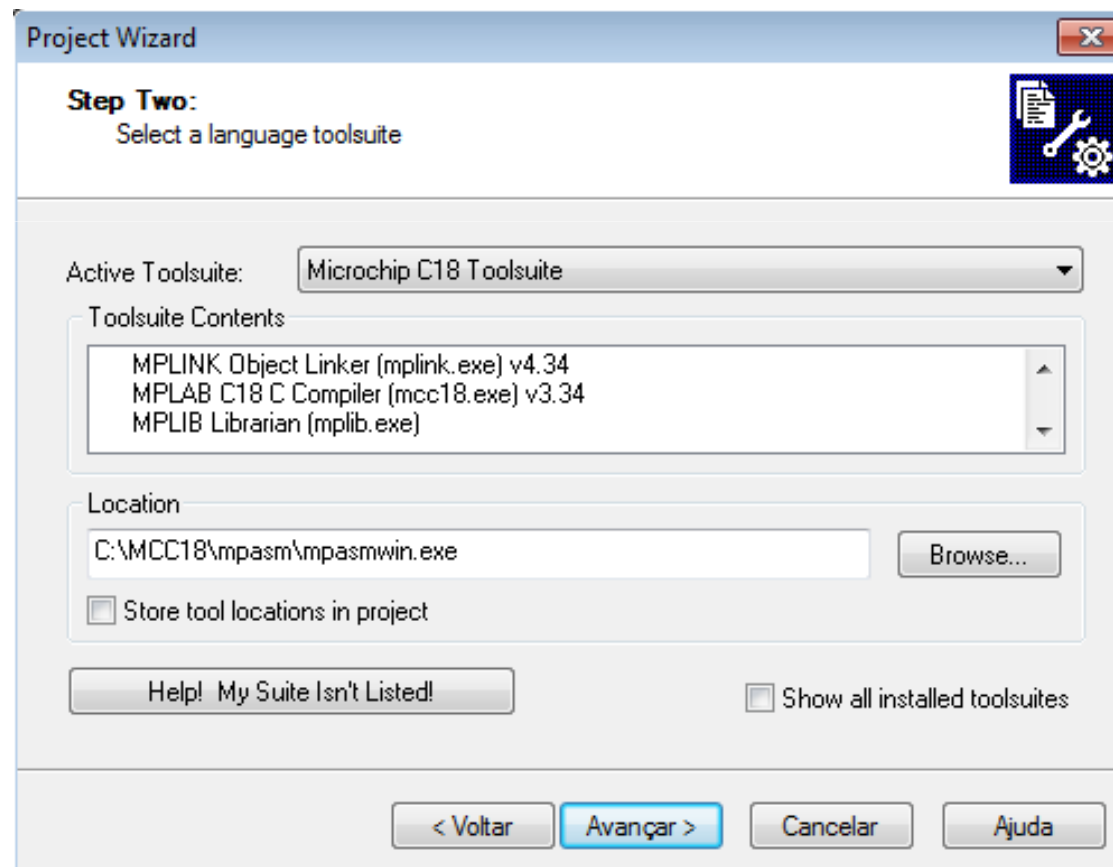
- A. Clicar no menu **Project/Project Wizard**;
- B. Escolher o **microcontrolador** que será utilizado (**PIC18F4520**);



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o **Project Wizard**

C. Seleção da ferramenta utilizada na compilação (**Microchip C18 Toolsuite**);



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o **Project Wizard**

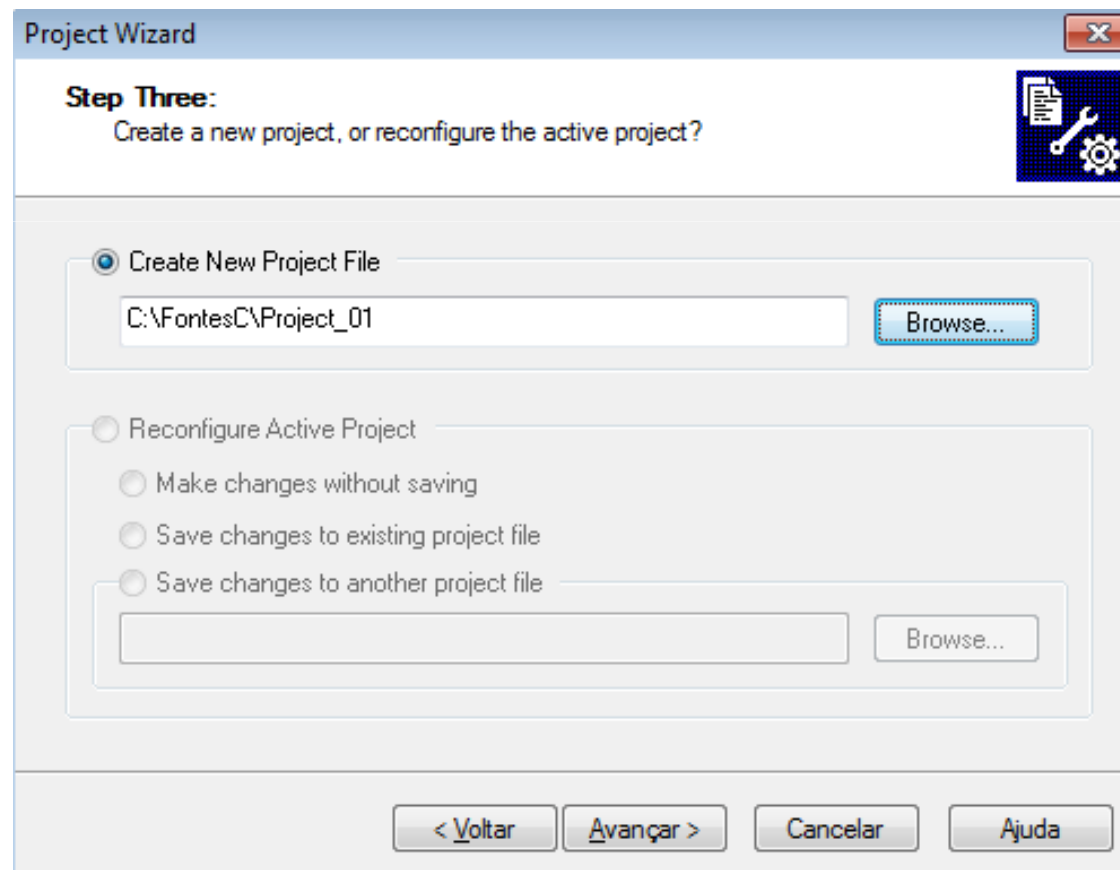
D. **Localização** dos programas utilizados na compilação;

Arquivos	Caminho
MPASMWIN.exe	C:\MCC18\mpasm\MPASMWIM.exe
mplink.exe	C:\MCC18\bin\mplink.exe
mcc18.exe	C:\MCC18\bin\mcc18.exe
mplib.exe	C:\MCC18\bin\mplib.exe

2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o **Project Wizard**

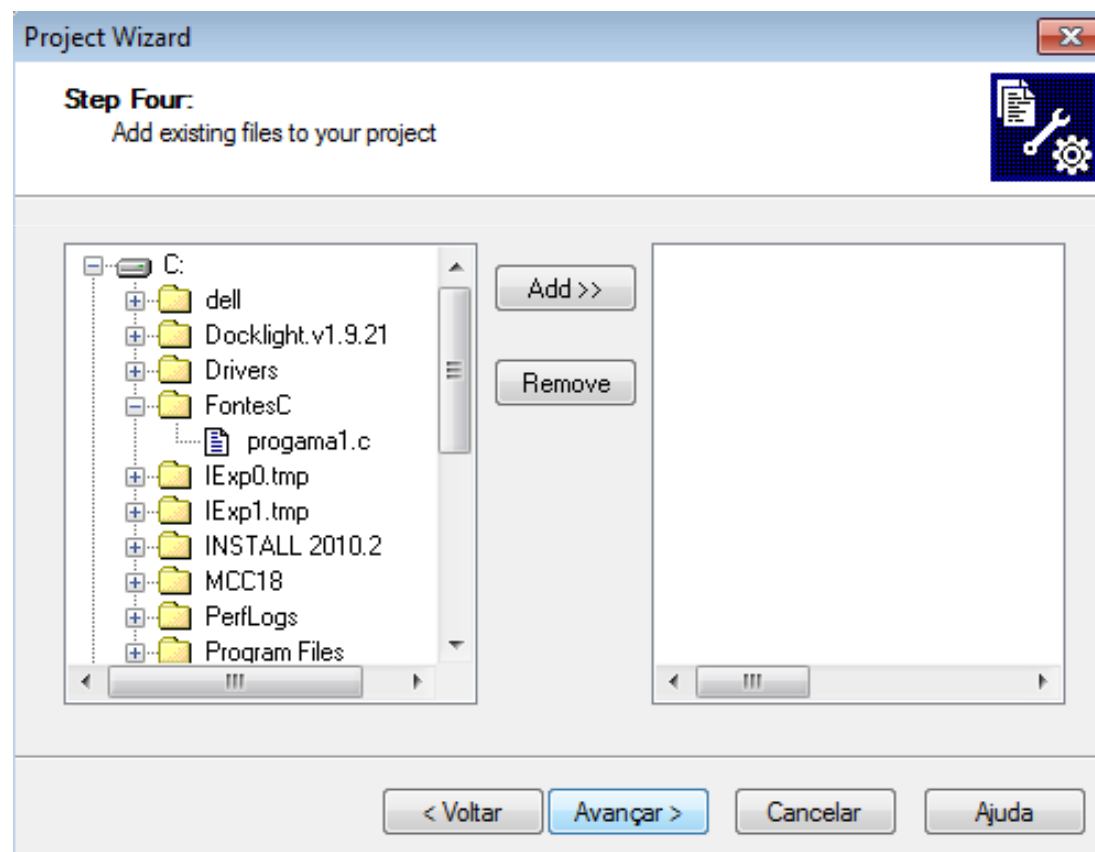
E. **Nome e localização** do projeto (diretório **C:\FontesC**);



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o **Project Wizard**

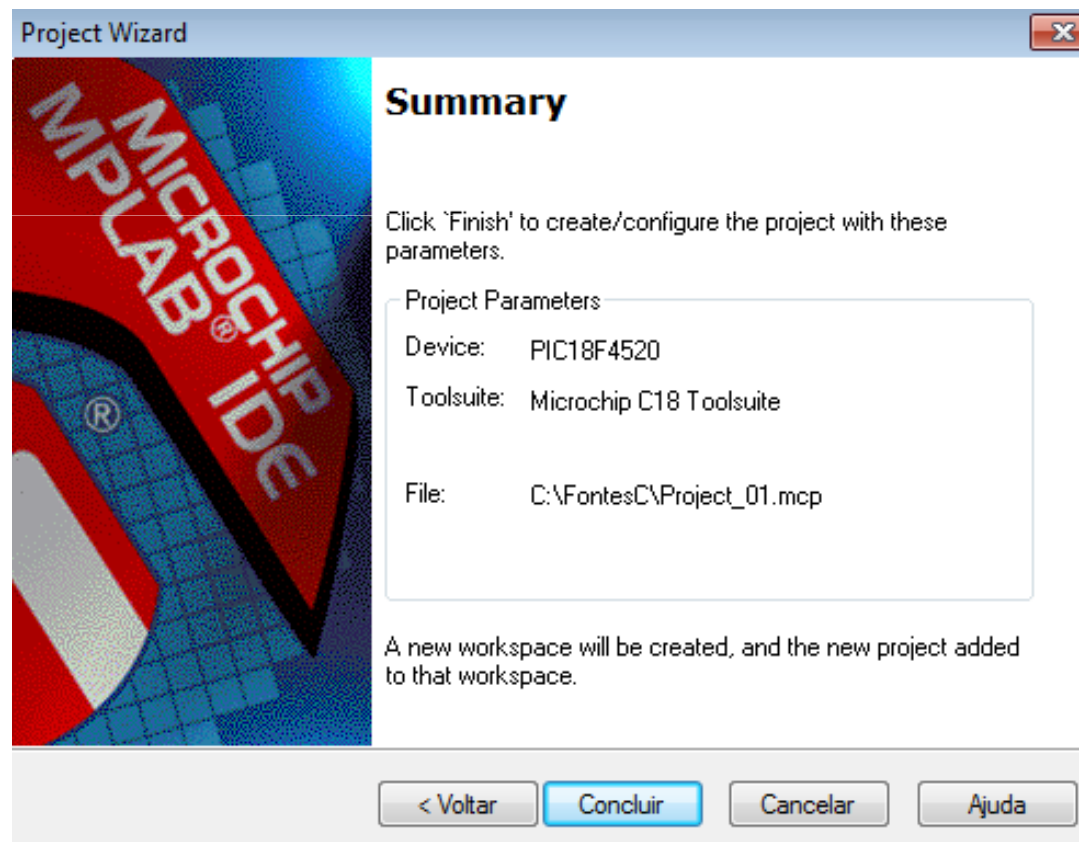
F. **Associar** o programa-fonte ao projeto;



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

2º Passo: Criação de um projeto com o **Project Wizard**

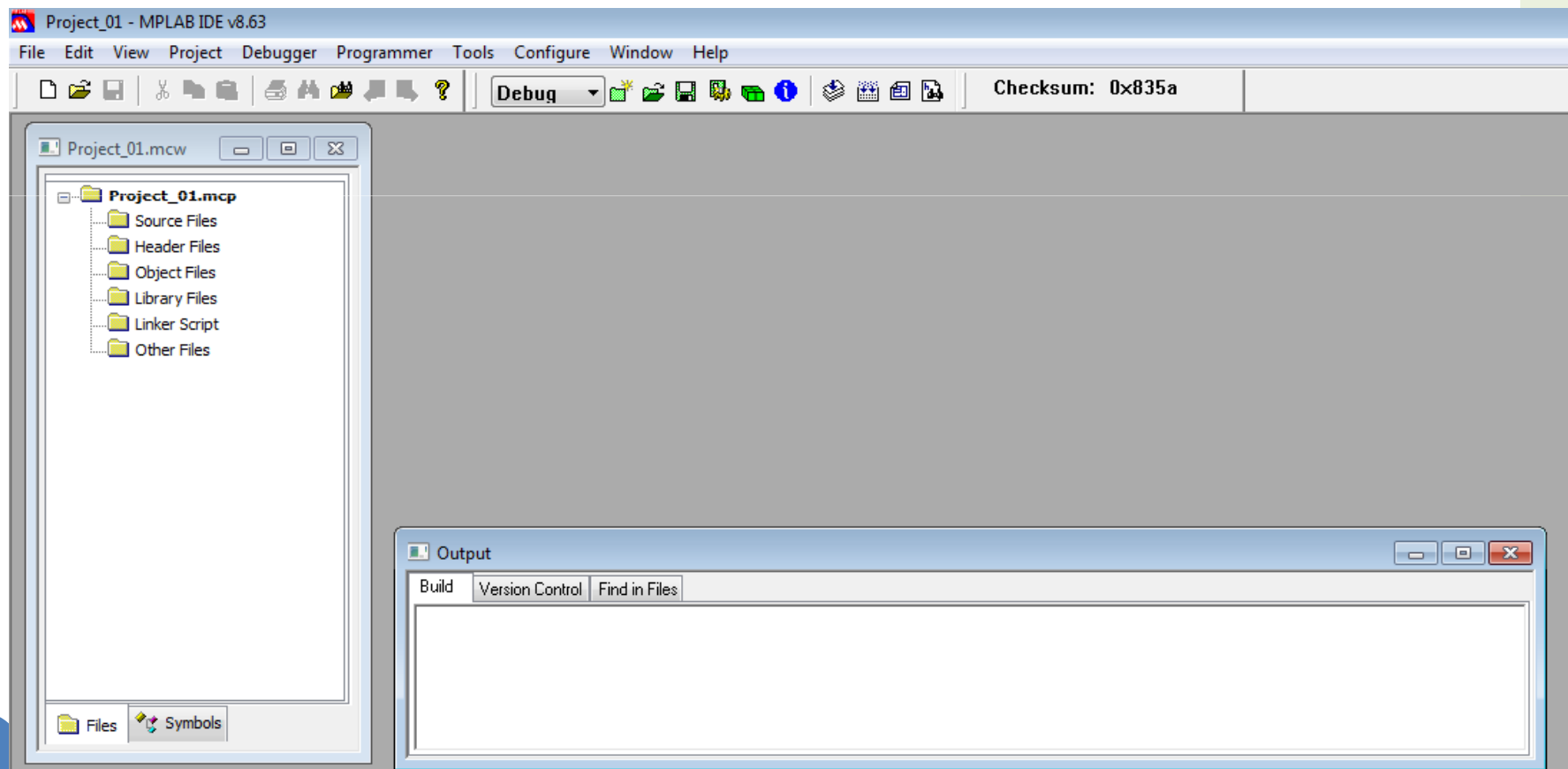
G. **Sumário** de criação do projeto;



