



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

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf726t-i-ss

Email: info@E-XFL.COM

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

Pin Diagrams – 44-PIN TQFP (PIC16F724/727/PIC16LF724/727)



3.6 Time-out Sequence

On power-up, the time-out sequence is as follows: first, PWRT time out is invoked after POR has expired, then OST is activated after the PWRT time out has expired. The total time out will vary based on oscillator configuration and \overrightarrow{PWRTE} bit status. For example, in EC mode with \overrightarrow{PWRTE} bit = 1 (\overrightarrow{PWRT} disabled), there will be no time out at all. Figure 3-4, Figure 3-5 and Figure 3-6 depict time-out sequences.

Since the time outs occur from the POR pulse, if $\overline{\text{MCLR}}$ is kept low long enough, the time outs will expire. Then, bringing $\overline{\text{MCLR}}$ high will begin execution immediately (see Figure 3-5). This is useful for testing purposes or to synchronize more than one PIC16(L)F722/3/4/6/7 device operating in parallel.

Table 3-3 shows the Reset conditions for some special registers.

3.7 Power Control (PCON) Register

The Power Control (PCON) register has two Status bits to indicate what type of Reset that last occurred.

Bit 0 is $\overline{\text{BOR}}$ (Brown-out Reset). $\overline{\text{BOR}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if $\overline{\text{BOR}} = 0$, indicating that a brown-out has occurred. The $\overline{\text{BOR}}$ Status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (BOREN<1:0> = 00 in the Configuration Word register).

Bit 1 is POR (Power-on Reset). It is a '0' on Power-on Reset and unaffected otherwise. The user must write a '1' to this bit following a Power-on Reset. On a subsequent Reset, if POR is '0', it will indicate that a Power-on Reset has occurred (i.e., VDD may have gone too low).

For more information, see Section 3.5 "Brown-Out Reset (BOR)".

Occillator Configuration	Powe	er-up	Brown-o	Wake-up from	
Oscillator Configuration	PWRTE = 0	PWRTE = 1	PWRTE = 0	PWRTE = 1	Sleep
XT, HS, LP ⁽¹⁾	TPWRT + 1024 • Tosc	1024 • Tosc	TPWRT + 1024 • Tosc	1024 • Tosc	1024 • Tosc
RC, EC, INTOSC	TPWRT	_	TPWRT		

TABLE 3-2: TIME OUT IN VARIOUS SITUATIONS

Note 1: LP mode with T1OSC disabled.

TABLE 3-3: RESET BITS AND THEIR SIGNIFICANCE

POR	BOR	то	PD	Condition
0	u	1	1	Power-on Reset
1	0	1	1	Brown-out Reset
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up
u	u	u	u	MCLR Reset during normal operation
u	u	1	0	MCLR Reset during Sleep

Legend: u = unchanged, x = unknown

EXAMPLE 4-1: SAVING W, STATUS AND PCLATH REGISTERS IN RAM

MOVWF	W_TEMP	;Copy W to W_TEMP register
SWAPF	STATUS,W	;Swap status to be saved into W
		;Swaps are used because they do not affect the status bits
BANKSEL	STATUS_TEMP	;Select regardless of current bank
MOVWF	STATUS_TEMP	;Copy status to bank zero STATUS_TEMP register
MOVF	PCLATH,W	;Copy PCLATH to W register
MOVWF	PCLATH_TEMP	;Copy W register to PCLATH_TEMP
:		
:(ISR)		;Insert user code here
:		
BANKSEL	STATUS_TEMP	;Select regardless of current bank
MOVF	PCLATH_TEMP,W	;
MOVWF	PCLATH	;Restore PCLATH
SWAPF	STATUS_TEMP,W	;Swap STATUS_TEMP register into W
		;(sets bank to original state)
MOVWF	STATUS	;Move W into STATUS register
SWAPF	W_TEMP,F	;Swap W_TEMP
SWAPF	W_TEMP,W	;Swap W_TEMP into W

4.5.1 INTCON REGISTER

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, PORTB change and external RB0/INT/SEG0 pin interrupts.

```
Note: Interrupt flag bits are set when an interrupt
condition occurs, regardless of the state of
its corresponding enable bit or the global
enable bit, GIE of the INTCON register.
User software should ensure the
appropriate interrupt flag bits are clear
prior to enabling an interrupt.
```

4.5.3 PIE2 REGISTER

Γ.

bit 0

The PIE2 register contains the interrupt enable bits, as shown in Register 4-3.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 4-3: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	CCP2IE
bit 7							bit 0

Legena:			
R = Readable bit W	= Writable bit	U = Unimplemented bit, read	as '0'
-n = Value at POR '1'	= Bit is set	0' = Bit is cleared	x = Bit is unknown

bit 7-1 Unimplemented: Read as '0'

CCP2IE: CCP2 Interrupt Enable bit

1 = Enables the CCP2 interrupt

0 = Disables the CCP2 interrupt

6.2 PORTA and the TRISA Registers

PORTA is a 8-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 6-3). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 6-1 shows how to initialize PORTA.

