



## **Criação de aplicações para a placa DETPIC32 com o MPLAB X IDE**

Universidade de Aveiro  
 Departamento de Eletrónica, Telecomunicações e Informática  
 Paulo Pedreiras  
[paulo.pedreiras@gmail.com](mailto:paulo.pedreiras@gmail.com)

### **História**

Versão	Data	Descrição
1.0	7-2-2013	Release inicial
1.1	15-3-2013	Adição de secção sobre a aplicação para PC multiOS (Java)
1.2	12-7-2014	Adição de secção com makefile (contribuição de Ricardo Marau)

### **Índice**

1 Sumário.....	1
2 Requisitos.....	2
3 Criação do projeto.....	2
4 Edição e compilação.....	5
5 Carregamento do programa na placa DETPIC32.....	6
5.1 Bootloader da Microchip.....	6
5.2 Aplicação PC multiOS – Java.....	7
6 Agradecimentos.....	8

## **1 Sumário**

Este documento descreve o processo de criação de uma aplicação para execução na plataforma DETPIC32, usando a *toolchain* da Microchip (MPLAB IDE X e compilador XC32). Para além dos passos necessários à criação do projeto, são também sumariamente apresentadas as aplicações para PC que permitem efetuar o carregamento das aplicações.

## 2 Requisitos

- Deverão estar instalados o MPLAB IDE X e o XC32, ambos disponíveis no site da Microchip.
- Este software pode ser descarregado de:  
<http://www.microchip.com/pagehandler/en-us/family/mplabx/#downloads>
- Estes programas estão disponíveis para plataformas Linux e Windows e podem ser instalados em versão *freeware*
- A aplicação PC e o bootloader propriamente dito podem ser obtidos em:  
<http://ppedreiras.av.it.pt/resources.htm>

## 3 Criação do projeto

1. Executar o MPLab IDE
2. File->New project
3. Selecionar “Microchip Embedded” + “Standalone Project” + “Next” (Figura 1)
4. Selecionar o microcontrolador PIC32MX795F512H e botão “Next”(Figura 2)
5. Deixar a opção por defeito em “Hardware tools” (ICD3) e fazer “Next” (Figura 3)
6. Selecionar o compilador XC32 e fazer “Next” (Figura 4)
7. Atribuir um nome ao programa (caixa “Project Name”) e selecionar uma localização adequada para o projeto (Figura 5)

*O MPLAB X IDE cria automaticamente uma pasta com o mesmo nome do projeto, a que é adicionada extensão “X” (e.g. SampleProg.X).*