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

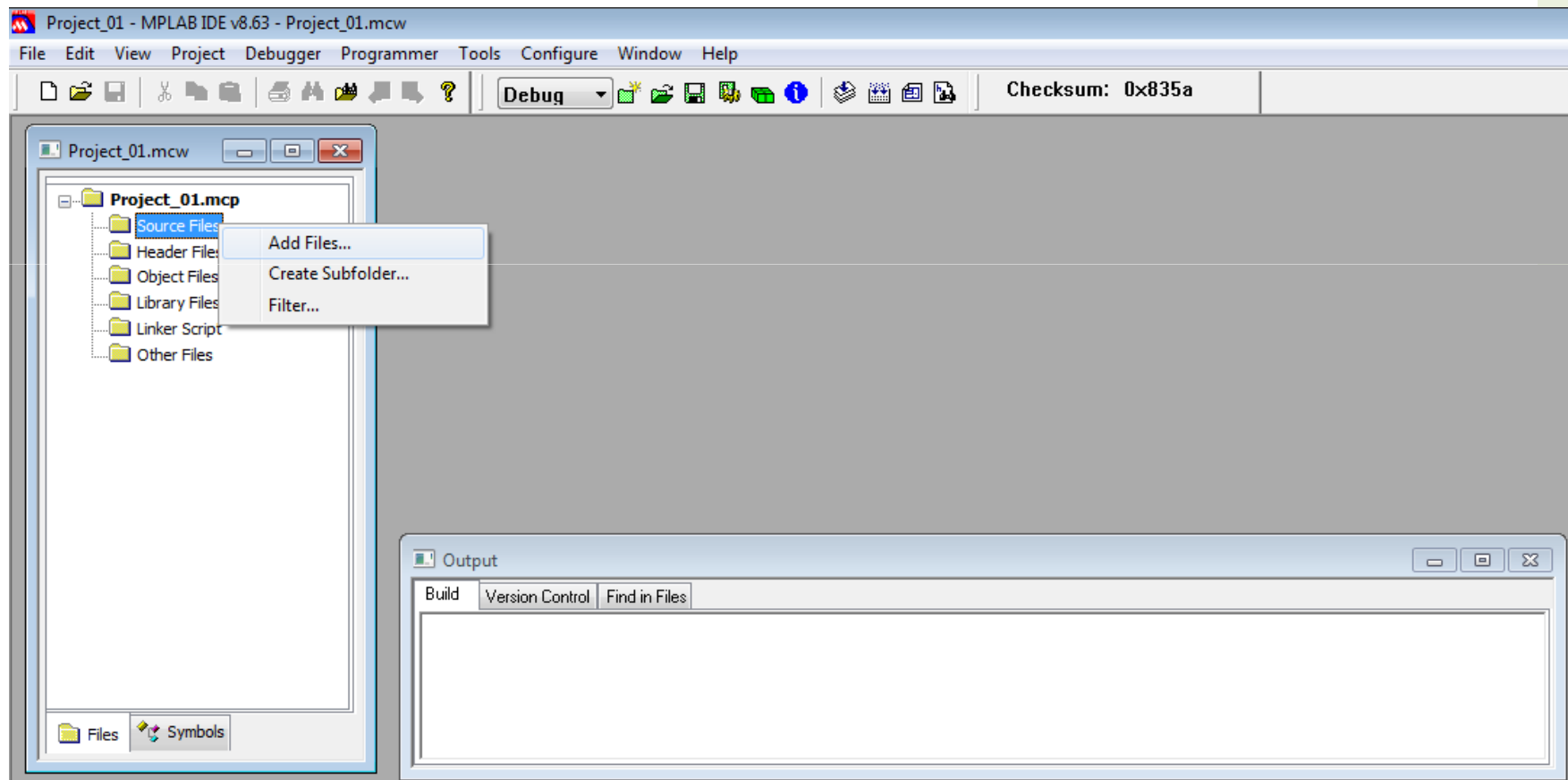
2º Passo: Criação de um projeto com o **Project Wizard**

H. **Tela principal** do MPLAB IDE;



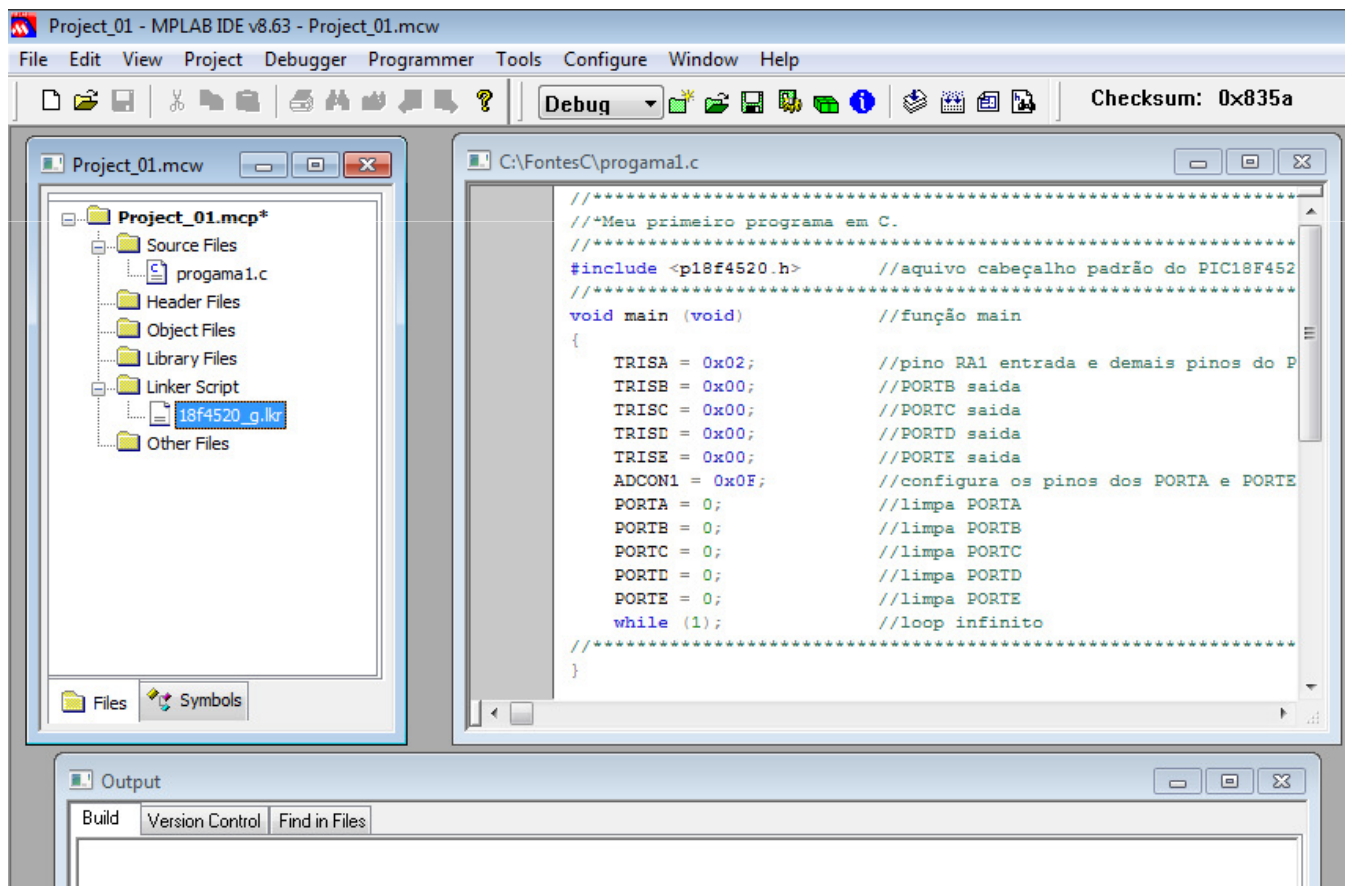
2. Criação de projeto no MPLAB® IDE para o MPLAB C18

3º Passo: Adicionar o arquivo-fonte “programa1.c” ao projeto



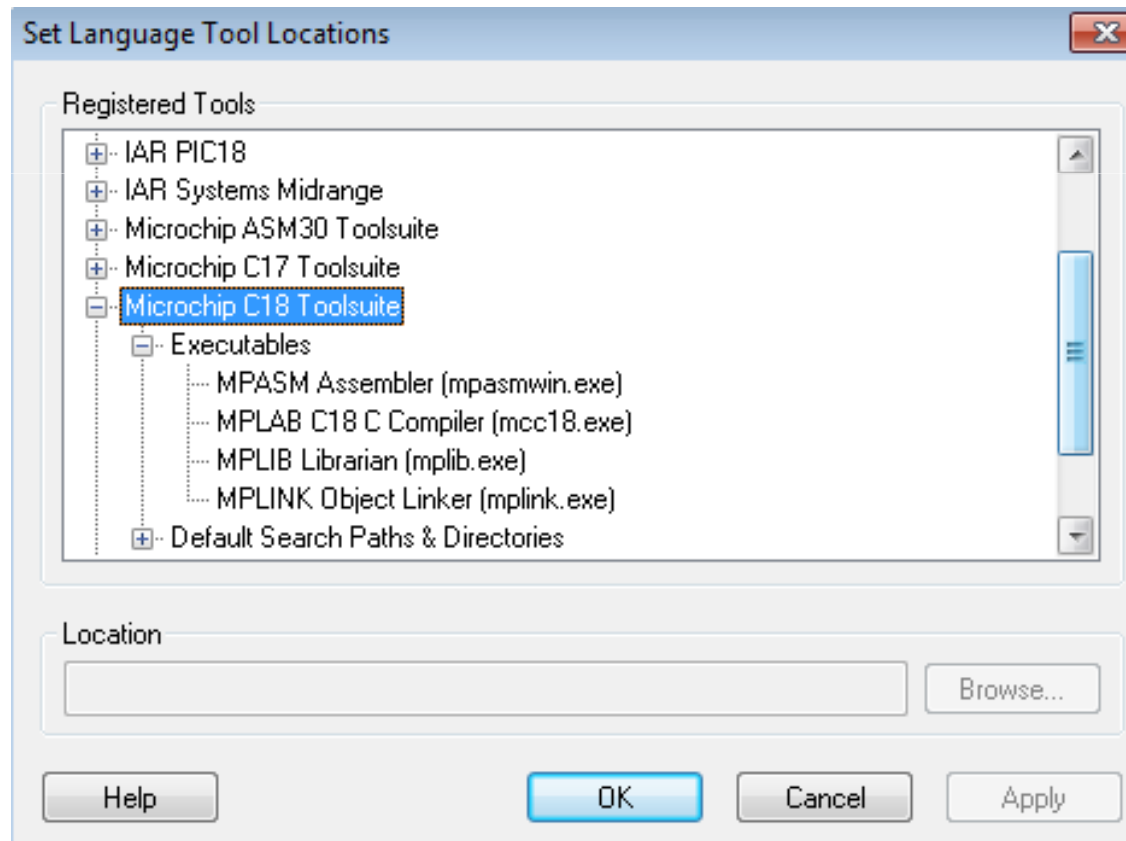
2. Criação de projeto no MPLAB® IDE para o MPLAB C18

3º Passo: Adicionar o arquivo “18F4520_g.lkr” ao projeto;
Diretório C:\MCC18\bin\LKR\18F4520_g.lkr



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

4º Passo: Configurar a **localização** da ferramenta de compilação:
Project\Set Language Tool Locations
Confira a **localização** de cada um dos arquivos executáveis;

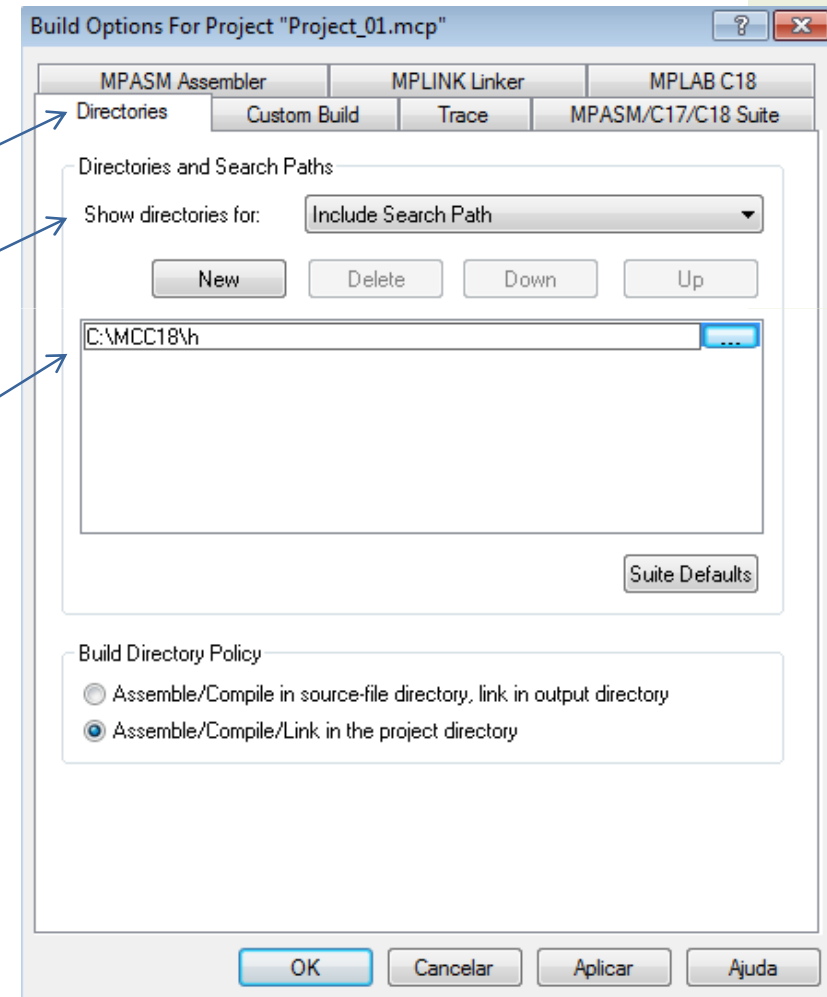


2. Criação de projeto no MPLAB® IDE para o MPLAB C18

5º Passo: Configurar os **diretórios de procura** da ferramenta de compilação

B. Caminho de procura dos **arquivos cabeçalho**.

- Clicar no menu **Project\BuildOption\Project** e selecione a guia **Directories**;
- No Item **"Show Directories for"** selecione a opção **"Include Search Path"**;
- Certifique-se de que caminho **C:\MCC18\h** aparece na janela baixo;
- Senão, clique em **New** e, em seguida no botão **...** e indique **C:\MCC18\h**

