

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

E·XFl

Product Status	Active
Core Processor	MIPS32® M4K™
Core Size	32-Bit Single-Core
Speed	80MHz
Connectivity	I ² C, IrDA, LINbus, PMP, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	-
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	121-TFBGA
Supplier Device Package	121-TFBGA (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic32mx360f256lt-80v-bg

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

DOCUMENT LAYOUT

This document is divided into 6 sections. Sections 1 through 4 provide an overview of PIC32 products. Section 5 provides an overview of PIC32 development tools. Section 6 describes the procedure to setup a development environment and debug a sample application. The manual layout is as follows:

- Chapter 1. PIC32 Features
- Chapter 2. PIC32 Product Family
- Chapter 3. PIC32 Architecture
- Chapter 4. PIC32 Tools
- Chapter 5. Step-by-Step Procedures to Setup, Build, and Run a Demo
 Project
- Chapter 6. Technical Support Resources

CONVENTIONS USED IN THIS GUIDE

This manual uses the following documentation conventions:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic characters	Referenced books	MPLAB [®] IDE User's Guide
	Emphasized text	is the only compiler
Initial caps	A window	the Output window
	A dialog	the Settings dialog
	A menu selection	select Enable Programmer
Quotes	A field name in a window or dialog	"Save project before build"
Underlined, italic text with right angle bracket	A menu path	<u>File>Save</u>
Bold characters	A dialog button	Click OK
	A tab	Click the Power tab
N'Rnnnn	A number in verilog format, where N is the total number of digits, R is the radix and n is a digit.	4'b0010, 2'hF1
Text in angle brackets < >	A key on the keyboard	Press <enter>, <f1></f1></enter>
Courier New font:		
Plain Courier New	Sample source code	#define START
	Filenames	autoexec.bat
	File paths	c:\mcc18\h
	Keywords	_asm, _endasm, static
	Command-line options	-Opa+, -Opa-
	Bit values	0, 1
	Constants	OxFF, `A'
Italic Courier New	A variable argument	<i>file.o</i> , where <i>file</i> can be any valid filename
Square brackets []	Optional arguments	<pre>mcc18 [options] file [options]</pre>
Curly brackets and pipe character: { }	Choice of mutually exclusive arguments; an OR selection	errorlevel {0 1}
Ellipses	Replaces repeated text	<pre>var_name [, var_name]</pre>
	Represents code supplied by user	<pre>void main (void) { }</pre>

RECOMMENDED READING

This user's guide describes how to use PIC32. Other useful documents are listed below. The following Microchip documents are available and recommended as supplemental reference resources.

Readme for the PIC32MX

For the latest information on using PIC32 microcontrollers, read the file (an ASCII text file) at the root level of the CD included in the PIC32 Starter Kit. The file generally contains the most current update information, as well as any issues that may not have been available when this document was published.

Readme Files

For the latest information on using other tools, read the tool-specific readme files in the Readmes subdirectory of the PIC32MX Starter Kit installation directory. The files contain update information, as well as any issues that may not have been available when this document was published.

THE MICROCHIP WEB SITE

Microchip provides online support via our web site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- Product Support Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- General Technical Support Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- Business of Microchip Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

DEVELOPMENT SYSTEMS CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer Change Notification and follow the registration instructions.

The Development Systems product group categories are:

- Compilers The latest information on Microchip C compilers and other language tools. These include the MPLAB C18 and MPLAB C32 C compilers; and MPLAB LIB30 object librarians.
- **Emulators** The latest information on Microchip in-circuit emulators. This includes the MPLAB ICE 2000 and MPLAB ICE 4000.
- In-Circuit Debuggers The latest information on the Microchip in-circuit debugger, MPLAB ICD 2.
- MPLAB[®] IDE The latest information on Microchip MPLAB IDE, the Windows[®] Integrated Development Environment for development systems tools. This list is focused on the MPLAB IDE, MPLAB SIM simulator, MPLAB IDE Project Manager and general editing and debugging features.
- Programmers The latest information on Microchip programmers. These include the MPLAB PM3 and PRO MATE[®] II device programmers and the PICSTART[®] Plus, PICkit[®] 1 and PICkit[®] 2 development programmers.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- · Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://support.microchip.com

DOCUMENT REVISION HISTORY

Revision A (October 2007)

• Initial Release of this Document.