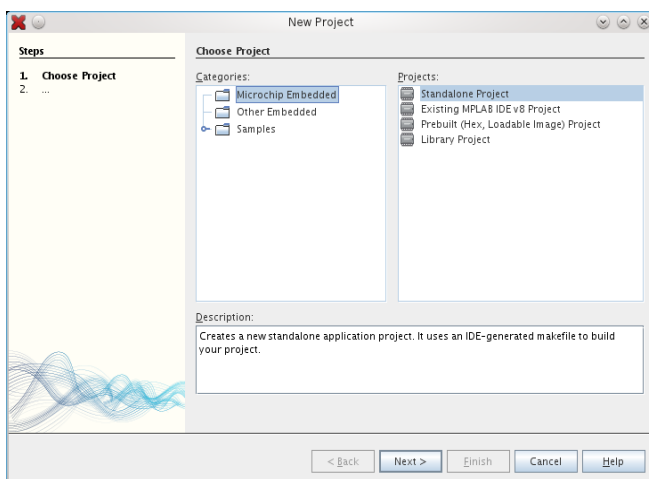


Figura 1

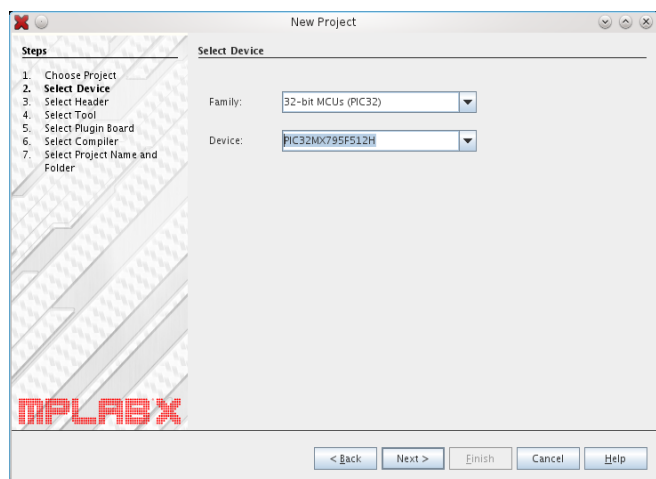


Figura 2

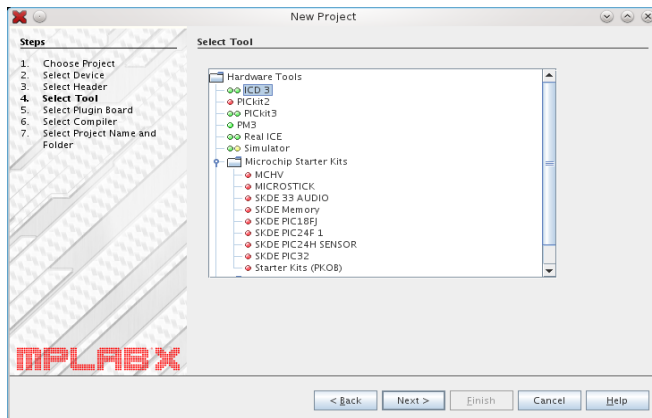


Figura 3

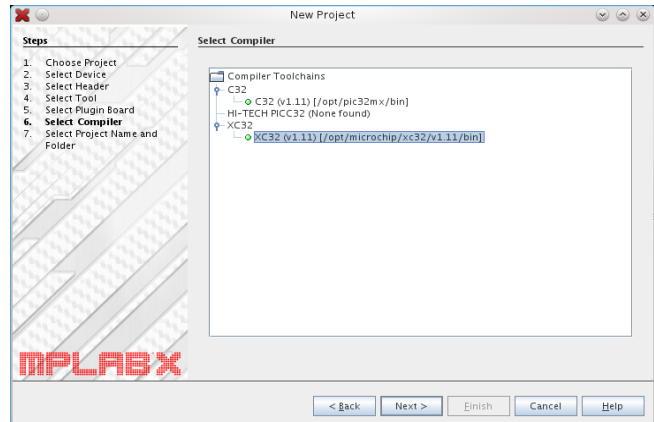


Figura 4

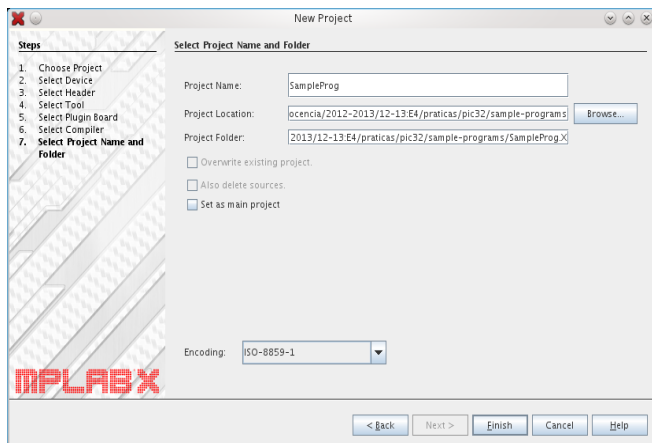


Figura 5

8. Copiar para a pasta do projeto o ficheiro “linker.ld”, disponível na página da disciplina.
9. Clicar com o botão direito do rato em “Linker Files”, selecionar “Add Existing item” e depois navegar até à pasta do projeto e selecionar o ficheiro “linker.ld” (Figura 6)