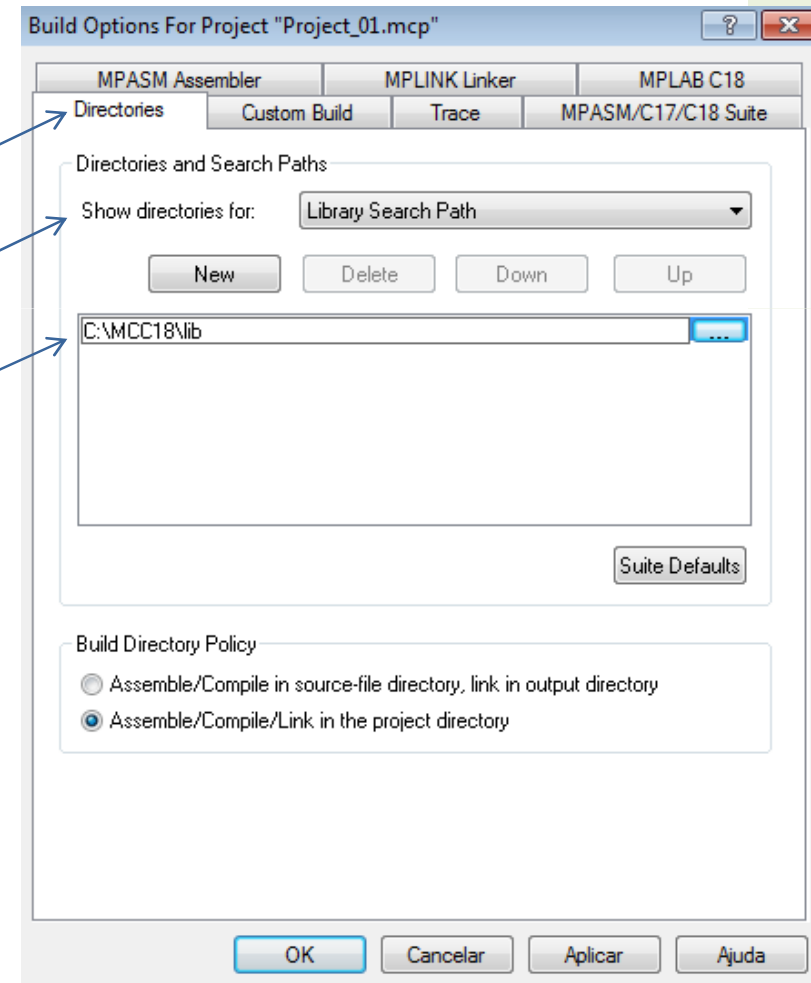


2. Criação de projeto no MPLAB® IDE para o MPLAB C18

5º Passo: Configurar os **diretórios de procura** da ferramenta de compilação

B. Caminho de procura dos **arquivos biblioteca**.

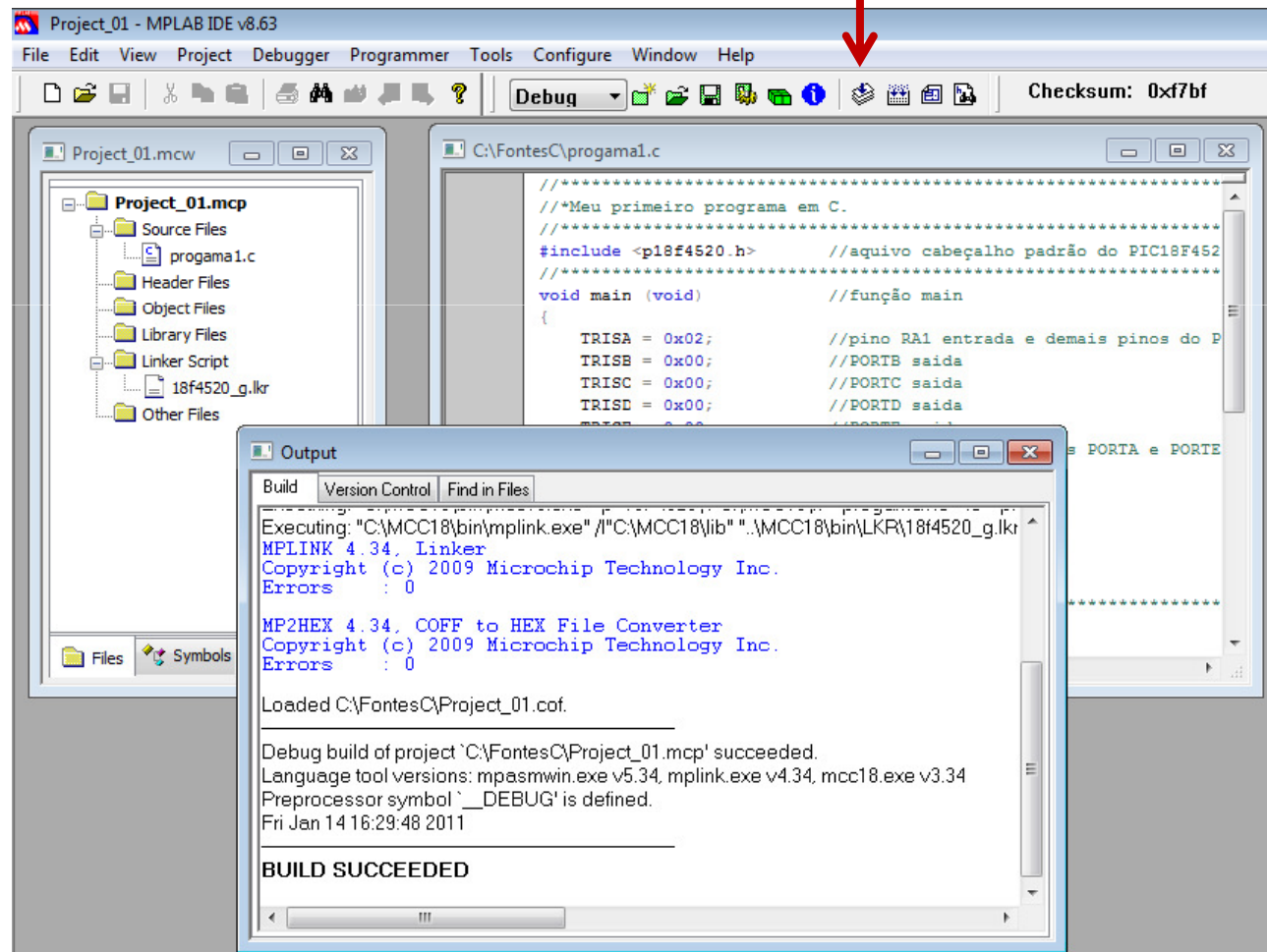
- Clicar no menu **Project\BuildOption\Project** e selecione a guia **Directories**;
- No Item “**Show Directories for**” selecione a opção “**Library Search path**”;
- Certifique-se de que caminho **C:\MCC18\lib** aparece na janela baixo;
- Senão, clique em **New** e, em seguida no botão **...** e indique **C:\MCC18\lib**



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

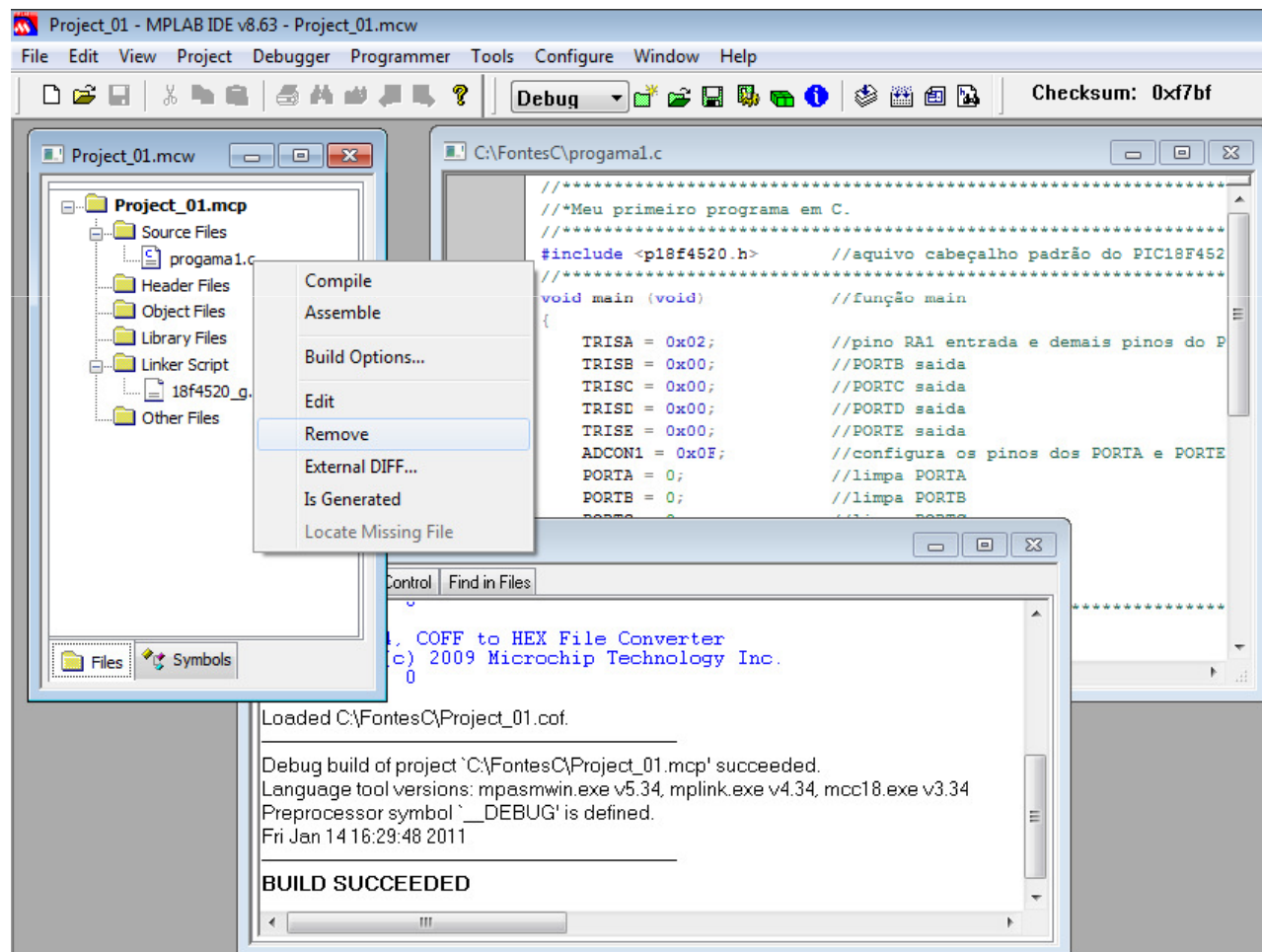
6º Passo: **Compilar** o projeto

Build All



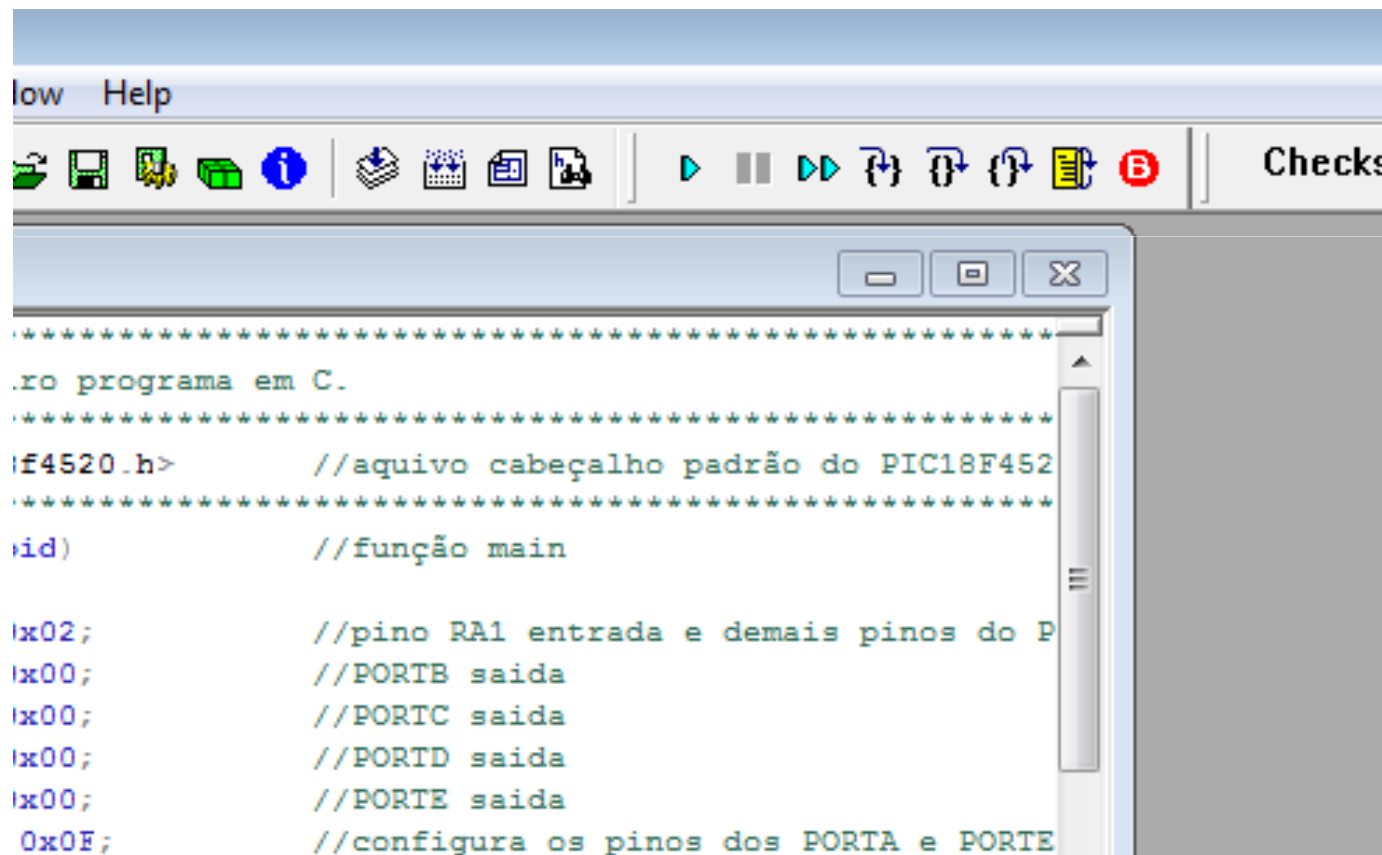
2. Criação de projeto no MPLAB® IDE para o MPLAB C18

7º Passo: Caso deseje **remover** um arquivo do projeto



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

8º Passo: Ativar o **MPASM SIM** para **simular** uma aplicação
Menu Debugger\Select Tool**MPLAB SIM**

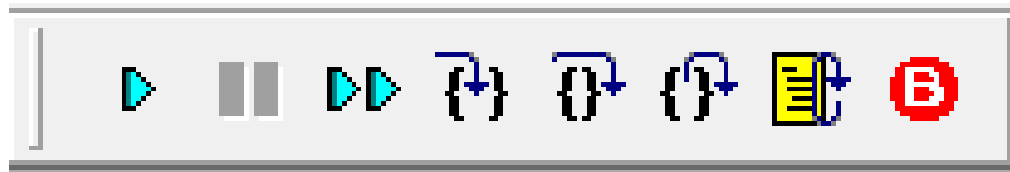


The screenshot shows the MPLAB IDE interface with a code editor window. The menu bar includes 'low' and 'Help'. The toolbar contains various icons for file operations, execution, and debugging. The code editor displays a C program for PIC18F4520. The code includes a header file 'f4520.h' and a 'main' function. The code is as follows:

```
*****  
ro programa em C.  
*****  
f4520.h>          //arquivo cabeçalho padrão do PIC18F452  
*****  
id)              //função main  
  
0x02;            //pino RA1 entrada e demais pinos do P  
0x00;            //PORTB saída  
0x00;            //PORTC saída  
0x00;            //PORTD saída  
0x00;            //PORTE saída  
0x0F;            //configura os pinos dos PORTA e PORTE
```