Revision B (March 2008)

Updates



Chapter 2. PIC32 Product Family

2.1 INTRODUCTION

The PIC32 family includes scalable devices ranging from 32KB to 512KB of Flash memory. Also a rich set of peripherals – Five timers, 16 channels of 10-bit A/D Converters and communication interfaces: SPI, I^2C^{TM} and UART.

Please consult the "*PIC32MX Family Data Sheet*" (DS61143) for a complete list of family variants, core and peripheral characteristics.





NOTES:

Communication channels:

- USB 2.0 compliant (FS. 12 Mbps), OTG, Host and Device-only capable
- 2 enhanced UART channels with hardware $IrDA^{\ensuremath{\mathbb{R}}}$
- · 2 Master/Slave/Frame mode SPI channels
- 2 Master/Slave I²C channels, 10/7 bits mode addressing, broadcast capable

Analog-to-Digital Converters:

- · Up to 16 Channels, each 10-bit resolution ADCs
- Up to 500+ kilo-samples per second (ksps) conversion speed
- · Software selectable Internal or External voltage reference
- Automatic Channel Scan mode
- Selectable conversion trigger source
- 16 word conversion result buffer
- Selectable Buffer Fill modes
- · Eight result alignment options
- · Operation during CPU Sleep mode

Timers:

- 5 16-bit timer/counter with the ability to form up to (2) 32-bit timer/counters
- · Software-selectable internal or external clock source
- Asynchronous timer/counter with built-in oscillator
- · Programmable interrupt generation and priority
- Gated external pulse counter
- Software-selectable prescalers.
- Operational during CPU Sleep mode

Core Timer:

• 32 bit timer in CPU for implementing a timer interrupt function.

RTCC (Real-Time Clock and Calendar):

- Time with hours, minutes and seconds
- Calendar with weekday, date, month and year
- Leap year detection
- Highly configurable alarm
- Calibration of up to 260 ppm of crystal error

Debug and Programming:

- 6 instructions and 2 data breakpoints
- 2 complex breakpoint logic blocks with qualified/primed breakpoint triggers, Pass counters, and stopwatch timers.
- 4-wire EJTAG and 2-wire Microchip interface
- 2-wire Microchip Interface:
 - 6 real-time read/write capture logic blocks
 - Read/write access to all data RAM and SFRs without stopping CPU
- Instruction Trace Port:
 - 5-Wire, nonintrusive trace port
 - Triggered by complex breakpoint logic block

GPIO:

- 5V tolerant inputs
- Individual output pin open-drain enable/disable
- · Individual input pin weak pull-up enable/disable
- · Monitor selective inputs and generate interrupt on mismatch condition
 - **Note:** Input pin weak pull-up and interrupt on mismatch features are available only on select I/O pins.

4.4 MPLAB C32 C COMPILER

The MPLAB C32 C Compiler package allows you to build your applications and contains the following key software components (see Figure 4-1).

- PIC32-gcc Compiler: complete ANSI C compiler with powerful integration capabilities and efficient code optimization. The compiler provides symbol information that is used by the MPLAB IDE debugger.
- **PIC32-gpp** Macro Processor: used automatically by the compiler to transform the program before compilation.
- PIC32-as Assembler: full-featured macro assembler.
- **PIC32-Id** Object Linker: links relocatable objects created by the assembler/compiler with those from precompiled libraries.
- **PIC32-ar** Archiver and Librarian: manages the creation and modification of library files of precompiled code.
- **PIC32-conv**: converts ELF executable files into ASCII or binary format, suitable for downloading onto a PROM programmer or evaluation board.

FIGURE 4-1: MPLAB[®] IDE DEVELOPMENT TOOLS COMPONENTS



4.5 PERIPHERAL LIBRARIES

PIC32 MCUs integrate a large number of on-chip high-performance peripherals.

To accelerate the usage of these peripherals, the MPLAB C32 compiler for PIC32 includes software peripheral libraries compatible with the 16-bit Microchip MCUs. The peripheral libraries are distributed in source and object format along with a detailed API description document. Software applications using MPLAB C32 compiler may call peripheral library functions by simply including the appropriate header file in their source files – the MPLAB C32 compiler has built-in knowledge of library header and archive files.