Reading the PORTA register (Register 6-2) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch.

The TRISA register (Register 6-3) controls the PORTA pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the

REGISTER 6-2: PORTA: PORTA REGISTER

TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

Note:	The ANSELA register must be initialized					
	to configure an analog channel as a digital					
	input. Pins configured as analog inputs					
	will read '0'.					

EXAMP	LE 6-1:	INITIALIZING PORTA
BANKSEL	DORTA	;
CLRF	PORTA	;Init PORTA
BANKSEL	ANSELA	;
CLRF	ANSELA	;digital I/O
BANKSEL	TRISA	i
MOVLW	0Ch	;Set RA<3:2> as inputs
MOVWF	TRISA	;and set RA<7:4,1:0>
		;as outputs

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
bit 7							bit 0
Logond							

Legend.			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as	ʻ0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

RA<7:0>: PORTA I/O Pin bit 1 = Port pin is > VIH 0 = Port pin is < VIL

REGISTER 6-3: TRISA: PORTA TRI-STATE REGISTER

| R/W-1 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| TRISA7 | TRISA6 | TRISA5 | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 |
| bit 7 | | | | | | | bit 0 |

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as	ʻO'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0

bit 7-0

TRISA<7:0>: PORTA Tri-State Control bit

1 = PORTA pin configured as an input (tri-stated)

0 = PORTA pin configured as an output





Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	00 0000	00 0000
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	11 1111	11 1111
APFCON	—	—	—	—	—	_	SSSEL	CCP2SEL	00	00
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	00 0000	00 0000
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	TOXCS	0 0000	0 0000
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	0000	0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000	0000 0000
OPTION_REG	RBPU	INTEDG	TOCS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	XXXX XXXX	XXXX XXXX
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T <u>1GGO</u> / DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	uuuu uxuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	1111 1111

TABLE 6-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTB.

11.0 TIMER0 MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (shared with Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Interrupt on overflow
- TMR0 can be used to gate Timer1

Figure 11-1 is a block diagram of the Timer0 module.

11.1 Timer0 Operation

The Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

11.1.1 8-BIT TIMER MODE

The Timer0 module will increment every instruction cycle, if used without a prescaler. 8-Bit Timer mode is selected by clearing the T0CS bit of the OPTION register.

When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note: The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

11.1.2 8-BIT COUNTER MODE

In 8-Bit Counter mode, the Timer0 module will increment on every rising or falling edge of the T0CKI pin or the Capacitive Sensing Oscillator (CPSOSC) signal.

8-Bit Counter mode using the T0CKI pin is selected by setting the T0CS bit in the OPTION register to '1' and resetting the T0XCS bit in the CPSCON0 register to '0'.

8-Bit Counter mode using the Capacitive Sensing Oscillator (CPSOSC) signal is selected by setting the TOCS bit in the OPTION register to '1' and setting the T0XCS bit in the CPSCON0 register to '1'.

The rising or falling transition of the incrementing edge for either input source is determined by the T0SE bit in the OPTION register.

FIGURE 11-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



12.11 Timer1 Control Register

The Timer1 Control register (T1CON), shown in Register 12-1, is used to control Timer1 and select the various features of the Timer1 module.

REGISTER 12-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	_	TMR1ON
bit 7							bit 0