2. Criação de projeto no MPLAB® IDE para o MPLAB C18

8º Passo: Ativar o **MPASM SIM** para **simular** uma aplicação
Menu Debugger\Select Tool\MPASM SIM

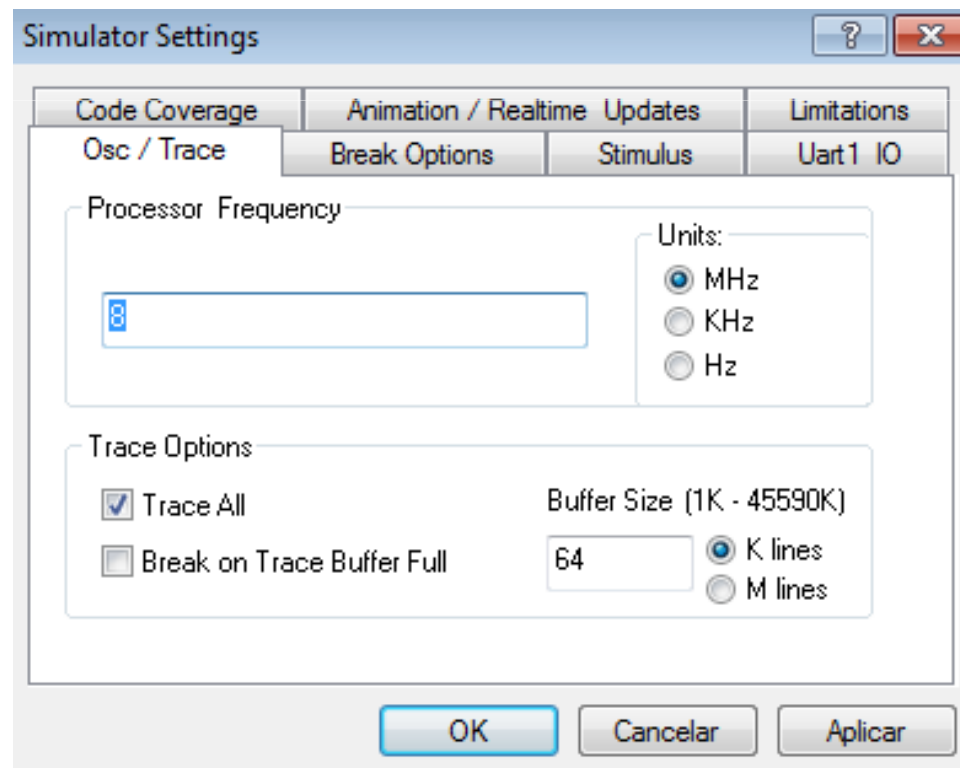


- **Run:** executa o programa (F9);
- **Halt:** parar a execução do programa (F5);
- **Animate:** Executa o programa no modo animação e mostra a linha que o programa está executando no momento;
- **Step Into:** executa o programa passo a passo (F7);
- **Step Over:** executa de uma só vez todos os comando da função;
- **Step Out:** sai da função e retorna ao ponto em que ela foi chamada;
- **Reset** (F6);
- **Breakpoint:** A execução é interrompida na linha em que ele foi inserido;

2. Criação de projeto no MPLAB® IDE para o MPLAB C18

8º Passo: Ativar o **MPASM SIM** para **simular** uma aplicação

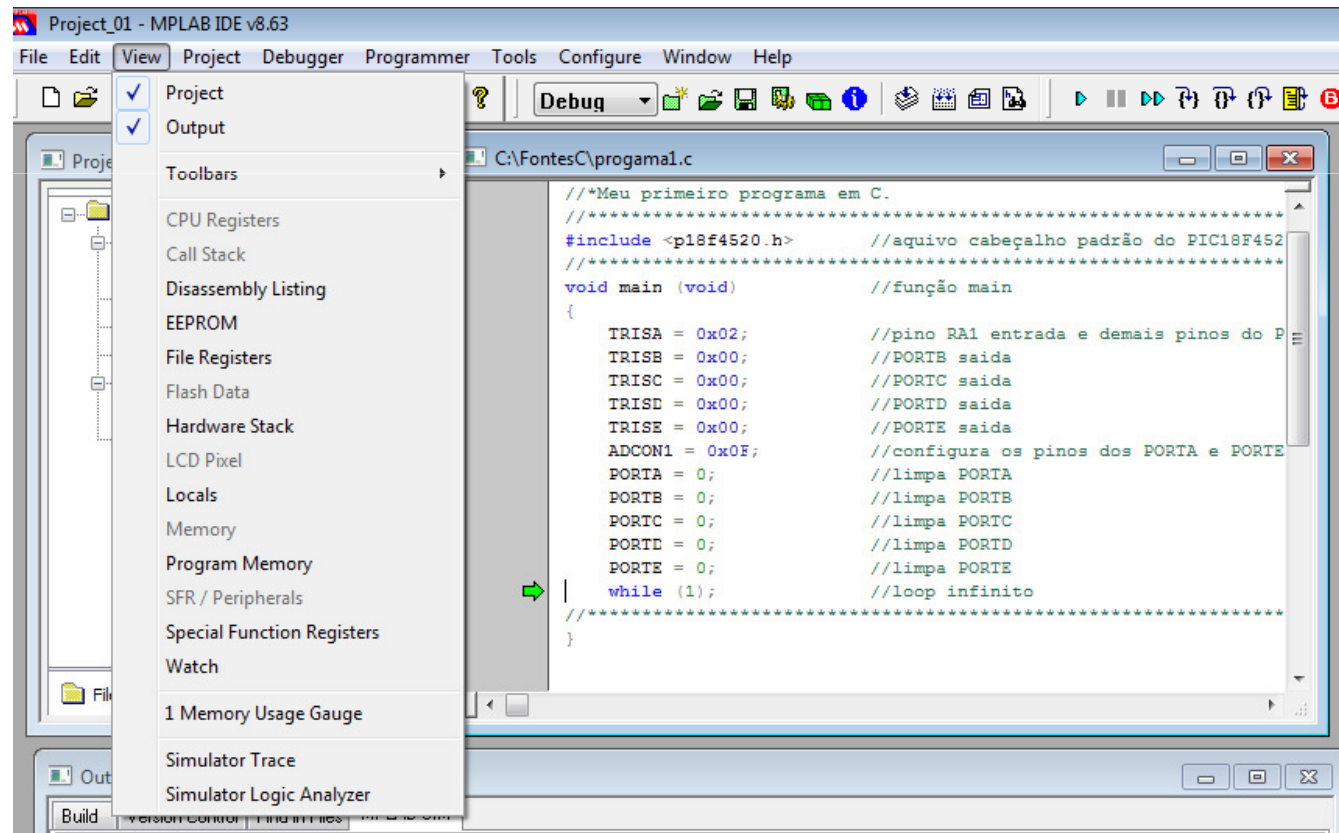
- Configurações de Simulação (**Debugger\Setitings**):
- Frequência de clock: **8Mhz** (**Osc/Trace**);
- Serial RS232: **Uart 1 I/O**;



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

8º Passo: Ativar o **MPASM SIM** para **simular** uma aplicação

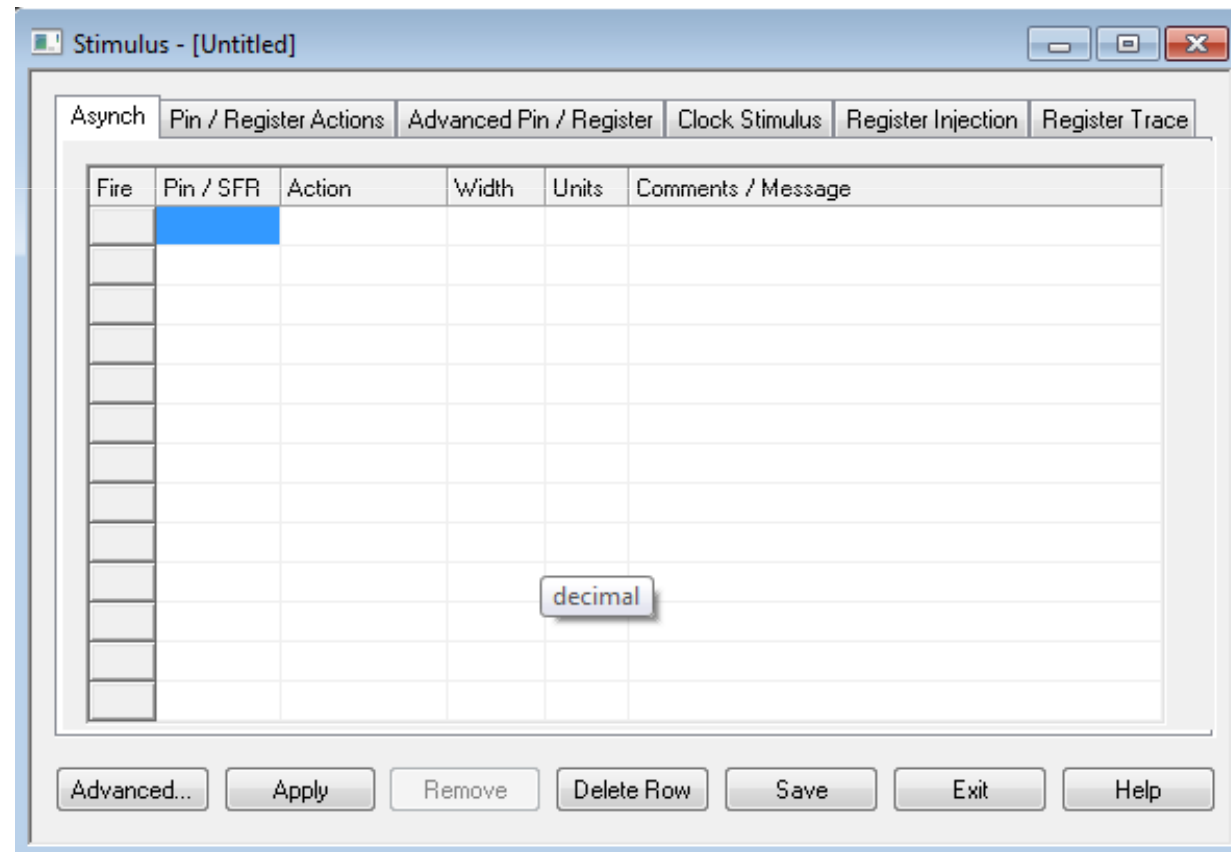
- Clicando no menu **View**, é possível visualizar o conteúdo das **memórias de programa, de dados e EEPROM**;



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

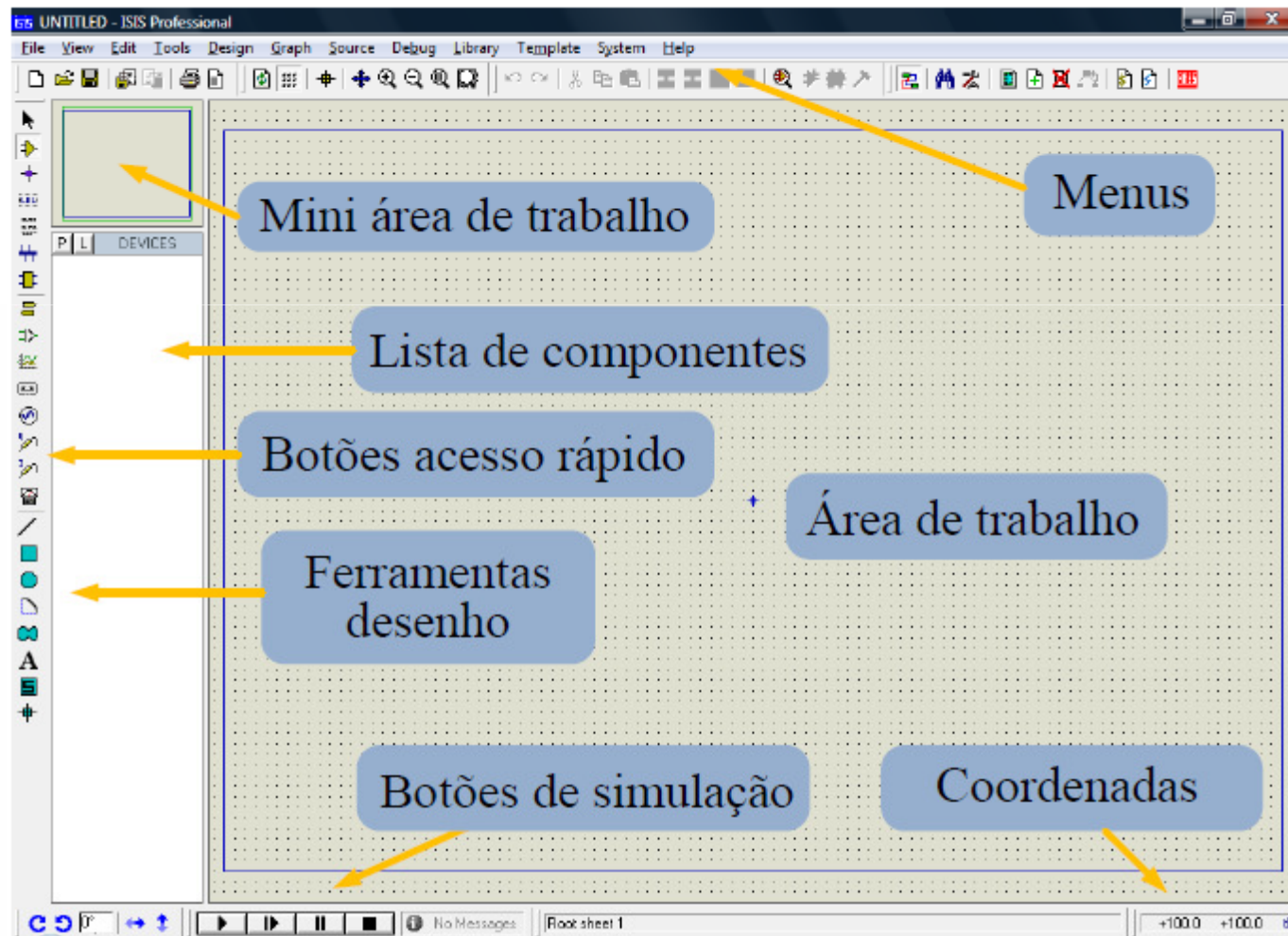
8º Passo: Ativar o **MPASM SIM** para **simular** uma aplicação

- Clicando no menu **Debugger/Stimulus** permite introduzir **estímulos nos pinos durante a simulação**;