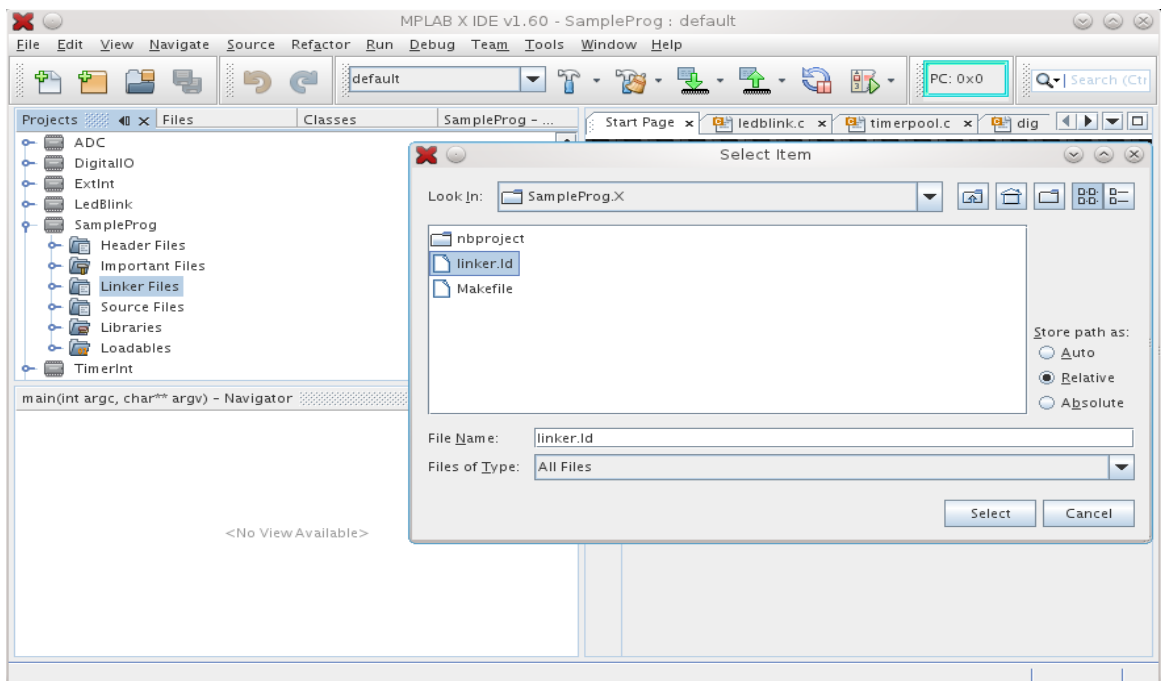


Figura 6

- Adicionar ficheiros “.c” (subpasta “Source Files”) e “.h” (subpasta Header Files”), conforme as necessidades.

*Estes ficheiros pode ser criados de raiz (usando o botão do lado direito sobre a subpasta adequada e escolhendo a opção “New”) ou ser copiados de outros projetos. Neste caso devem ser adicionados de forma semelhante ao referido para o “linker script”.*

## 4 Edição e compilação

### 4.1 Toolchain da Microchip

Uma vez criado o projeto, deverá obter-se uma vista semelhante à reportada na Figura 7.

