



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	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 14x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f727-i-p

Email: info@E-XFL.COM

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

3.0 RESETS

The PIC16(L)F722/3/4/6/7 differentiates between various kinds of Reset:

- a) Power-on Reset (POR)
- b) WDT Reset during normal operation
- c) WDT Reset during Sleep
- d) MCLR Reset during normal operation
- e) MCLR Reset during Sleep
- f) Brown-out Reset (BOR)

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on:

- Power-on Reset (POR)
- MCLR Reset
- MCLR Reset during Sleep
- WDT Reset
- Brown-out Reset (BOR)

Most registers are not affected by a WDT wake-up since this is viewed as the resumption of normal operation. TO and PD bits are set or cleared differently in different Reset situations, as indicated in Table 3-3. These bits are used in software to determine the nature of the Reset.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 3-1.

The MCLR Reset path has a noise filter to detect and ignore small pulses. See **Section 23.0** "**Electrical Specifications**" for pulse-width specifications.

FIGURE 3-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



TABLE 3-5: INITIALIZATION CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	0000h	0001 1xxx	0x
MCLR Reset during normal operation	0000h	000u uuuu	uu
MCLR Reset during Sleep	0000h	0001 Ouuu	uu
WDT Reset	0000h	0000 uuuu	uu
WDT Wake-up	PC + 1	uuu0 0uuu	uu
Brown-out Reset	0000h	0001 1xxx	10
Interrupt Wake-up from Sleep	PC + 1 ⁽¹⁾	uuul Ouuu	uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

Note 1: When the wake-up is due to an interrupt and Global Interrupt Enable bit, GIE, is set, the PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

TABLE 3-6: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS

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 ⁽¹⁾
STATUS	IRP	RP1	RP0	TO	PD	Z	DC	С	0001 1xxx	000q quuu
PCON	—	—	—	—	—	—	POR	BOR	dd	uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition. Shaded cells are not used by Resets.

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.









7.2 Clock Source Modes

Clock source modes can be classified as external or internal.

- Internal clock source (INTOSC) is contained within the oscillator module and derived from a 500 kHz high precision oscillator. The oscillator module has eight selectable output frequencies, with a maximum internal frequency of 16 MHz.
- External clock modes rely on external circuitry for the clock source. Examples are: oscillator modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.

The system clock can be selected between external or internal clock sources via the FOSC bits of the Configuration Word 1.

7.3 Internal Clock Modes

The oscillator module has eight output frequencies derived from a 500 kHz high precision oscillator. The IRCF bits of the OSCCON register select the postscaler applied to the clock source dividing the frequency by 1, 2, 4 or 8. Setting the PLLEN bit of the Configuration Word 1 locks the internal clock source to 16 MHz before the postscaler is selected by the IRCF bits. The PLLEN bit must be set or cleared at the time of programming; therefore, only the upper or low four clock source frequencies are selectable in software.

7.3.1 INTOSC AND INTOSCIO MODES

The INTOSC and INTOSCIO modes configure the internal oscillators as the system clock source when the device is programmed using the oscillator selection or the FOSC<2:0> bits in the CONFIG1 register. See **Section 8.0** "**Device Configuration**" for more information.

In INTOSC mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT outputs the selected internal oscillator frequency divided by 4. The CLKOUT signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements.

In INTOSCIO mode, OSC1/CLKIN and OSC2/CLKOUT are available for general purpose I/O.

7.3.2 FREQUENCY SELECT BITS (IRCF)

The output of the 500 kHz INTOSC and 16 MHz INTOSC, with Phase-Locked Loop enabled, connect to a postscaler and multiplexer (see Figure 7-1). The Internal Oscillator Frequency Select bits (IRCF) of the OSCCON register select the frequency output of the internal oscillator. Depending upon the PLLEN bit, one of four frequencies of two frequency sets can be selected via software:

If PLLEN = 1, frequency selection is as follows:

- 16 MHz
- 8 MHz (default after Reset)
- 4 MHz
- 2 MHz
- If PLLEN = 0, frequency selection is as follows:
- 500 kHz
- 250 kHz (default after Reset)
- 125 kHz
- 62.5 kHz

Note: Following any Reset, the IRCF<1:0> bits of the OSCCON register are set to '10' and the frequency selection is set to 8 MHz or 250 kHz. The user can modify the IRCF bits to select a different frequency.

There is no start-up delay before a new frequency selected in the IRCF bits takes effect. This is because the old and new frequencies are derived from INTOSC via the postscaler and multiplexer.

Start-up delay specifications are located in the Table 23-2 in Section 23.0 "Electrical Specifications".

7.5 Oscillator Tuning

The INTOSC is factory calibrated but can be adjusted in software by writing to the OSCTUNE register (Register 7-2). The default value of the OSCTUNE register is '0'. The value is a 6-bit two's complement number.