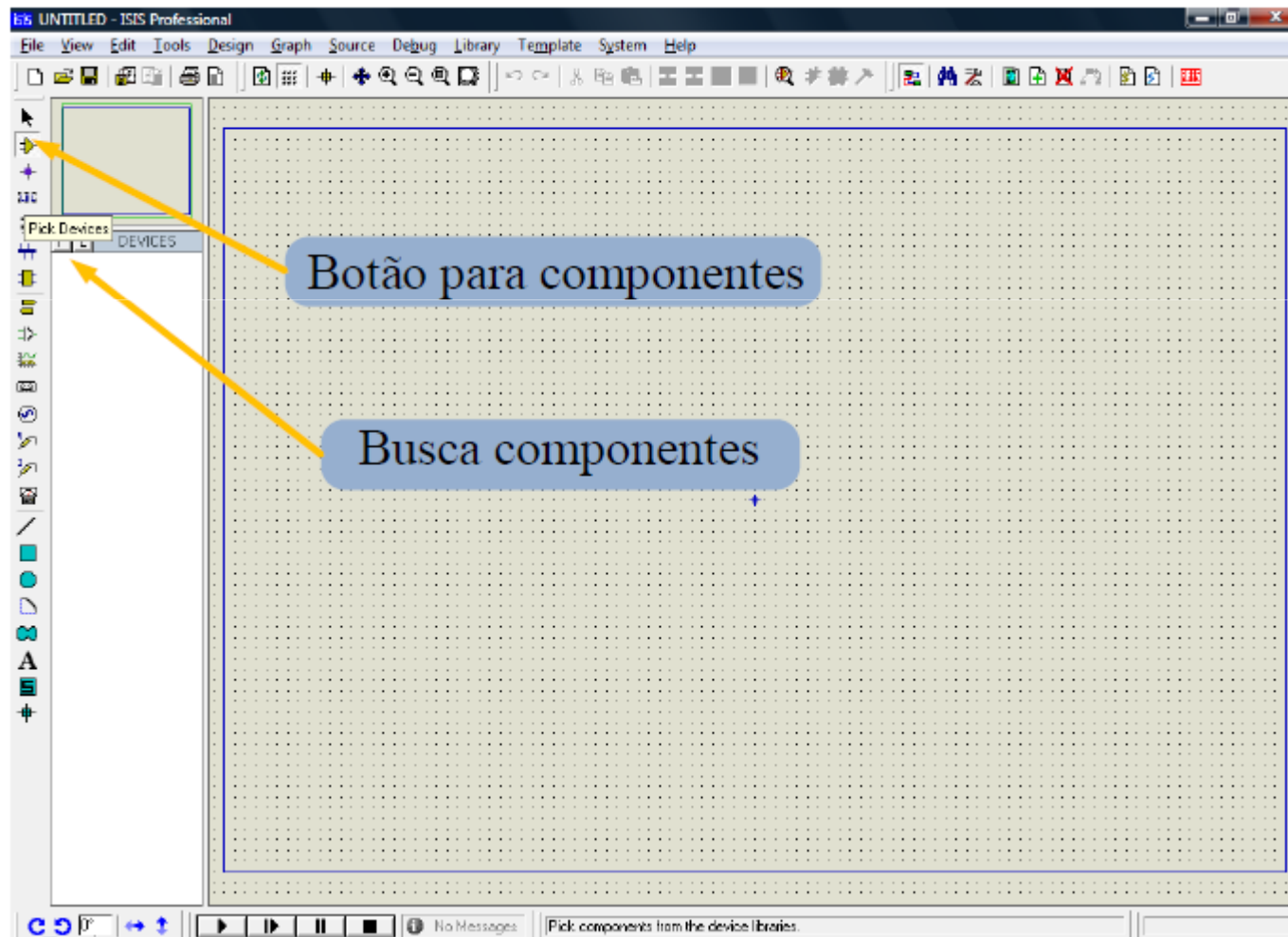
3. Simulador - Proteus 7.4 (ISIS)

- Tela de abertura do Proteus 7.4 (ISIS)



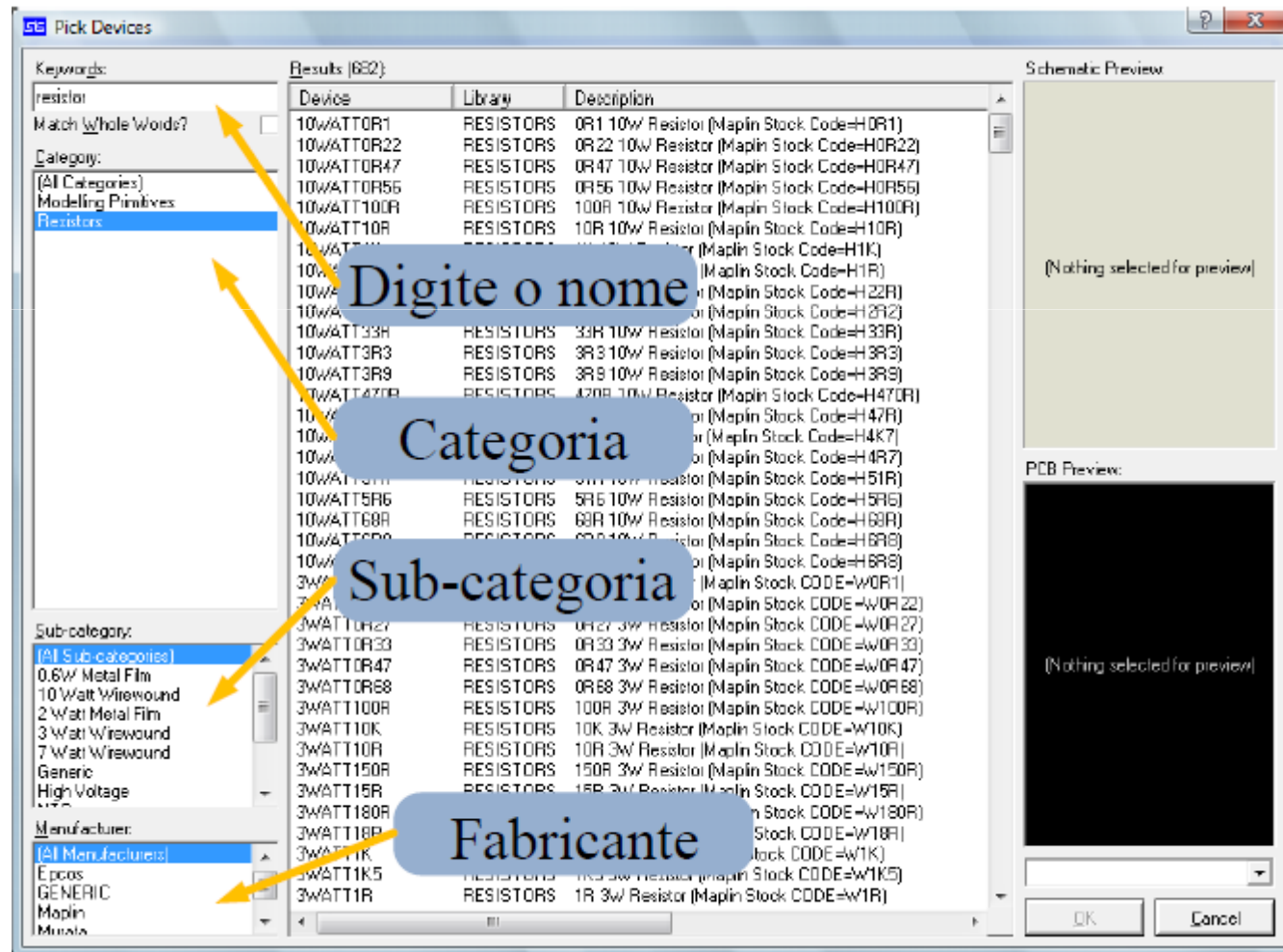
3. Simulador - Proteus 7.4 (ISIS)

- 1º Passo: **Busca** de componentes;



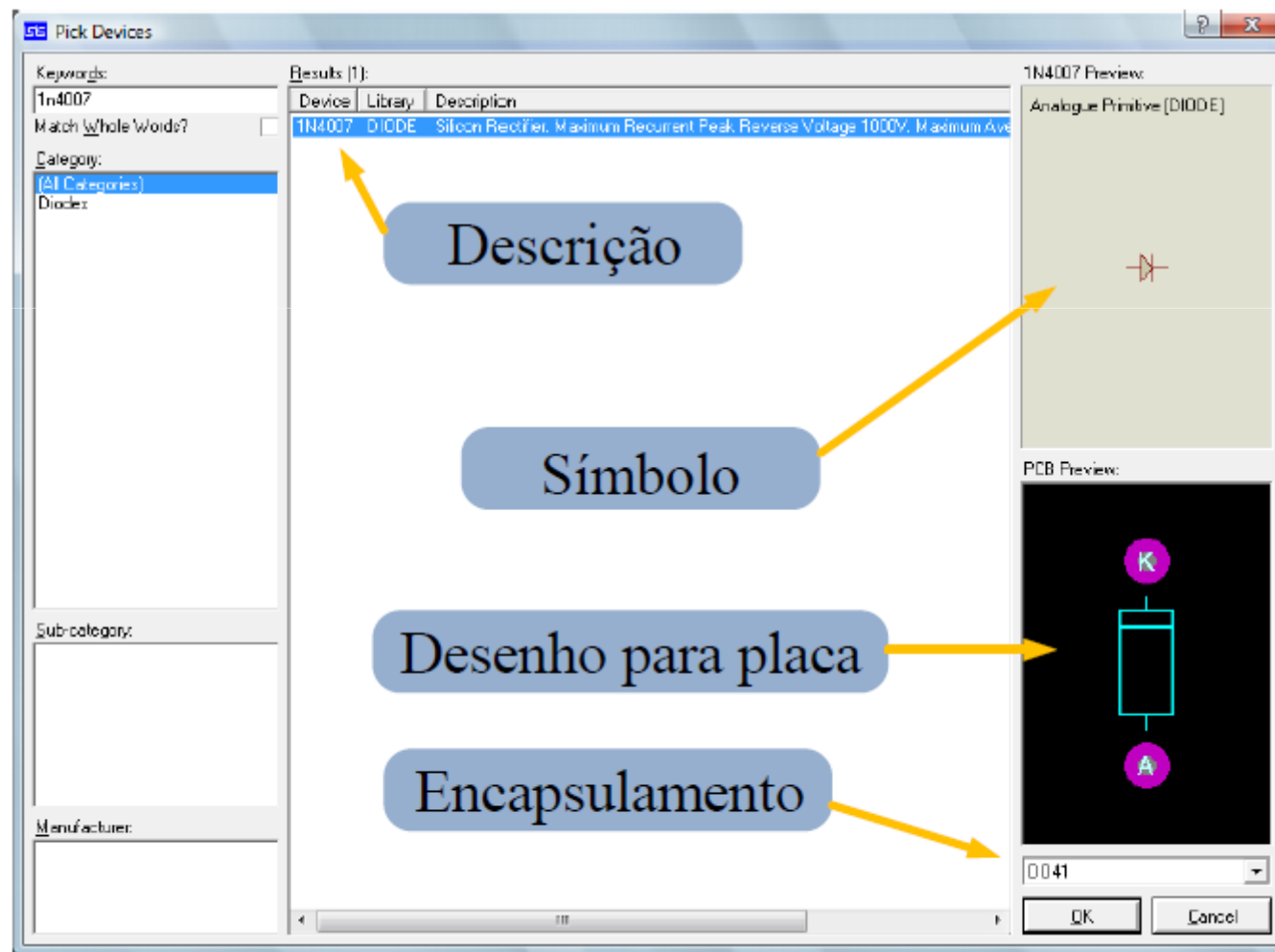
3. Simulador - Proteus 7.4 (ISIS)

- 1º Passo: **Busca** de componentes / Tela de busca;



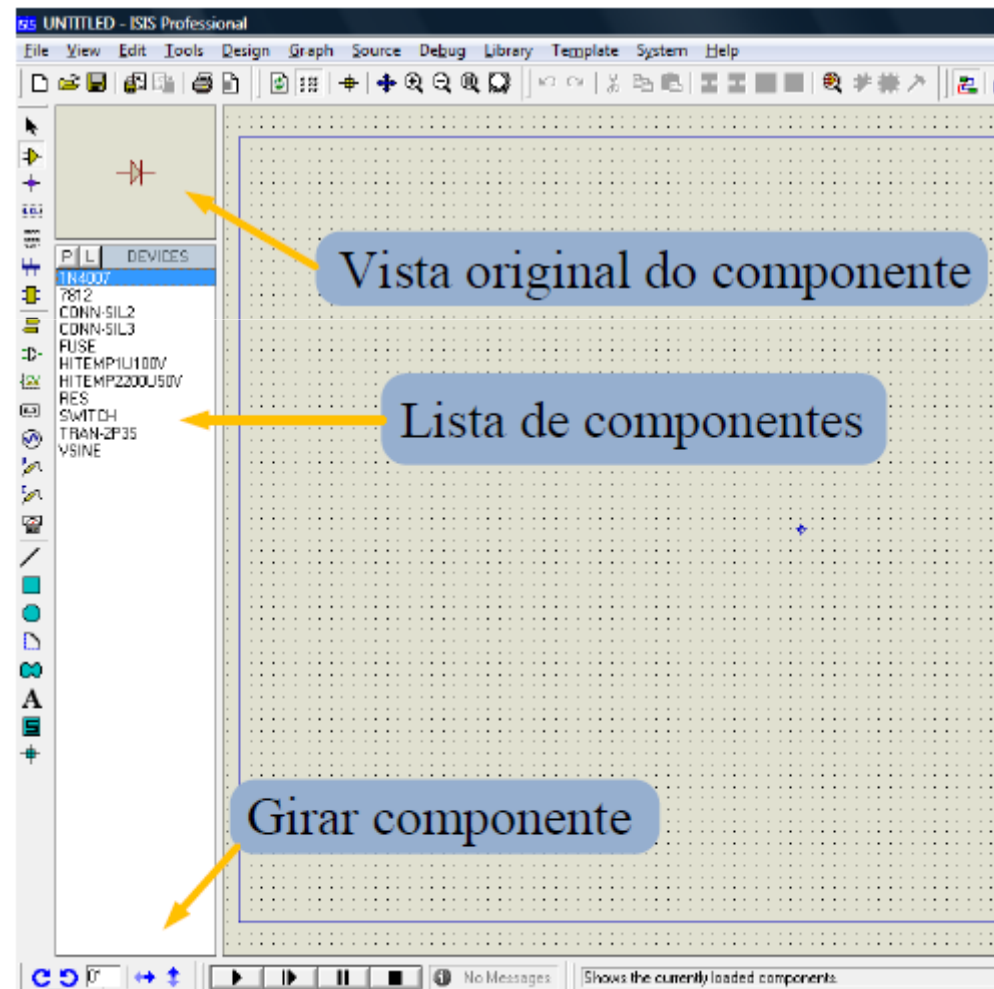
3. Simulador - Proteus 7.4 (ISIS)

- 1º Passo: **Busca** de componentes / Exemplo de componente;



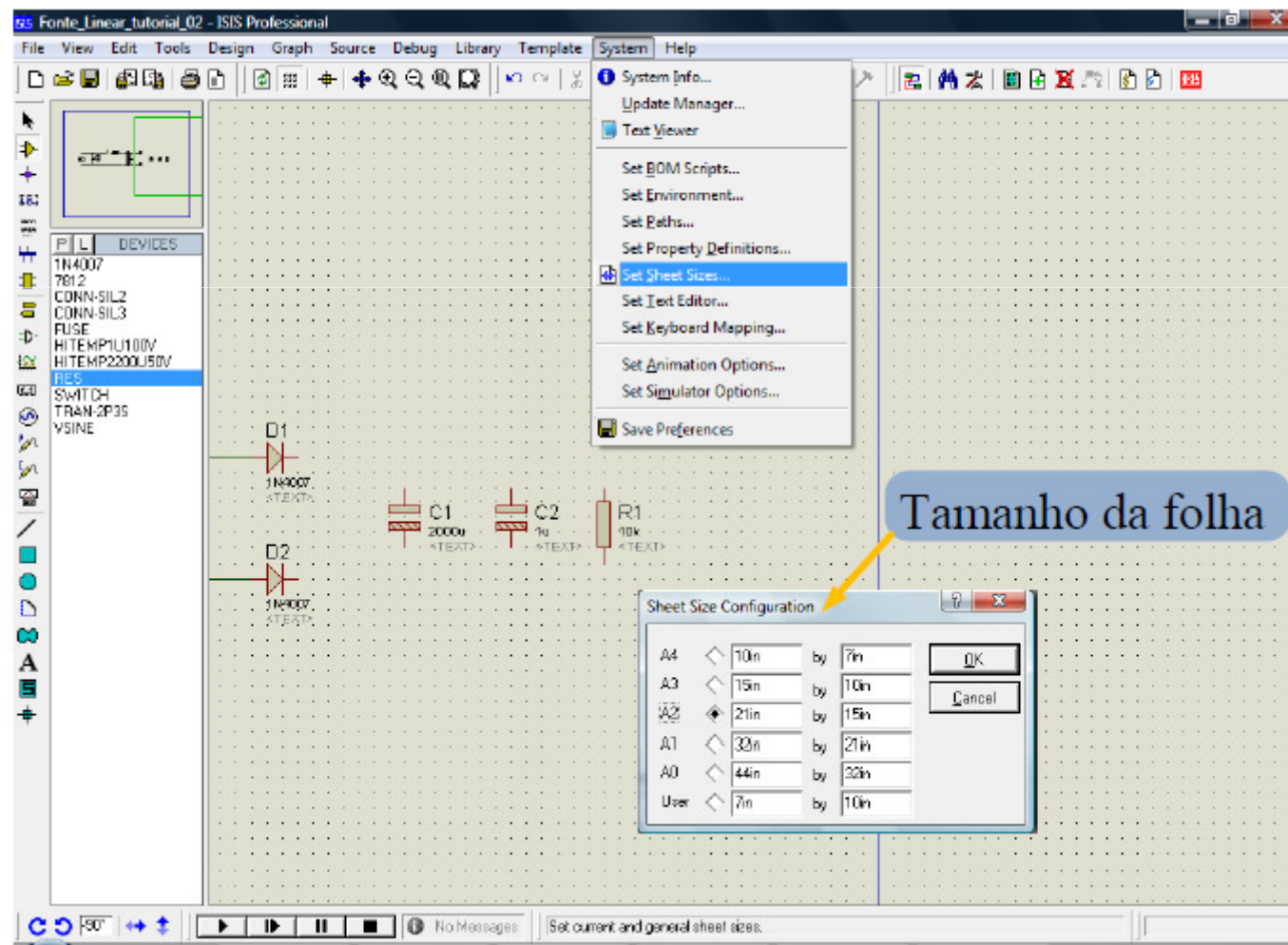
3. Simulador - Proteus 7.4 (ISIS)