Legend:					
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'			
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown		

bit 7-6	TMR1CS<1:0>: Timer1 Clock Source Select bits
	11 = Timer1 clock source is Capacitive Sensing Oscillator (CAPOSC)
	10 = Timer1 clock source is pin or oscillator:
	$\underline{If T1OSCEN} = \underline{0}:$
	External clock from T1CKI pin (on the rising edge)
	$\frac{\text{If } 110\text{SCEN} = 1}{\text{Crystel equilator on T10SI/T10SO pine}}$
	01 – Timer1 clock source is system clock (FOSC)
	00 = Timer1 clock source is instruction clock (FOSC/4)
bit 5-4	T1CKPS<1:0>: Timer1 Input Clock Prescale Select bits
	11 = 1.8 Prescale value
	10 = 1:4 Prescale value
	01 = 1:2 Prescale value
	00 = 1:1 Prescale value
bit 3	T1OSCEN: LP Oscillator Enable Control bit
	1 = Dedicated Timer1 oscillator circuit enabled
	0 = Dedicated Timer1 oscillator circuit disabled
bit 2	T1SYNC: Timer1 External Clock Input Synchronization Control bit
	<u>TMR1CS<1:0> = $1X$</u>
	1 = Do not synchronize external clock input
	0 = Synchronize external clock input with system clock (FOSC)
	TMR1CS<1:0> = 0X
	This bit is ignored. Timer1 uses the internal clock when TMR1CS<1:0> = $1X$.
bit 1	Unimplemented: Read as '0'
bit 0	TMR1ON: Timer1 On bit
	1 = Enables Timer1
	0 = Stops Timer1
	Clears Timer1 Gate flip-flop

REGISTER 17-5: SSPMSK: SSP MASK REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1				
MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0				
bit 7							bit 0				
Legend:											
R = Readable	bit	W = Writable bit		U = Unimplemented bit, read as '0'							
-n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is					x = Bit is unkno	is unknown					
bit 7-1	MSK<7:1>: M	ask bits									
	1 = The recei	ved address bit n	is compared to	o SSPADD <n> to detect I²C addres</n>	detect I ² C addr	ess match					

	0 = The received address bit his hot used to detect if C address match
bit 0	MSK<0>: Mask bit for I ² C Slave Mode, 10-bit Address
	I ² C Slave Mode, 10-bit Address (SSPM<3:0> = 0111):
	1 = The received address bit '0' is compared to SSPADD<0> to detect I^2C address match
	0 = The received address bit '0' is not used to detect I^2C address match
	All other SSP modes: this bit has no effect.

REGISTER 17-6: SSPADD: SSP I²C ADDRESS REGISTER

| R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| ADD7 | ADD6 | ADD5 | ADD4 | ADD3 | ADD2 | ADD1 | ADD0 |
| bit 7 | | | | | | | bit 0 |
| | | | | | | | |

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as	ʻ0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 ADD<7:0>: Address bits Received address

TABLE 17-7: REGISTERS ASSOCIATED WITH I²C OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
SSPADD	Synchronous	Serial Por	t (I ² C mod	e) Address	s Register				0000 0000	0000 0000
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPMSK ⁽²⁾	(⁽²⁾ Synchronous Serial Port (I ² C mode) Address Mask Register							1111 1111	1111 1111	
SSPSTAT	SMP ⁽¹⁾	CKE ⁽¹⁾	D/A	Р	S	R/W	UA	BF	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by SSP module in I²C mode.

Note 1: Maintain these bits clear in I^2C mode.

2: Accessible only when SSPM<3:0> = 1001.

20.0 IN-CIRCUIT SERIAL PROGRAMMING[™] (ICSP[™])

ICSP[™] programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP[™] programming:

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VDD
- Vss

The device is placed into Program/Verify mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on MCLR/VPP from 0v to VPP. In Program/Verify mode the program memory, User IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ISCPCLK pin is the clock input. For more information on ICSP, refer to the "*PIC16(L)F72x Memory Programming Specification*" (DS41332).

Note: The ICD 2 produces a VPP voltage greater than the maximum VPP specification of the PIC16(L)F722/3/4/6/7. When using this programmer, an external circuit, such as the AC164112 MPLAB ICD 2 VPP voltage limiter, is required to keep the VPP voltage within the device specifications.



FIGURE 20-1: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING

22.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM[™] and dsPICDEM[™] demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ[®] security ICs, CAN, IrDA[®], PowerSmart battery management, SEEVAL[®] evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

22.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent[®] and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika[®]











Note 1: If the A/D clock source is selected as RC, a time of TcY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.



FIGURE 24-3: PIC16F722/3/4/6/7 TYPICAL IDD vs. Fosc OVER VDD, EC MODE, VCAP = 0.1 µF



























© 2007-2015 Microchip Technology Inc.



FIGURE 24-49: TTL INPUT THRESHOLD VIN vs. VDD OVER TEMPERATURE





44-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

	MILLIM	ETERS		
Dimension	MIN	NOM	MAX	
Contact Pitch	E		0.80 BSC	
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076A