When the OSCTUNE register is modified, the INTOSC frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

REGISTER 7-2: OSCTUNE: OSCILLATOR TUNING REGISTER

U-0	U-0	U-0 R/W-0		R/W-0	R/W-0	R/W-0	R/W-0
_	— — TUN5 TI		TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

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

bit 7-6 Unimplemented: Read as '0'

bit 5-0

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
ADCON1	_	ADCS2	ADCS1	ADCS0	_	_	ADREF1	ADREF0	-00000	-00000
ANSELA	_	_	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	11 1111	11 1111
ANSELB	_	_	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	11 1111	11 1111
ANSELE	_	_	_	_	_	ANSE2	ANSE1	ANSE0	111	111
ADRES				A/D Result	Register Byte	e			xxxx xxxx	uuuu uuuu
CCP2CON	_	_	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	00 0000	00 0000
FVRCON	FVRRDY	FVREN	_	_	_	_	ADFVR1	ADFVR0	q000	d000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISE	_	_	_	_	TRISE3	TRISE2	TRISE1	TRISE0	1111	1111

TABLE 9-2: SUMMARY OF ASSOCIATED ADC REGISTERS

Legend: x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends on condition. Shaded cells are not used for ADC module.

10.0 FIXED VOLTAGE REFERENCE

This device contains an internal voltage regulator. To provide a reference for the regulator, a band gap reference is provided. This band gap is also user accessible via an A/D converter channel.

User level band gap functions are controlled by the FVRCON register, which is shown in Register 10-1.

REGISTER 10-1: FVRCON: FIXED VOLTAGE REFERENCE REGISTER

R-q	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0		
FVRRDY	FVREN	—		—	—	ADFVR1	ADFVR0		
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					
q = Value depe	ends on condition	on							

bit 7	FVRRDY⁽¹⁾: Fixed Voltage Reference Ready Flag bit 0 = Fixed Voltage Reference output is not active or stable 1 = Fixed Voltage Reference output is ready for use
bit 6	FVREN⁽²⁾: Fixed Voltage Reference Enable bit
	0 = Fixed Voltage Reference is disabled1 = Fixed Voltage Reference is enabled
bit 5-2	Unimplemented: Read as '0'
bit 1-0	ADFVR<1:0>: A/D Converter Fixed Voltage Reference Selection bits
	00 = A/D Converter Fixed Voltage Reference Peripheral output is off. 01 = A/D Converter Fixed Voltage Reference Peripheral output is 1x (1.024V) 10 = A/D Converter Fixed Voltage Reference Peripheral output is 2x (2.048V) ⁽²⁾ 11 = A/D Converter Fixed Voltage Reference Peripheral output is 4x (4.096V) ⁽²⁾
Note 1:	FVRRDY is always '1' for the PIC16F72X devices.

2: Fixed Voltage Reference output cannot exceed VDD.



FIGURE 12-4: TIMER1 GATE TOGGLE MODE



FIGURE 12-5:	TIMER1 GATE SINGLE-PULSE MODE
TMR1GE	
T1GPOL	
T1GSPM	
T1GG <u>O/</u> DONE	Cleared by hardware on falling edge of T1GVAL
T1G_IN	rising edge of T1G
Т1СКІ	
T1GVAL	
TIMER1	N N + 1 N + 2
TMR1GIF	Cleared by software Cleared by hardware on falling edge of T1GVAL

14.0 CAPACITIVE SENSING MODULE

The capacitive sensing module allows for an interaction with an end user without a mechanical interface. In a typical application, the capacitive sensing module is attached to a pad on a printed circuit board (PCB), which is electrically isolated from the end user. When the end user places their finger over the PCB pad, a capacitive load is added, causing a frequency shift in the capacitive sensing module. The capacitive sensing module requires software and at least one timer resource to determine the change in frequency. Key features of this module include:

- · Analog MUX for monitoring multiple inputs
- · Capacitive sensing oscillator
- Multiple timer resources
- Software control
- · Operation during Sleep