- 2º Passo: **Girando** componentes antes de posicioná-los;



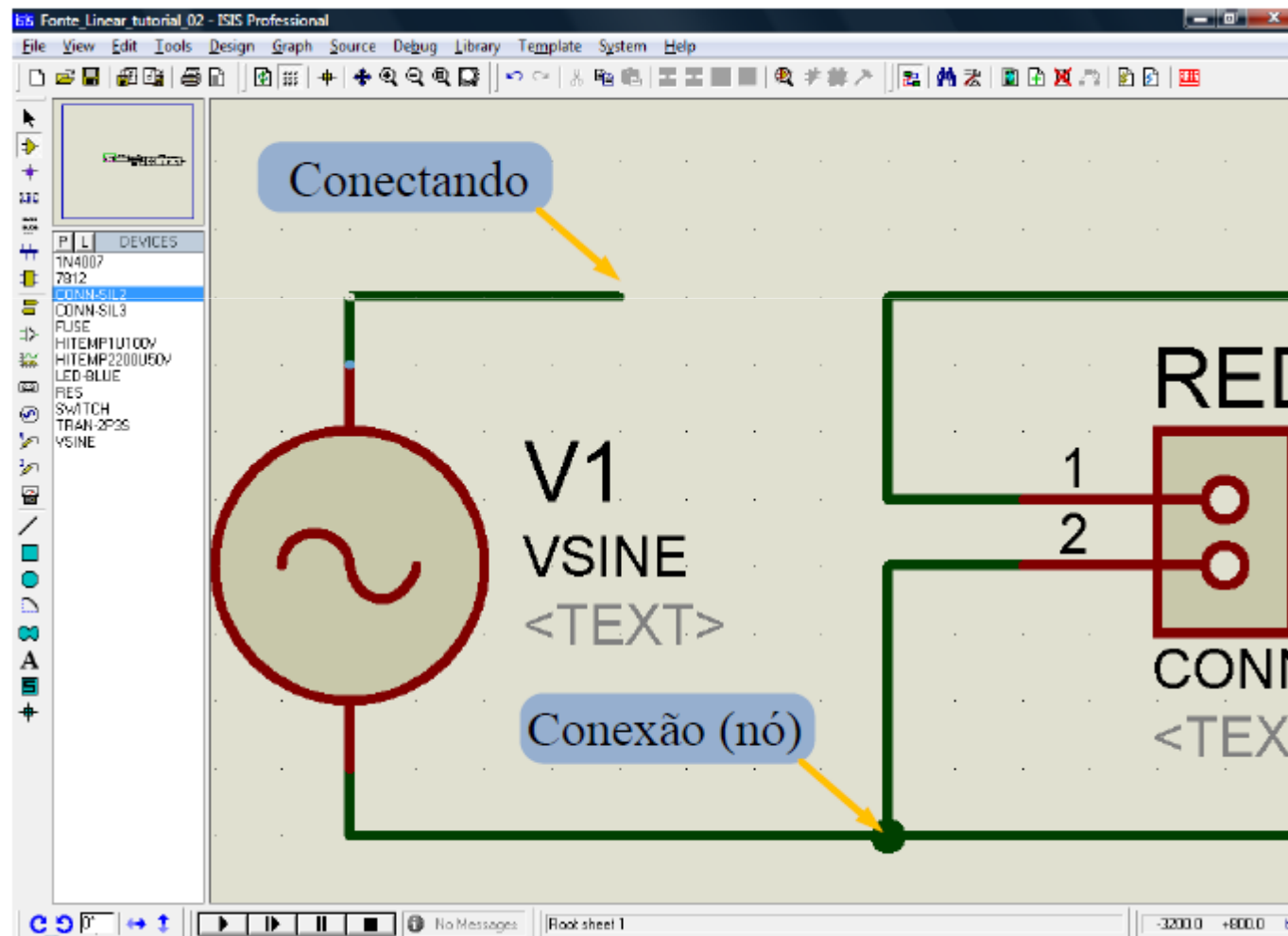
3. Simulador - Proteus 7.4 (ISIS)

- 3º Passo: Alterando o tamanho da folha;



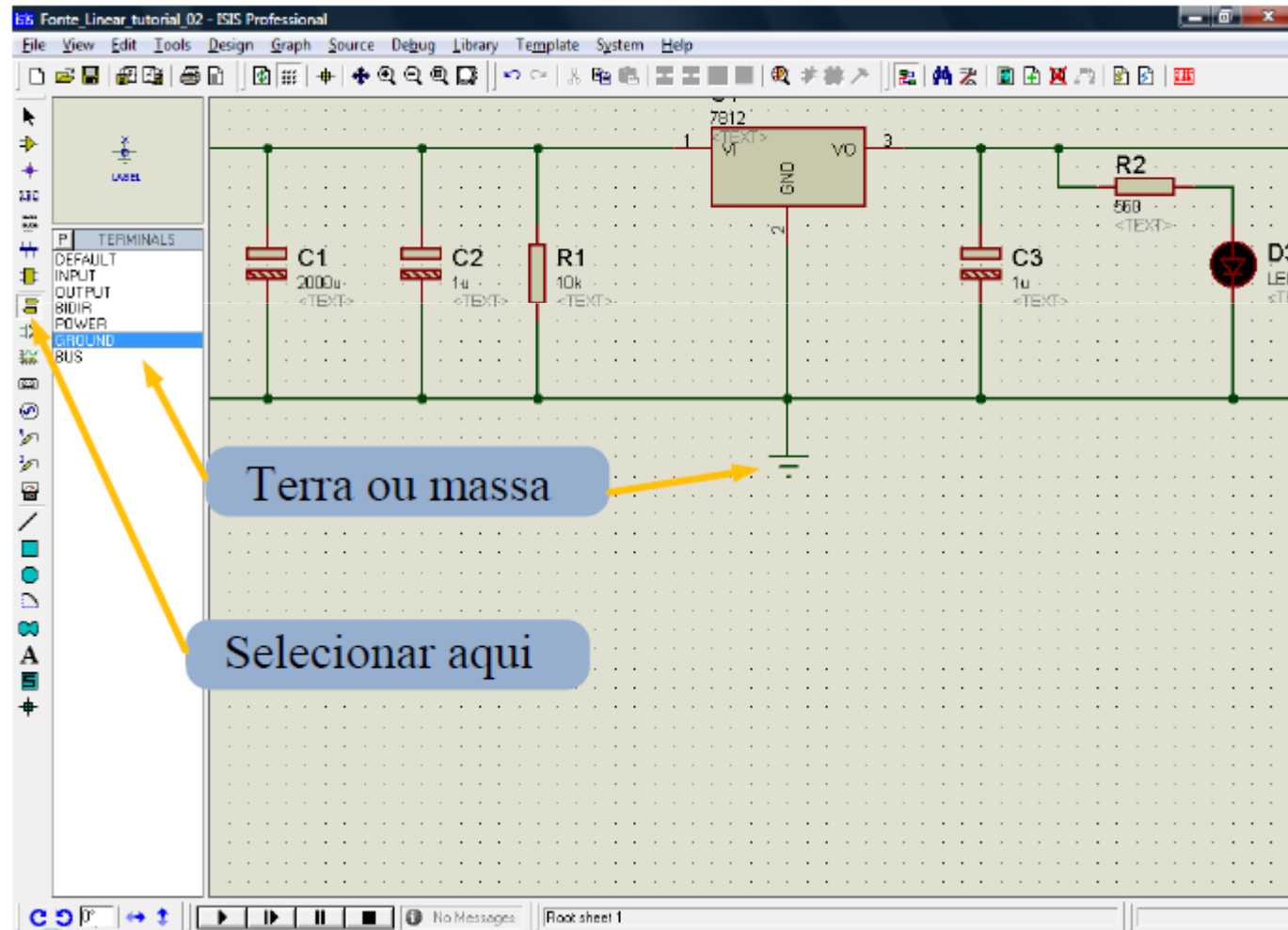
3. Simulador - Proteus 7.4 (ISIS)

- 4º Passo: Conectando os componentes;



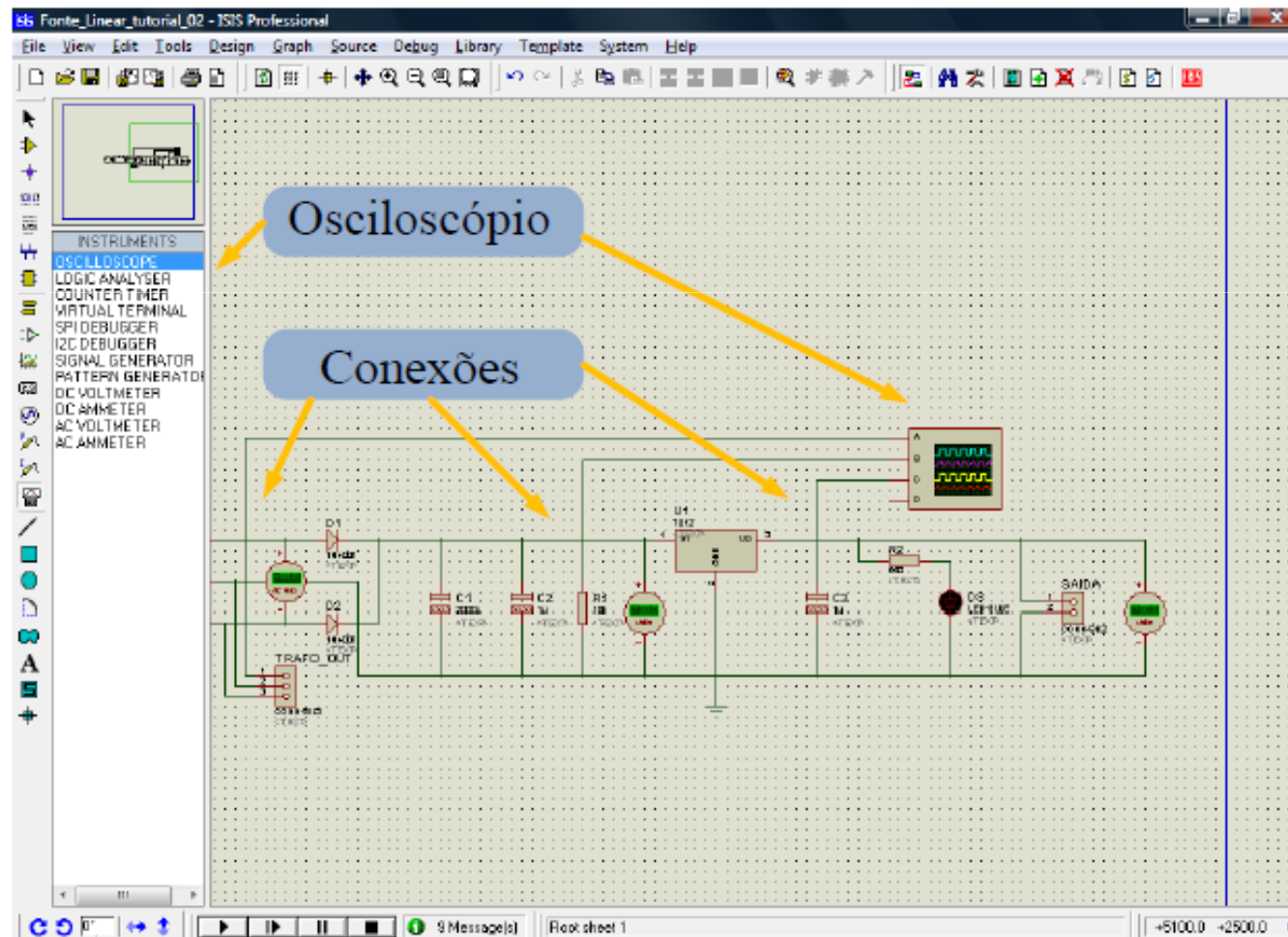
3. Simulador - Proteus 7.4 (ISIS)

- 4º Passo: Conectando os componentes;

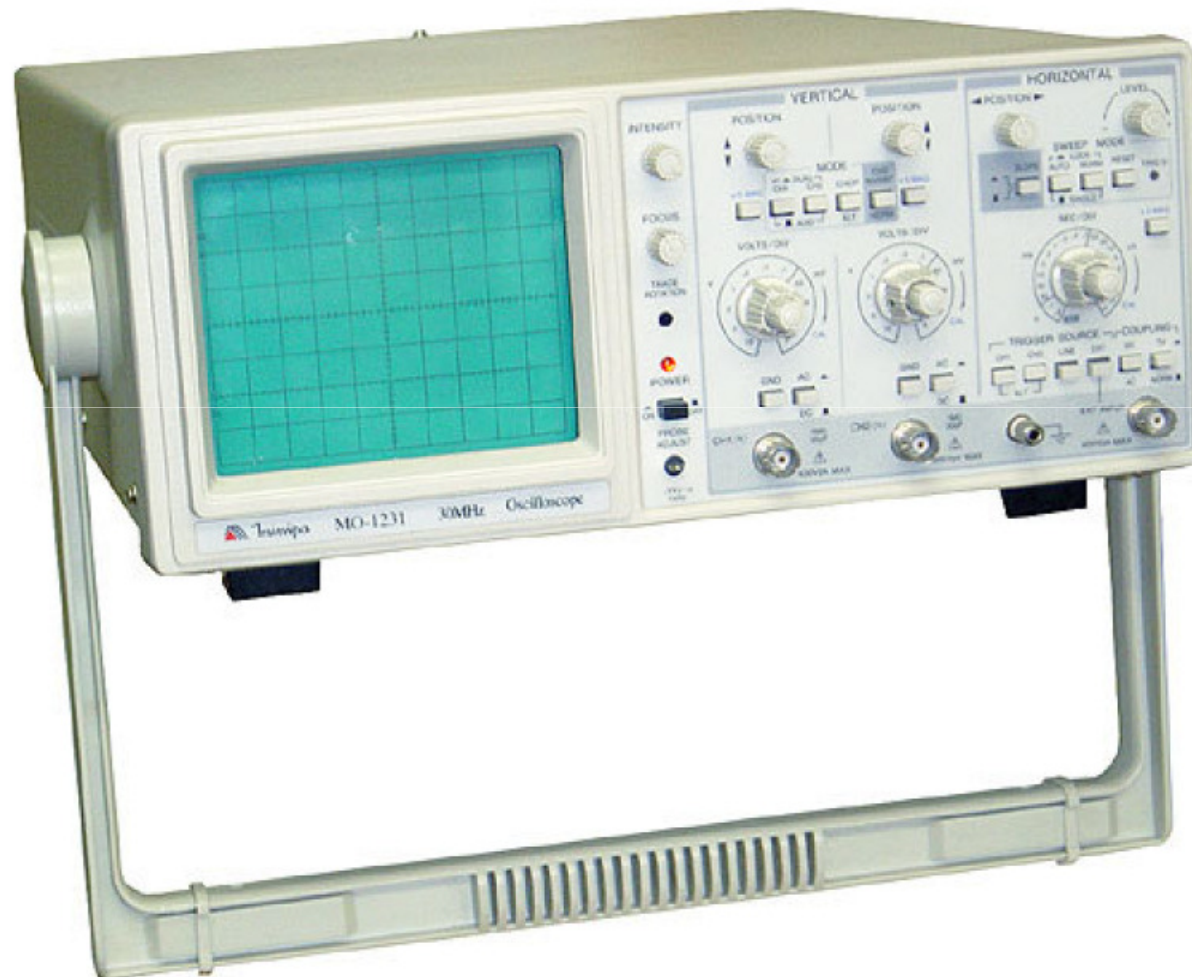


3. Simulador - Proteus 7.4 (ISIS)

- 5º Passo: Conectando um **osciloscópio**;