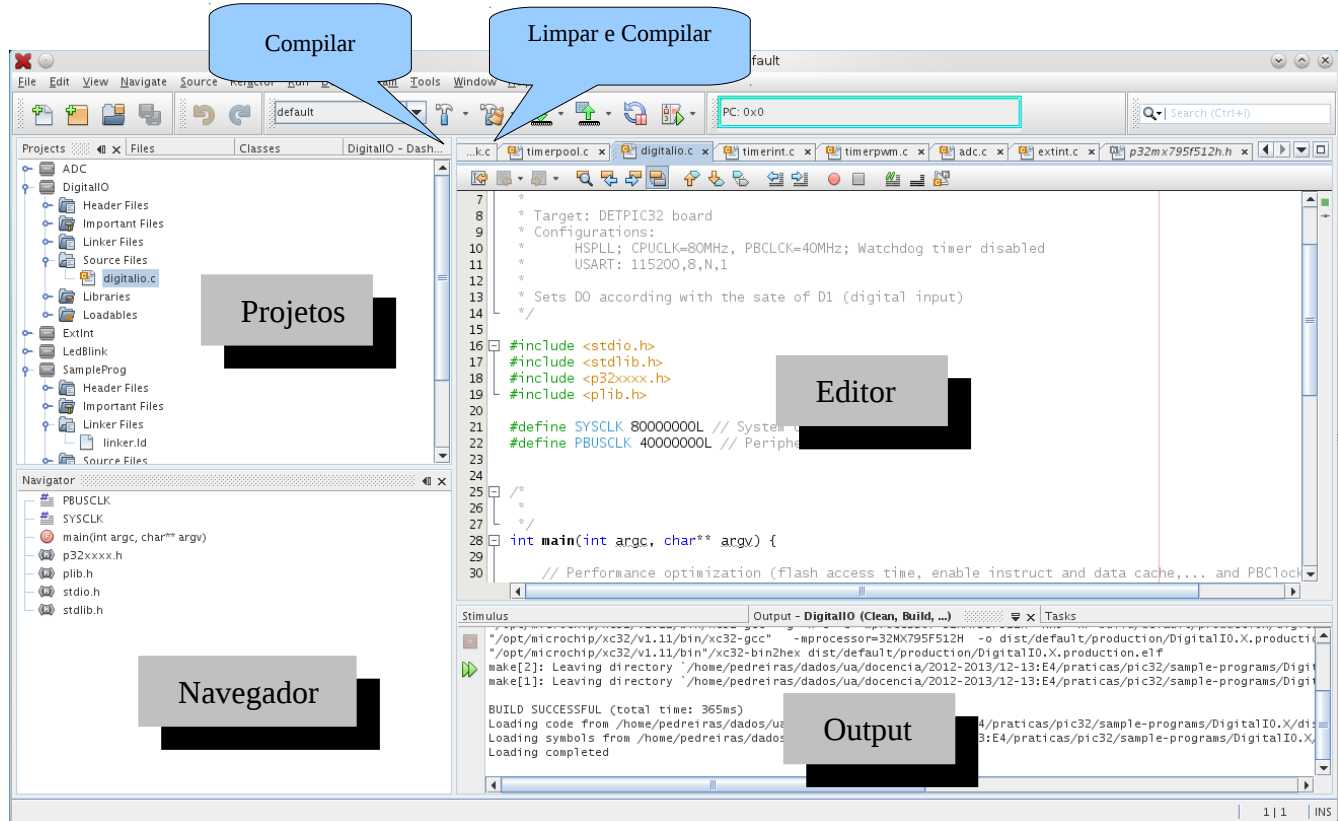


Figura 7

A utilização deste IDE (baseado em NetBeans <http://netbeans.org/>) é bastante intuitivo. O ecrã encontra-se dividido em quatro janelas principais:

- “**Projects**”: gerir projetos (adicionar, eliminar e associar ficheiros a projetos)
- “**Navigator**”: variáveis, definições, funções e header files usadas no projeto
- “**Editor**”: Editor de texto
- “**Output**”: resultados da compilação

No topo encontra-se um conjunto de menus e botões. Existem inúmeras funções disponíveis (copiar/colar texto, pesquisa, formatação, gravar, etc).

Aconselha-se a leitura da documentação relativa ao IDE, em particular o “MPLAB® X IDE User’s Guide” (<http://ww1.microchip.com/downloads/en/DeviceDoc/52027B.pdf>).



## 4.2 Editor básico e makefile

A criação de aplicações também pode ser efectuada com recurso a um editor de texto básico. Neste caso, é conveniente automatizar o processo de compilação e *linkagem*, por forma a evitar-se a digitação de longas linhas de comando que, além de consumirem tempo, podem induzir em erros, alguns deles pouco óbvios e que podem levar a perdas de tempo significativas (e.g. esquecer de compilar um “.c” de uma biblioteca de funções, levando à inclusão de código antigo).

A ferramenta de eleição, em ambientes Linux, para automatizar este tipo de processos é o utilitário “Make”. Este utilitário permite uma automatização de todo o processo de compilação e *linkagem*. Este utilitário tem como entrada um ficheiro, usualmente chamado “makefile”, o qual define como é que a compilação deve ser efectuada (linha de comando do compilador, nome da aplicação, ficheiros que a compõem, ...).

Em “<http://ppedreiras.av.it.pt/resources/detpic32-bootloader/blinker.tgz>” pode encontrar uma “makefile”, desenvolvida por Ricardo Marau ([rmarau@gmail.com](mailto:rmarau@gmail.com)), a qual pode ser facilmente adaptada a novos projectos.

## 5 Carregamento do programa na placa DETPIC32

Por defeito, o compilador cria o ficheiro “.hex”, a carregar na placa DETPIC32, dentro da pasta “[PROJ\_FOLDER]/dist/default/production/”, com nome “PROJ\_NAME.X.production.hex”, em que “PROJ\_FOLDER” é a pasta em que o projeto foi criado e “PROJ\_NAME” é o nome dado ao projeto.

Para efetuar a carga, é necessário que a placa DETPIC32 esteja ligada ao computador por meio do cabo USB e que se use um programa adequado.

### 5.1 Bootloader da Microchip

A Microchip disponibiliza um *bootloader* que pode ser adaptado para a placa DETPIC32.

Este software, que inclui o código fonte, pode ser descarregado de:

[ww1.microchip.com/downloads/en/AppNotes/AN1388\\_Source\\_Code\\_011112.zip](http://ww1.microchip.com/downloads/en/AppNotes/AN1388_Source_Code_011112.zip)

Instruções sobre processo de adaptação poderão ser solicitadas ao autor deste documento.