FIGURE 14-1: CAPACITIVE SENSING BLOCK DIAGRAM



R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R-0	R/W-0			
CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	TOXCS			
bit 7							bit 0			
Legend:										
R = Readable I	bit	W = Writable	bit	U = Unimpler	mented bit, read	l as '0'				
-n = Value at P	OR	'1' = Bit is set		'0' = Bit is cle	ared	x = Bit is unkr	nown			
 bit 7 CPSON: Capacitive Sensing Module Enable bit 1 = Capacitive sensing module is operating 0 = Capacitive sensing module is shut off and consumes no operating current 										
bit 6-4	Unimplemen	ted: Read as '	כ'							
bit 3-2	bit 3-2 CPSRNG<1:0>: Capacitive Sensing Oscillator Range bits 00 = Oscillator is Off. 01 = Oscillator is in low range. Charge/discharge current is nominally 0.1 μA. 10 = Oscillator is in medium range. Charge/discharge current is nominally 1.2 μA. 11 = Oscillator is in bidb range. Charge/discharge current is nominally 1.8 μA.									
bit 1	bit 1 CPSOUT: Capacitive Sensing Oscillator Status bit 1 = Oscillator is sourcing current (Current flowing out the pin) 0 = Oscillator is sinking current (Current flowing into the pin)									
bit 0 TOXCS: Timer0 External Clock Source Select bit $\frac{\text{If TOCS} = 1}{\text{The TOXCS bit controls which clock external to the core/Timer0 module supplies Timer0:}$ $1 = \text{Timer0 Clock Source is the capacitive sensing oscillator}$ $0 = \text{Timer0 Clock Source is the TOCKI pin}$ $\frac{\text{If TOCS} = 0}{\text{Timer0 clock source is controlled by the core/Timer0 module and is Fosc/4.}}$										

REGISTER 14-1: CPSCON0: CAPACITIVE SENSING CONTROL REGISTER 0

16.0 ADDRESSABLE UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (AUSART)

The Addressable Universal Synchronous Asynchronous Receiver Transmitter (AUSART) module is a serial I/O communications peripheral. It contains all the clock generators, shift registers and data buffers necessary to perform an input or output serial data transfer independent of device program execution. The AUSART, also known as a Serial Communications Interface (SCI), can be configured as a full-duplex asynchronous system or half-duplex synchronous system. Full-Duplex mode is useful for communications with peripheral systems, such as CRT terminals and personal computers. Half-Duplex Synchronous mode is intended for communications with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs or other microcontrollers. These devices typically do not have internal clocks for baud rate generation and require the external clock signal provided by a master synchronous device.

The AUSART module includes the following capabilities:

- Full-duplex asynchronous transmit and receive
- Two-character input buffer
- One-character output buffer
- Programmable 8-bit or 9-bit character length
- Address detection in 9-bit mode
- Input buffer overrun error detection
- Received character framing error detection
- Half-duplex synchronous master
- Half-duplex synchronous slave
- Sleep operation

Block diagrams of the AUSART transmitter and receiver are shown in Figure 16-1 and Figure 16-2.

FIGURE 16-1: AUSART TRANSMIT BLOCK DIAGRAM



16.1.2.8 Asynchronous Reception Set-up:

- Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (refer to Section 16.2 "AUSART Baud Rate Generator (BRG)").
- 2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
- 3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 4. If 9-bit reception is desired, set the RX9 bit.
- 5. Enable reception by setting the CREN bit.
- The RCIF interrupt flag bit of the PIR1 register will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE bit of the PIE1 register was also set.
- 7. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
- 8. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register.
- 9. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

16.1.2.9 9-bit Address Detection Mode Set-up

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

- Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (refer to Section 16.2 "AUSART Baud Rate Generator (BRG)").
- 2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
- 3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 4. Enable 9-bit reception by setting the RX9 bit.
- 5. Enable address detection by setting the ADDEN bit.
- 6. Enable reception by setting the CREN bit.
- 7. The RCIF interrupt flag bit of the PIR1 register will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit of the PIE1 register was also set.
- 8. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
- 9. Get the received eight Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
- 10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
- 11. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

RX/DT pin	Start bit / bit 0 / bit 1 / (/ ybit 7/8/ Stop / bit / bit 0 / (/ ybit 7/8/ Stop / bit / / / ybit 7/8/ Stop
Rcv Shift Reg Rcv Buffer Reg	Word 1 Word 2 Word 2 PCPEC
Read Rcv Buffer Reg – RCREG	
RCIF (Interrupt Flag)	
OERR bit _ CREN _	
Note: This t causir	iming diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, ng the OERR (overrun) bit to be set.

FIGURE 16-5: ASYNCHRONOUS RECEPTION

16.3 AUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective receive and transmit shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The AUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

16.3.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the AUSART for Synchronous Master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the AUSART.

16.3.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/ CK line. The TX/CK pin output driver is automatically enabled when the AUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

16.3.1.2 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the AUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character, the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note: The TSR register is not mapped in data memory, so it is not available to the user.

16.3.1.3 Synchronous Master Transmission Setup:

- Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (refer to Section 16.2 "AUSART Baud Rate Generator (BRG)").
- 2. Enable the synchronous master serial port by setting bits SYNC, SPEN, and CSRC.
- 3. Disable Receive mode by clearing bits SREN and CREN.
- 4. Enable Transmit mode by setting the TXEN bit.
- 5. If 9-bit transmission is desired, set the TX9 bit.
- 6. If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
- 7. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
- 8. Start transmission by loading data to the TXREG register.