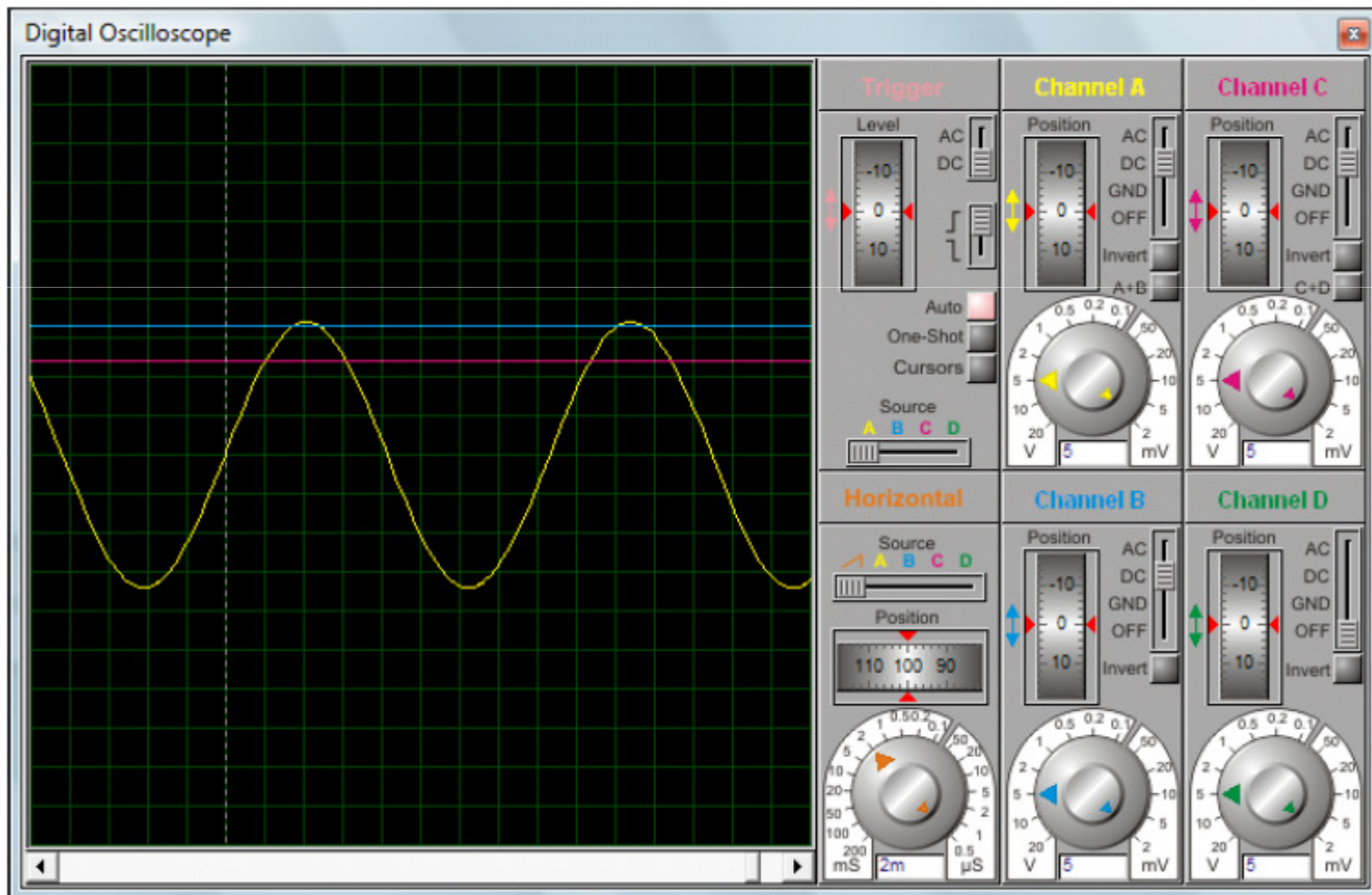


6. Osciloscópio



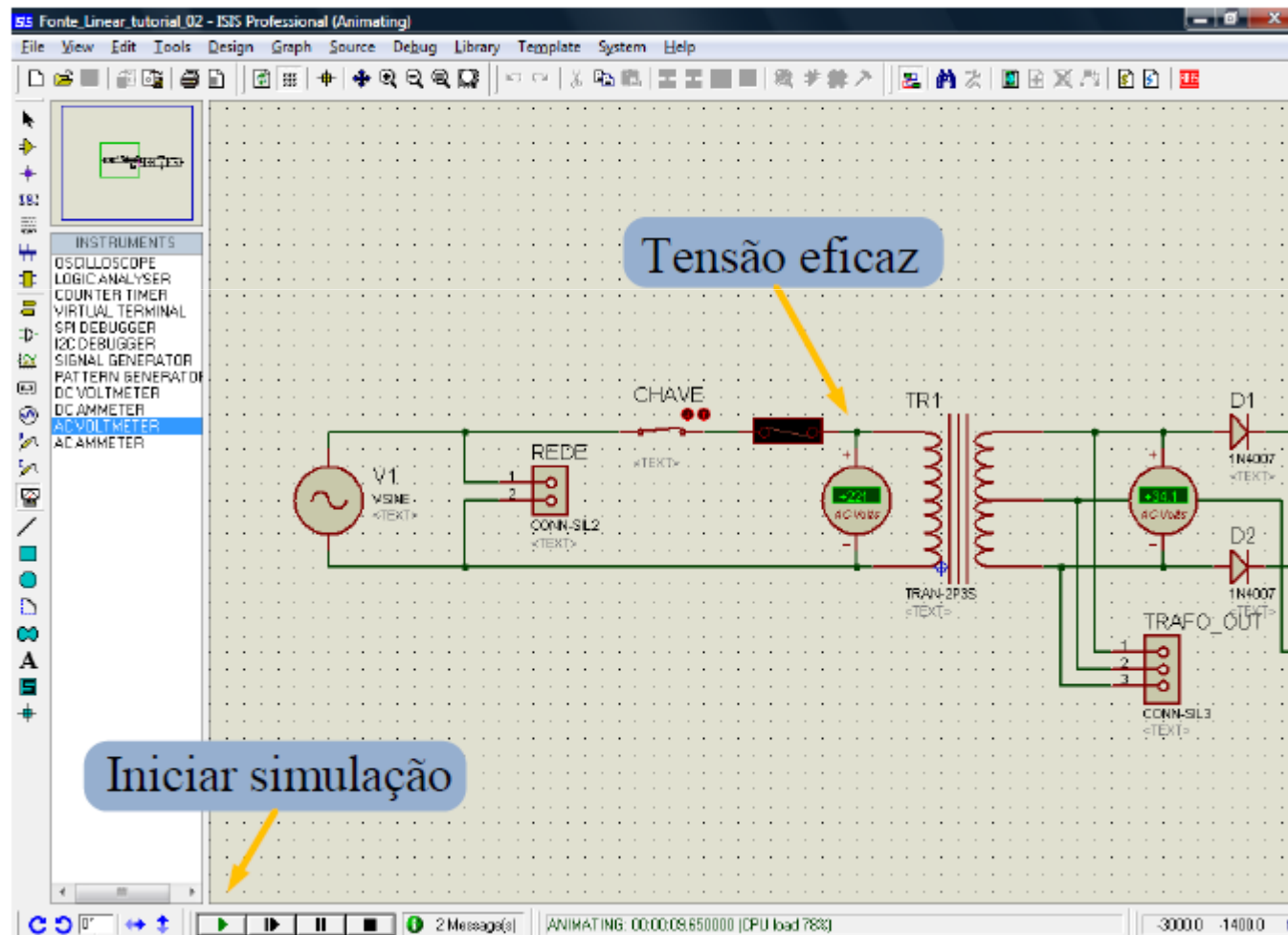
3. Simulador - Proteus 7.4 (ISIS)

- 5º Passo: Conectando um [osciloscópio](#);



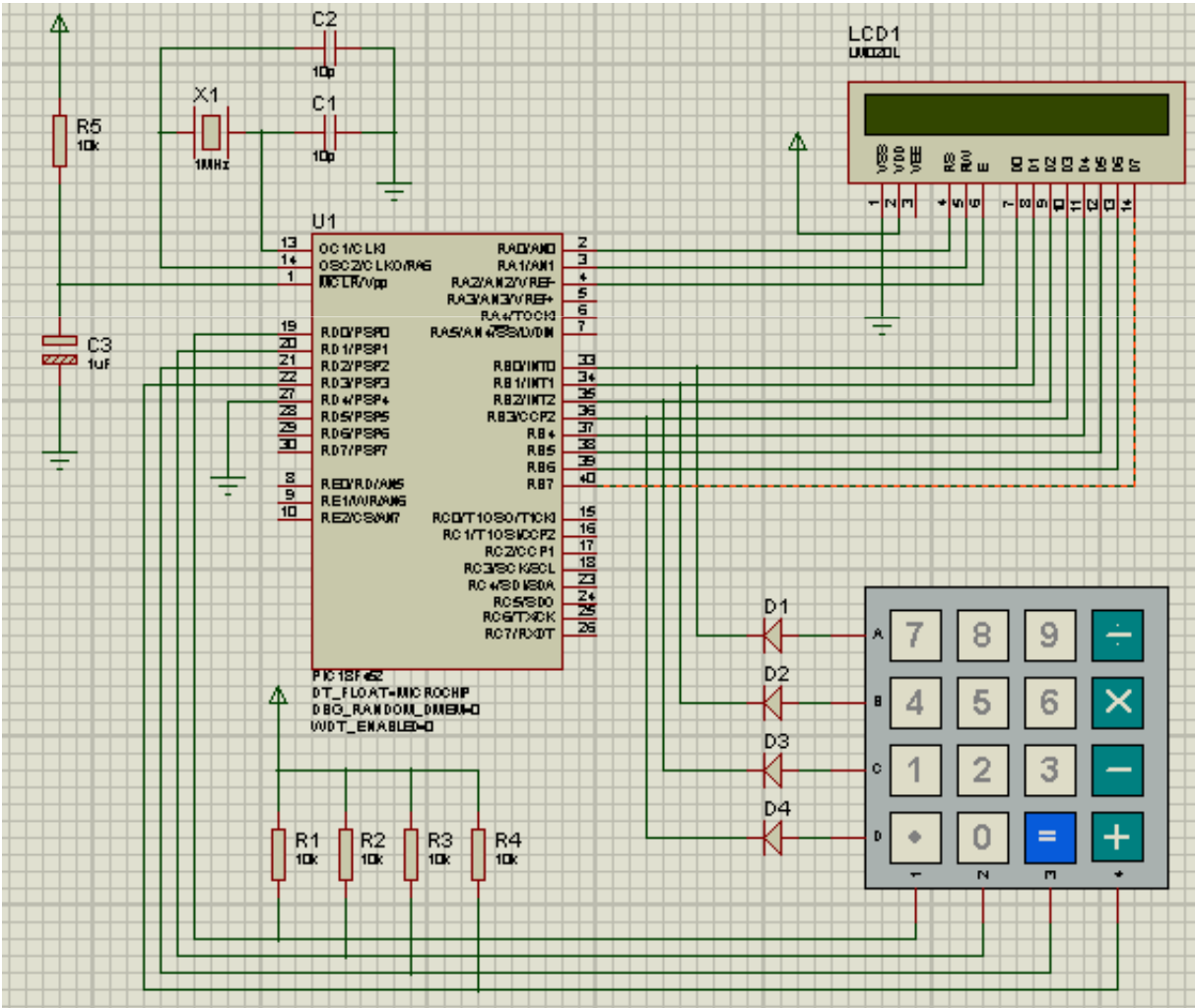
3. Simulador - Proteus 7.4 (ISIS)

- 5º Passo: **Iniciando** a simulação;



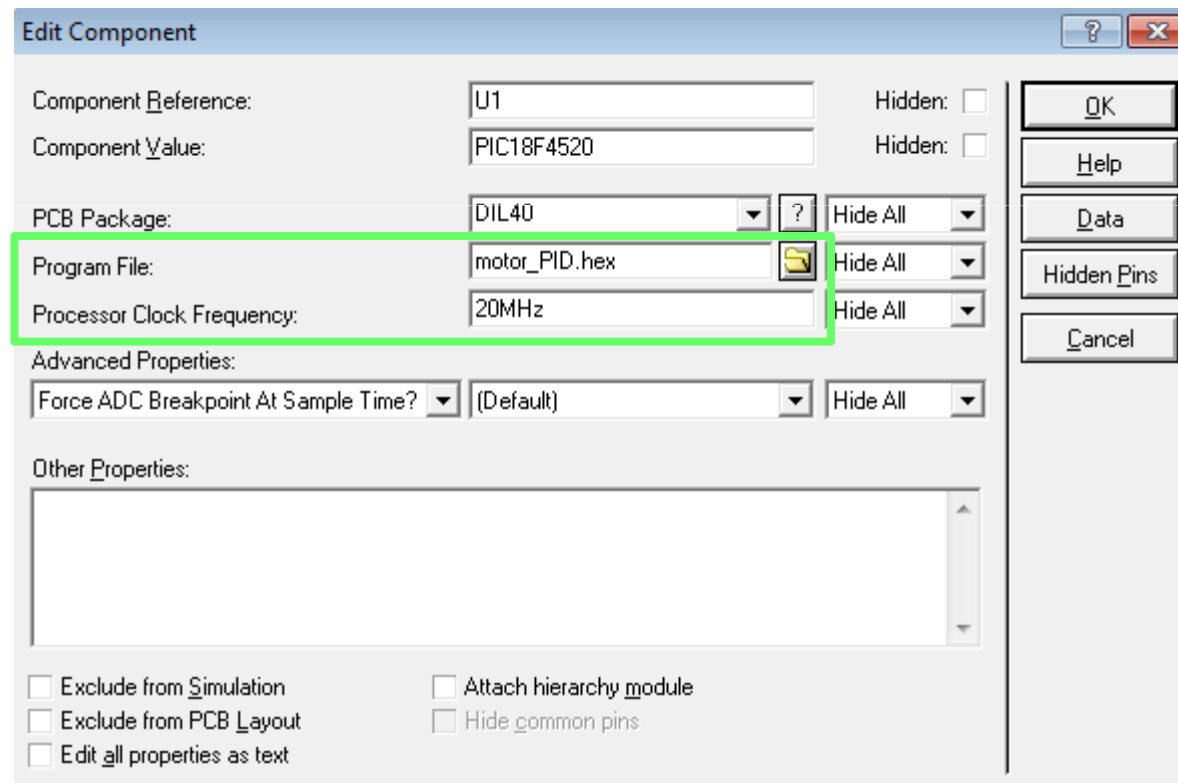
3. Simulador - Proteus 7.4 (ISIS)