O software da Microchip inclui uma aplicação PC, criada com Visual C.Net, desenvolvida para ambiente Windows. Esta aplicação pode também ser utilizada em ambiente Linux, via Wine (Windows Emulator, <http://www.winehq.org/> )

No caso de se usar o ambiente Linux, é necessário:

1- Configurar uma porta COM

Em “~/wine/dosdevices/” criar um link entre com1 e o dispositivo linux

E.g. `ln -s /dev/ttyUSB0 com1`

## 2 - Ter permissões “rwx” em /dev/ttyUSBx

Após estas configurações, basta fazer “\$wine [PATH\_TO\_BOOTLOADER]/PIC32UBL.exe”, em que “PATH\_TO\_BOOTLOADER” representa a pasta para a qual foi copiada a aplicação PC, i.e., o ficheiro “PIC32UBL.exe” que faz parte do pacote de software acima referido.

Independentemente do ambiente ser Windows ou Linux, ao executar o programa acima referido, aparece a janela representada na Figura 8. O processo de carga consiste nos seguintes passos:

1. Carregar no botão “Connect”
2. Pressionar o botão de reset da placa DETPIC32  
Deverá aparecer uma mensagem do tipo  
“Device connected  
Bootloader Firmware Version: 1.0”
3. Carregar no botão “Load Hex File” e seleccionar o ficheiro “.hex” pretendido (ter em atenção a nota acima, relativa ao local onde o ficheiro “.hex”  
Deverá aparecer a mensagem:  
“Hex file loaded successfully”
4. Carregar no botão “Erase-Program-Verify”  
Deverá aparecer a mensagem:  
“Flash Erased  
Programming completed  
Verification successful”
5. Carregar no botão “Run Application” ou efetuar *reset* à placa para executar a aplicação
6. Findo este processo deverá efetuar-se o “Disconnect”.

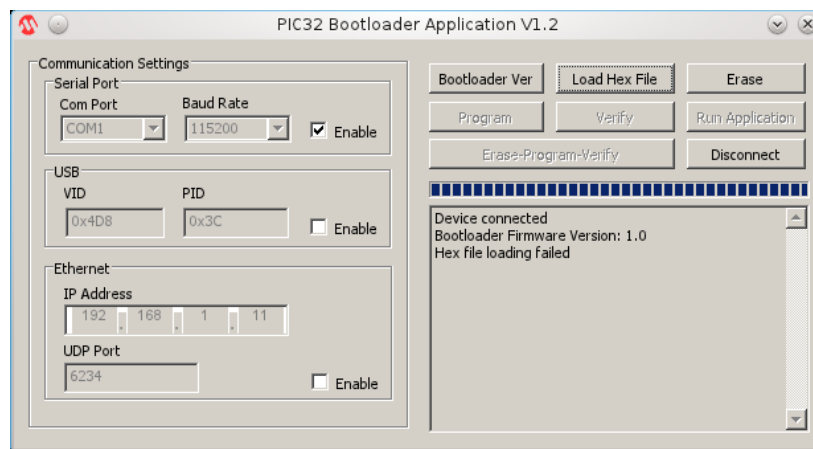


Figura 8

## 5.2 Aplicação PC multiOS – Java

Encontra-se também disponível uma aplicação Java, a qual funciona em Windows, Linux e Mac OS.



Esta variante tem a vantagem de funcionar em linha de comando, abrindo e fechando automaticamente a porta de comunicação, tornando-se assim mais prática.

A sua utilização é muito simples:

```
[PATH_TO_JAVA]java -jar bl.jar -p COM_INT -b -e -r -l XXXXXX.hex
```

em que:

- COM\_INT: identificação da interface série no sistema operativo
- -b: devolve a versão do bootloader
- -e: apaga a memória
- -r: executa o programa
- -l XXXXXX.hex: identificação do ficheiro a carregar (formato Intel hex)
- -h: ajuda

E.g.

```
/usr/java/jre1.7.0_15/bin/java -jar bl.jar -p /dev/ttyUSB0 -b -e -r -l blink.hex
```

## 6 Agradecimentos

O port do bootloader da Microchip para a plataforma DETPIC32 foi efetuado por:

- Diego Mendes - [diego.mendes@ua.pt](mailto:diego.mendes@ua.pt)
- Cristóvão Cruz - [cac@ua.pt](mailto:cac@ua.pt)
- Rómulo Antão - [romantao@gmail.com](mailto:romantao@gmail.com)

A “makefile” e exemplo associado foi desenvolvida por:

- Ricardo Marau - [rmarau@gmail.com](mailto:rmarau@gmail.com)