16.3.2 SYNCHRONOUS SLAVE MODE

The following bits are used to configure the AUSART for Synchronous slave operation:

- SYNC = 1
- CSRC = 0
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Clearing the CSRC bit of the TXSTA register configures the device as a slave. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the AUSART.

16.3.2.1 AUSART Synchronous Slave Transmit

The operation of the Synchronous Master and Slave modes are identical (refer to **Section 16.3.1.2 "Synchronous Master Transmission")**, except in the case of the Sleep mode. If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

- 1. The first character will immediately transfer to the TSR register and transmit.
- 2. The second word will remain in TXREG register.
- 3. The TXIF bit will not be set.
- After the first character has been shifted out of TSR, the TXREG register will transfer the second character to the TSR and the TXIF bit will now be set.
- If the PEIE and TXIE bits are set, the interrupt will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine.
- 16.3.2.2 Synchronous Slave Transmission Setup:
- 1. Set the SYNC and SPEN bits and clear the CSRC bit.
- 2. Clear the CREN and SREN bits.
- If using interrupts, ensure that the GIE and PEIE bits of the INTCON register are set and set the TXIE bit.
- 4. If 9-bit transmission is desired, set the TX9 bit.
- 5. Enable transmission by setting the TXEN bit.
- 6. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
- 7. If 9-bit transmission is selected, insert the Most Significant bit into the TX9D bit.
- 8. Start transmission by writing the Least Significant eight bits to the TXREG register.

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	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000X	0000 000X
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART Transmit Data Register									0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

TABLE 16-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Transmission.

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
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	11 1111	11 1111
APFCON	-	—	—	—	-	-	SSSEL	CCP2SEL	00	00
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PR2	Timer2 Period Register						1111 1111	1111 1111		
SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register					XXXX XXXX	uuuu uuuu			
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPSTAT	SMP	CKE	D/A	Р	S	R/W	UA	BF	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
T2CON		TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000

TABLE 17-1: SUMMARY OF REGISTERS ASSOCIATED WITH SPI OPERATION

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

RLF	Rotate	Rotate Left f through Carry					
Syntax:	[label]] RLF	f,d				
Operands:	$0 \le f \le d \in [0,]$	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$					
Operation:	See de	See description below					
Status Affected:	С	С					
Description:	The co rotated the Ca result is If 'd' is back in	The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.					
Words:	1						
Cycles:	1	1					
Example:	RLF	REG1	,0				
	Before Instruction						
		REG1	=	1110	0110		
		C	=	0			
	After In	struction					
		REG1	=	1110	0110		
		W	=	1100	1100		
		С	=	1			

SLEEP	Enter Sleep mode			
Syntax:	[label] SLEEP			
Operands:	None			
Operation:	$\begin{array}{l} \text{00h} \rightarrow \text{WDT,} \\ 0 \rightarrow \text{WDT prescaler,} \\ 1 \rightarrow \overline{\text{TO}}, \\ 0 \rightarrow \overline{\text{PD}} \end{array}$			
Status Affected:	TO, PD			
Description:	The power-down Status bit, \overline{PD} is cleared. Time-out Status bit, \overline{TO} is set. Watchdog Timer and its prescaler are cleared. The processor is put into Sleep mode with the oscillator stopped.			

RRF	Rotate Right f through Carry				
Syntax:	[<i>label</i>] RRF f,d				
Operands:	$\begin{array}{l} 0\leq f\leq 127\\ d\in [0,1] \end{array}$				
Operation:	See description below				
Status Affected:	С				
Description:	The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.				
	C Register f				

SUBLW	Subtract W from literal				
Syntax:	[<i>label</i>] SUBLW k				
Operands:	$0 \leq k \leq 255$				
Operation:	$k \text{ - } (W) \rightarrow (W)$				
Status Affected:	C, DC, Z				
Description:	The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.				
	C = 0	W > k			
	C = 1	$W \le k$			

DC = 0

DC = 1

W<3:0> > k<3:0>

 $W < 3:0 > \le k < 3:0 >$



FIGURE 24-13: PIC16F722/3/4/6/7 MAXIMUM IDD vs. VDD OVER Fosc, XT MODE, VCAP = 0.1 µF







FIGURE 24-61: PIC16F722/3/4/6/7 A/D INTERNAL RC OSCILLATOR PERIOD



FIGURE 24-62: PIC16F722/3/4/6/7 CAP SENSE OUTPUT CURRENT, POWER MODE = HIGH



FIGURE 24-65: PIC16F722/3/4/6/7 CAP SENSOR HYSTERESIS, POWER MODE = HIGH