The peripheral include files are located in C:\Program Files\Microchip\MPLAB C32\pic32mx\include\peripheral and the full source code is located in C:\Program Files\Microchip\MPLAB C32\pic32-libs\peripheral location in corresponding peripheral sub-directory.

The Peripheral Library contains following include files:

Header File	Description
adc10.h	Library for the Analog-to-Digital converters support
bmx.h	Library for configuring the Bus Matrix internal buses, priorities and memory layout
cmp.h	Library for the Comparator modules support
cvref.h	Library for the Comparators Voltage reference support
dma.h	Library for the DMA Controller and CRC module support
nvm.h	Library providing access to Flash erase/program functions
i2c.h	Library for I ² C™ module support
incap.h	Library for the Input Capture module suport
int.h	Library for the vectored Interrupt Controller support
lock.h	Library for the system lock and unlock support
outcompare.h	Library for Output Compare module support
osc.h	Library for Oscillators support
pcache.h	Library for prefetch-cache module support
power.h	Library for Sleep and Idle operationsl power modes support
pmp.h	Library for Parallel Master Port support
ports.h	Library for configuring the I/O ports, reading or writing I/O data
reset.h	Library for Reset control and status
rtcc.h	Library for Real-Time Clock and Calendar support
system.h	Library for system level operations
spi.h	Library for Serial Peripheral Interface support
timer.h	Library to support 16 and 32-bit timers
uart.h	Library supporting the enhanced UART module
wdt.h	Library to support the Watchdog Timer module
	Header File adc10.h bmx.h bmx.h cmp.h cwref.h dma.h nvm.h il2c.h incap.h incap.h outcompare.h power.h power.h power.h spsten.h system.h spi.h uart.h

TABLE 4-1: PERIPHERAL LIBRARY

Some very simple code will be added to the template file to print a "Hello World..." string to a serial console connected to the evaluation board. We will use the UART Peripheral Library provided by Microchip.

- Build Project

The project will be built – causing the source files to be compiled and linked into machine code that can run on the selected PIC32 MCU.

- Test Code

Finally, the code will be tested by running it on the evaluation board.

Note: Some aspects of the user interface will change in future product releases and the screen shots in this guide may not exactly match the appearance of the MPLAB IDE desktop in later releases.

5.5 SELECTING THE DEVICE

Choose <u>Configure>Select Device</u> from the top IDE menu. In the Device dialog, select the PIC32 variant from the drop-down list.

Device:	Device Family:
FIG3ZWA360F31ZL	
	Microchip Tool Support
Programmers	
🥥 PICSTART Plus	O MPLAB ICD 2 🥥 PICkit 2
🥥 PRO MATE II	PICkit 1
MPLAB PM3	O MPLAB REALICE
Language and Desigr	n Tools
ASSEMBLER	OMPILER 🔾 VDI
Debuggers	
	MPLAB ICD 2 A PICkit 2
MPLAB REAL ICI	E
MPLABICE 200	IU MIPLABICE 4000 ICEJICD Headers
2	

FIGURE 5-2: MPLAB IDE SELECT DEVICE

The "lights" indicate which MPLAB IDE components support this device.

- A green light indicates full support.
- A yellow light indicates preliminary support for an upcoming part by the particular MPLAB IDE tool component. Components with a yellow light instead of a green light are often intended for early adopters of new parts who need quick support and understand that some operations or functions may not be available.
- A red light indicates no support for this device. Support may be forthcoming or inappropriate for the tool.

5.7 SETTING UP LANGUAGE TOOLS

Step Two of the Project Wizard sets up the language tools that are used with this project. Make sure the "Show all installed toolsuites" checkbox is checked. Select Microchip PIC32 C Compiler Toolsuite in the Active Toolsuite list box.

Then "MPLAB PIC32 Assembler (PIC32-as.exe)", "MPLAB PIC32 C Compiler (PIC32-gcc.exe)", "MPLAB PIC32 Object Linker (PIC32-Id.exe)", and "MPLAB PIC32 Archiver (PIC32-ar.exe)", should be visible in the Toolsuite Contents box. Click on each one to see its location. If MPLAB IDE was installed into the default directory, the paths for these files will be:

- for the MPLAB PIC32 assembler:
 - C:\Program Files\Microchip\MPLAB IDE\ MPLAB C32\bin\PIC32-as.exe
- for the MPLAB PIC32 compiler:
 - C:\Program Files\Microchip\MPLAB IDE\ MPLAB C32\bin\ PIC32-gcc.exe
- for the MPLAB PIC32 Object Linker:
 - C:\Program Files\Microchip\MPLAB IDE\ MPLAB C32\bin\PIC32-ld.exe
- for the MPLAB PIC32 Archiver:
 - C:\Program Files\Microchip\MPLAB IDE\ MPLAB C32\bin\PIC32-ar.exe

If these paths do not show up correctly, use the **Browse** button to set them to the proper files in the MPLAB IDE subfolders.