- Simulação de **microcontroladores**;



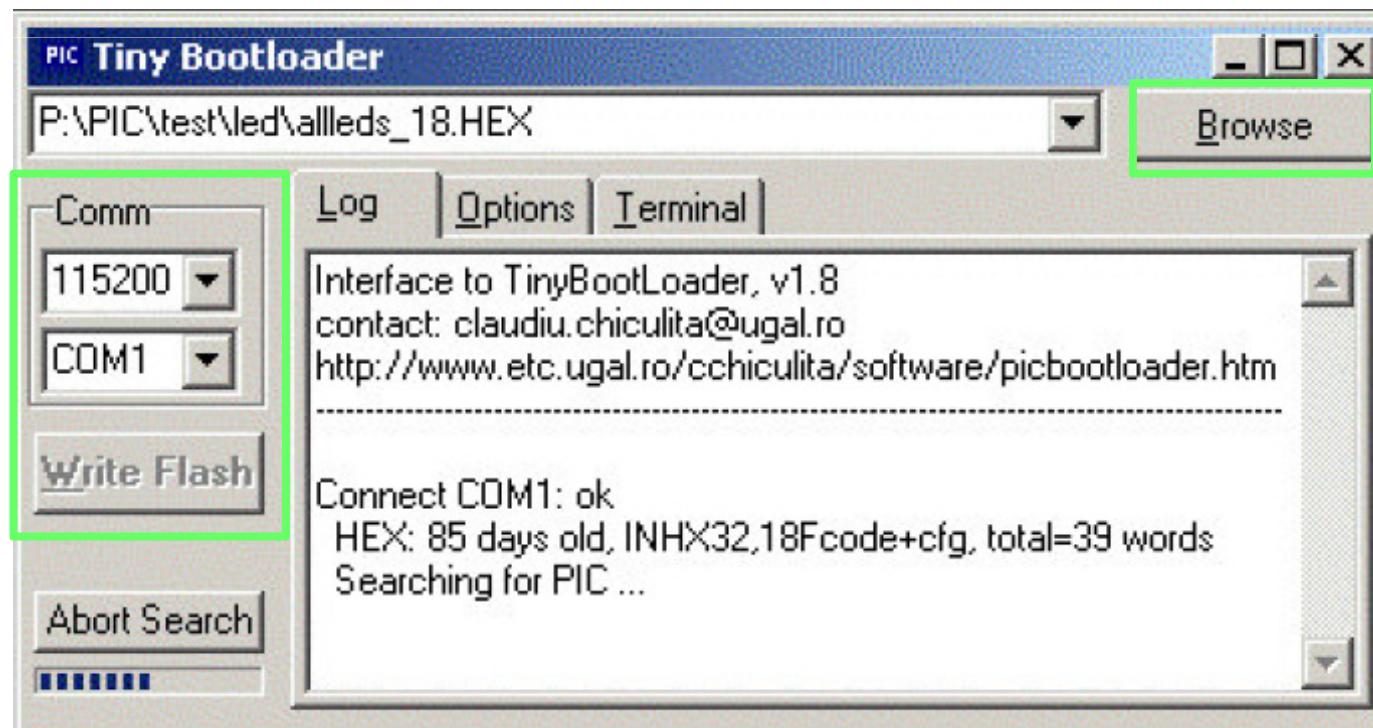
3. Simulador - Proteus 7.4 (ISIS)

- “Program File” - selecionar o arquivo **.hex** à ser programado ;
- “Processor Clock Frequency” - especificar a frequência de clock;



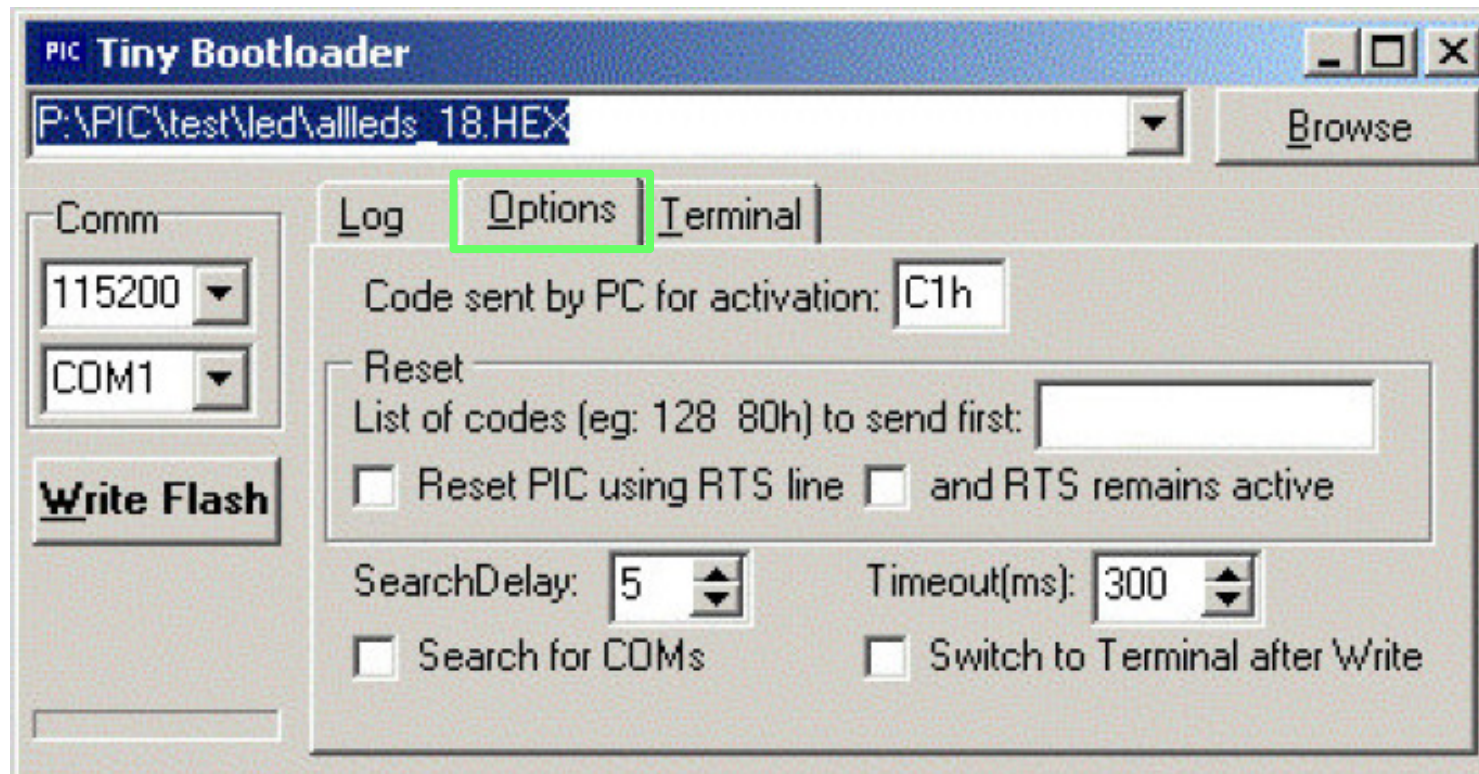
4. Gravador - Tiny Bootloader

- 1º Passo: Selecione o arquivo **.Hex** à ser programado;
- 2º Passo: Conecte a **placa** à porta serial;
- 3º Passo: Selecione a **porta serial** conectada (Com1, Com2, etc.);
- 4º Passo: Pressione no **software** o botão **"Write"**;
- 5º Passo: Pressione na **placa** o botão de **"Reset"**;



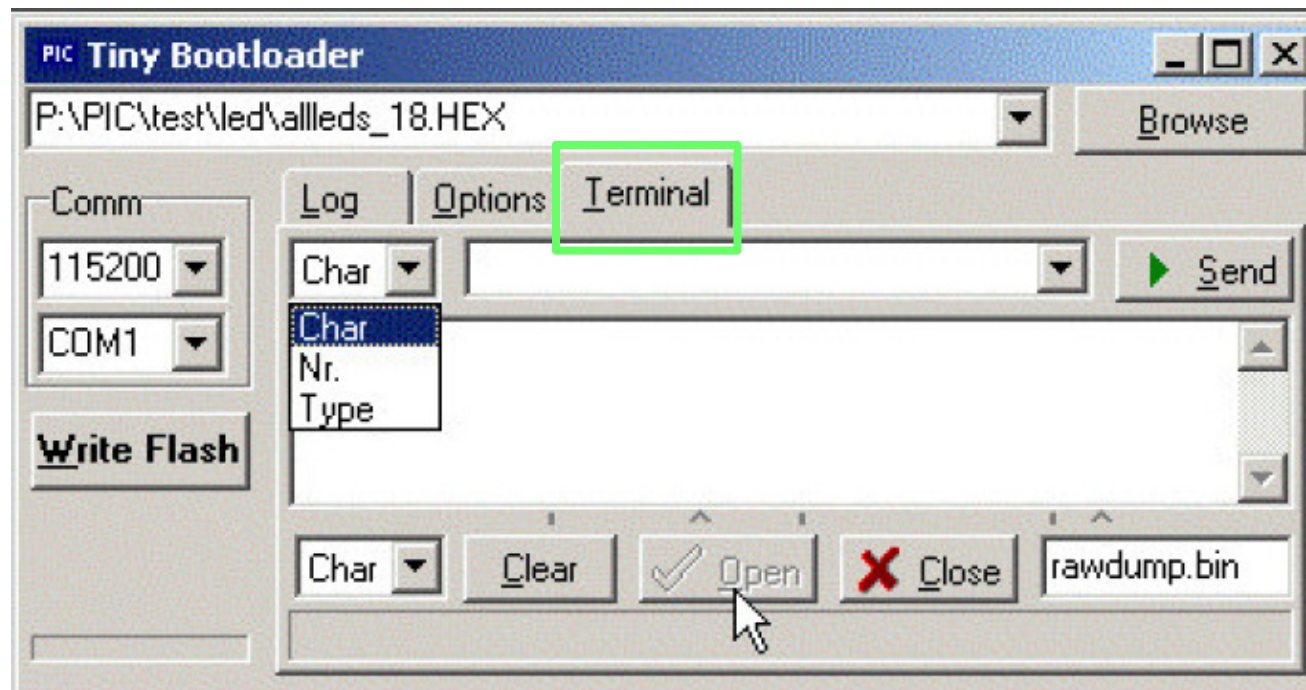
4. Gravador - Tiny Bootloader

- “SearchDelay” é o tempo que o aplicativo tentará contato com o pic;
- “Timeout” é o tempo limite para todas as operações;



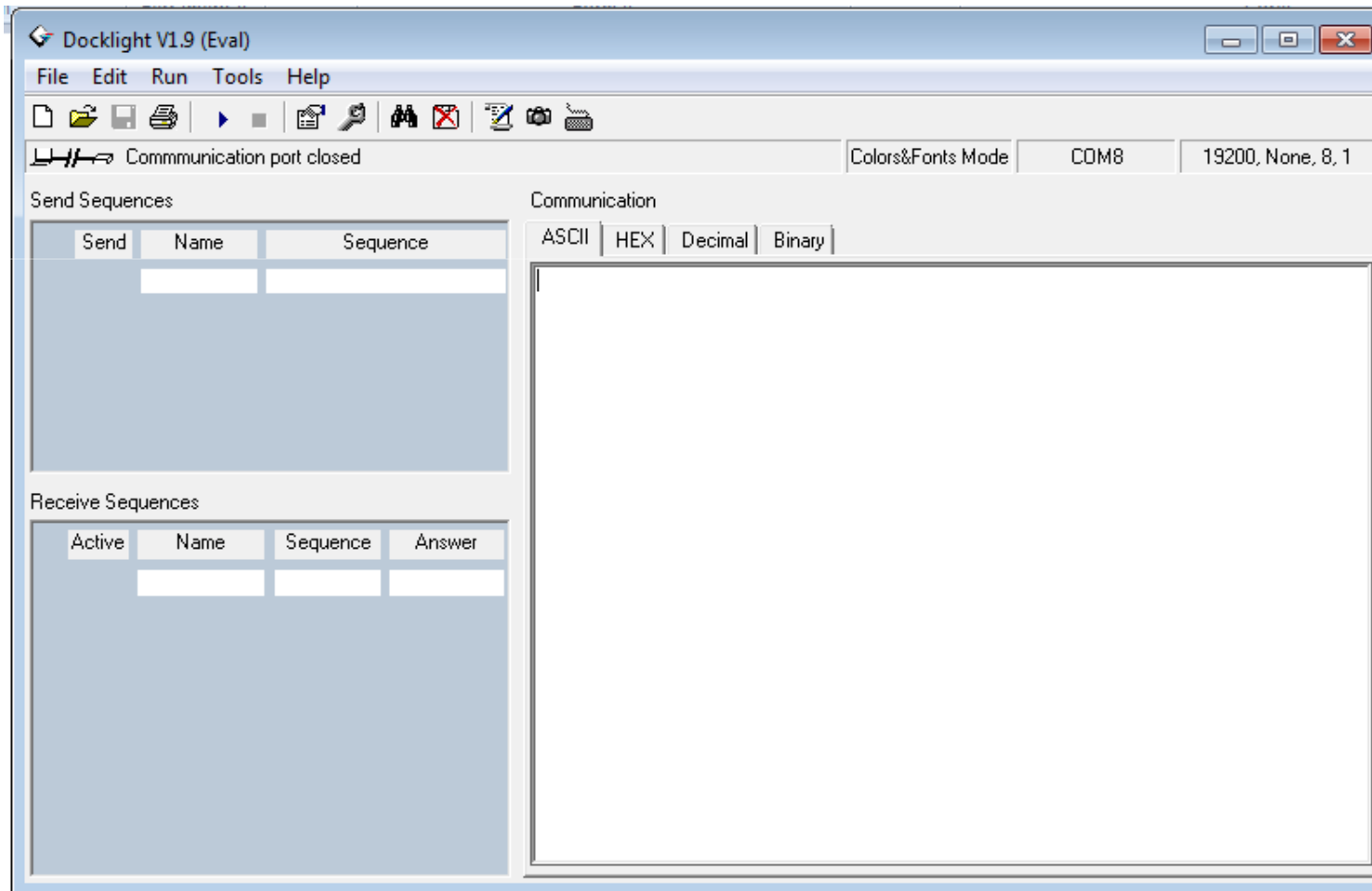
4. Gravador - Tiny Bootloader

- Terminal rudimentar, com as seguintes possibilidades:
 - **Mostrar** dados recebidos;
 - **Salva** automaticamente os dados brutos;
 - **Pode enviar**: um por vez ou uma seqüência de caracteres ASCII, decimal ou hexa;
 - **Buffer de exibição** padrão de 10k (ajustável);



5. Comunicação Serial - DockLight

- Clicando no menu **Tools\Project Settings** permite configurar os parâmetros de comunicação (porta, taxa de comunicação, etc);



5. Comunicação Serial – RComSerial



2. Criação de projeto no MPLAB® IDE para o MPLAB C18

Exemplo 02 (Exercício)

1. Salvar o arquivo “programa2.c”;
2. Criar projeto;
3. Compilar;
4. Simular;
5. Executar passo-a-passo;
6. Testar no kit de desenvolvimento;

```
#include <p18f4520.h> //arquivo cabeçalho padrão do PIC18F4520
//*****
int counter;
void main (void)
{
    counter = 1;
    TRISB = 0;          /* configure PORTB for output */
    while (counter <= 15)
    {
        PORTB = counter; /* display value of 'counter' on the LEDs */
        counter++;
    }
}
```

Próxima Aula

Aula 05

Arquitetura PIC18 (18F4520 - Uma Visão Geral) Parte II