When you are finished, click Next>.

Select a lang	uage toolsuite
Active Toolsuite:	Microchip PIC32 C-Compiler Toolsuite
Toolsuite Conter	nts
MPLAB ASM MPLAB C32 MPLAB LIN MPLAB LIN MPLAB LIN	M32 Assembler (pic32-as.exe) 2 C Compiler (pic32-gcc.exe) K32 Object Linker (pic32-ld.exe) 32 Object Librarian/Archiver (pic32-ar.eve)
Location	s/Microchin/MPLAB C32/bin/nic32-acc.eve
Help! My !	Suite Isn't Listed!

5.8 NAMING THE PROJECT

Step Three of the wizard allows you to name the project and put it into a folder. This sample project will be called MyTestProject. Using the **Browse** button, place the project in a folder named Projects32. Click **Next>**.

Create a new project, or reco	nfigure the active project?		-7 _{iĝ}
- 🖲 Create New Project File —			
C:\Projects32\MyTestProject	24	Browse	
Beconfigure Active Project			
Make changes without s	aving		
C Save changes to existing) project file		
C Save changes to anothe	r project file		
		Browse	
· ·			-

FIGURE 5-5: MPLAB IDE NAMING THE PROJECT

Be sure to save the workspace in the Projects32 directory and name this workspace using the same name as for the project: MyTestProject.

Click Save.

The project space is now completely created and it should look like that:





Now we'll create our C source file. If the MPLAB IDE Editor window is not open, click <u>*File->New*</u> from the top menu or the New File menu shortcut on the standard toolbar. The Editor window will show up.

Let's type in a very simple Hello World program in the Editor window and save it in Projects32 directory as main.c. We'll use the UART Peripheral Library provided by Microchip:



To add the newly created main.c file to our project just right click on the Source Files folder in the MPLAB IDE MyTestProject.mcp window and select Add File. Browse and select main.c file from the Projects32 directory.

Note: If the Project window is not open, select View->Project from the top IDE menu.

The linker script that's needed for this project is selected automatically by the MPLAB IDE. We don't need to add any other file to our project.

FIGURE 5-9: MPLAB IDE MAIN.C FILE

Right click on MyTestProject.mcp window and select **Save**. The test project should be saved. The MyTestProject.mcp should look like:

🗋 😂 🖬 👗 🛤 📾 🗛 🛛	🗑 🏆 Checksum: 0×81eee98	Release 💌 💣 🖨 😡 🕕 🧐 🖽	
MyTestProject.mcw 💌	Unbilled main.c		×
 Copyrights22/Myr deshapec Copyrights22/Myr deshapec Copyrights22/Myr deshapec Copyr Files Copyr Files Copyr Files Copyr Files Copyr Files Copyr Files 	<pre>// this is a first test file finclude setting. file() finclude setting. file() fi</pre>	<pre>item Clock = 72 NH:)</pre>	2 01V_1
	Dutput		-
No.	Build Version Control Find in Files		

FIGURE 5-10: MPLAB IDE PROJECT SAVED

TIP: Files can be added and projects saved by using the right mouse button in the project window. In case of error, files can be manually deleted by selecting them and using the right mouse click menu.



FIGURE 5-14: MPLAB IDE BREAKPOINT REACHED

- 1. The debugging tool selected to connect to the demo board
- 2. The device selected on the target board
- 3. The menus available under the Debugger top menu
- 4. The Debug toolbar
- 5. The Source window, with the breakpoint reached
- 6. The Disassembly window
- 7. The Output window
- 8. The Program Memory window

Now, if you take a look to the PC Hyper Terminal window, you should see the output sent by our simple test program to the serial line:

FIGURE 5-15: HYPER TERMINAL RECEIVED CHARACTERS

Hello Wor	ld from PIC32		 		1
		-			

NOTES:

NOTES: