



Welcome to [E-XFL.COM](https://www.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 "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	ECANbus, I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.6K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf26k80t-i-so

PIC18F66K80 FAMILY

TABLE 5-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices			Power-on Reset, Brown-out Reset	MCLR Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
RXF3EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF3SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF3SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF2EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF2EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF2SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF2SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF1EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF1EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF1SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF1SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF0EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF0EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXF0SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxx- x-xx	uuu- u-uu	uuu- u-uu
RXF0SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
CANCON_RO4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
CANSTAT_RO4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
B5D7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D4	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D3	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D2	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D1	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5D0	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5DLC	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	-xxx xxxx	-uuu uuuu	uuuu uuuu
B5EIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5EIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5SIDL	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx x-xx	uuuu u-uu	uuuu u-uu
B5SIDH	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B5CON	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	0000 0000	0000 0000	uuuu uuuu
CANCON_RO5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
CANSTAT_RO5	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	1000 0000	1000 0000	uuuu uuuu
B4D7	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu
B4D6	PIC18F2XK80	PIC18F4XK80	PIC18F6XK80	xxxx xxxx	uuuu uuuu	uuuu uuuu

Legend: u = unchanged; x = unknown; - = unimplemented bit, read as '0'; q = value depends on condition.
Shaded cells indicate conditions do not apply for the designated device.

- Note 1:** One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).
- 2:** When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).
- 3:** When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.
- 4:** See Table 5-3 for Reset value for specific conditions.
- 5:** Bits 6 and 7 of PORTA, LATA and TRISA are enabled, depending on the oscillator mode selected. When not enabled as PORTA pins, they are disabled and read as '0'.

8.0 DATA EEPROM MEMORY

The data EEPROM is a nonvolatile memory array, separate from the data RAM and program memory, that is used for long-term storage of program data. It is not directly mapped in either the register file or program memory space, but is indirectly addressed through the Special Function Registers (SFRs). The EEPROM is readable and writable during normal operation over the entire VDD range.

Five SFRs are used to read and write to the data EEPROM, as well as the program memory. They are:

- EECON1
- EECON2
- EEDATA
- EEADR
- EEADRH

The data EEPROM allows byte read and write. When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and the EEADRH:EEADR register pair holds the address of the EEPROM location being accessed.

The EEPROM data memory is rated for high erase/write cycle endurance. A byte write automatically erases the location and writes the new data (erase-before-write). The write time is controlled by an on-chip timer; it will vary with voltage and temperature, as well as from chip-to-chip. Please refer to Parameter D122 (Table 31-1 in **Section 31.0 “Electrical Characteristics”**) for exact limits.

8.1 EEADR and EEADRH Registers

The EEADRH:EEADR register pair is used to address the data EEPROM for read and write operations. EEADRH holds the two MSBs of the address; the upper 6 bits are ignored. The 10-bit range of the pair can address a memory range of 1024 bytes (00h to 3FFh).

8.2 EECON1 and EECON2 Registers

Access to the data EEPROM is controlled by two registers: EECON1 and EECON2. These are the same registers which control access to the program memory and are used in a similar manner for the data EEPROM.

The EECON1 register (Register 8-1) is the control register for data and program memory access. Control bit, EEPGD, determines if the access will be to program memory or data EEPROM memory. When clear, operations will access the data EEPROM memory. When set, program memory is accessed.

Control bit, CFGS, determines if the access will be to the Configuration registers or to program memory/data EEPROM memory. When set, subsequent operations access Configuration registers. When CFGS is clear, the EEPGD bit selects either program Flash or data EEPROM memory.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set in hardware when the WREN bit is set and cleared, when the internal programming timer expires and the write operation is complete.

Note: During normal operation, the WRERR is read as '1'. This can indicate that a write operation was prematurely terminated by a Reset, or a write operation was attempted improperly.

The WR control bit initiates write operations. The bit cannot be cleared, only set, in software; it is cleared in hardware at the completion of the write operation.

Note: The EEIF interrupt flag bit (PIR4<6>) is set when the write is complete. It must be cleared in software.

Control bits, RD and WR, start read and erase/write operations, respectively. These bits are set by firmware and cleared by hardware at the completion of the operation.

The RD bit cannot be set when accessing program memory (EEPGD = 1). Program memory is read using table read instructions. See **Section 7.1 “Table Reads and Table Writes”** regarding table reads.

The EECON2 register is not a physical register. It is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

PIC18F66K80 FAMILY

REGISTER 10-10: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
OSCFIE	—	—	—	BCLIE	HLVDIE	TMR3IE	TMR3GIE
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 **OSCFIE:** Oscillator Fail Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 6-4 **Unimplemented:** Read as '0'

bit 3 **BCLIE:** Bus Collision Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 2 **HLVDIE:** High/Low-Voltage Detect Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 1 **TMR3IE:** TMR3 Overflow Interrupt Enable bit

1 = Enabled

0 = Disabled

bit 0 **TMR3GIE:** Timer3 Gate Interrupt Enable bit

1 = Enabled

0 = Disabled

PIC18F66K80 FAMILY

10.6 INTx Pin Interrupts

External interrupts on the RB0/INT0, RB1/INT1, RB2/INT2 and RB3/INT3 pins are edge-triggered. If the corresponding INTEDGx bit in the INTCON2 register is set (= 1), the interrupt is triggered by a rising edge. If that bit is clear, the trigger is on the falling edge.

When a valid edge appears on the RBx/INTx pin, the corresponding flag bit, INTxIF, is set. This interrupt can be disabled by clearing the corresponding enable bit, INTxIE. Before re-enabling the interrupt, the flag bit (INTxIF) must be cleared in software in the Interrupt Service Routine.

All external interrupts (INT0, INT1, INT2 and INT3) can wake up the processor from the power-managed modes, if bit, INTxIE, was set prior to going into the power-managed modes. If the Global Interrupt Enable bit (GIE) is set, the processor will branch to the interrupt vector following wake-up.

The interrupt priority for INT1, INT2 and INT3 is determined by the value contained in the Interrupt Priority bits, INT1IP (INTCON3<6>), INT2IP (INTCON3<7>) and INT3IP (INTCON2<1>).

There is no priority bit associated with INT0; it is always a high-priority interrupt source.

10.7 TMR0 Interrupt

In 8-bit mode (the default), an overflow in the TMR0 register (FFh → 00h) will set flag bit, TMR0IF. In 16-bit mode, an overflow in the TMR0H:TMR0L register pair (FFFFh → 0000h) will set TMR0IF.

The interrupt can be enabled/disabled by setting/clearing enable bit, TMR0IE (INTCON<5>). Interrupt priority for Timer0 is determined by the value contained in the interrupt priority bit, TMR0IP (INTCON2<2>). For further details on the Timer0 module, see **Section 13.0 “Timer0 Module”**.

10.8 PORTB Interrupt-on-Change

An input change on PORTB<7:4> sets flag bit, RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit, RBIE (INTCON<3>), and each individual pin can be enabled/disabled by its corresponding bit in the IOCB register.

Interrupt priority for PORTB interrupt-on-change is determined by the value contained in the interrupt priority bit, RBIP (INTCON2<0>).

REGISTER 10-20: IOCB: INTERRUPT-ON-CHANGE PORTB CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
IOCB7 ⁽¹⁾	IOCB6 ⁽¹⁾	IOCB5 ⁽¹⁾	IOCB4 ⁽¹⁾	—	—	—	—
bit 7							bit 0

Legend:

R = Readable bit
-n = Value at POR

W = Writable bit
'1' = Bit is set

U = Unimplemented bit, read as '0'
'0' = Bit is cleared
x = Bit is unknown

bit 7-4 **IOCB<7:4>**: Interrupt-on-Change PORTB Control bits⁽¹⁾

1 = Interrupt-on-change is enabled
0 = Interrupt-on-change is disabled

bit 3-0 **Unimplemented**: Read as '0'

Note 1: Interrupt-on-change also requires that the RBIE bit of the INTCON register be set.

For more details on selecting the optimum C1 and C2 for a given crystal, see the crystal manufacture's applications information. The optimum value depends in part on the amount of parasitic capacitance in the circuit, which is often unknown. For that reason, it is highly recommended that thorough testing and validation of the oscillator be performed after values have been selected.

14.5.1 USING SOSC AS A CLOCK SOURCE

The SOSC oscillator is also available as a clock source in power-managed modes. By setting the clock select bits, $SCS<1:0>$ ($OSCCON<1:0>$), to '01', the device switches to SEC_RUN mode and both the CPU and peripherals are clocked from the SOSC oscillator. If the IDLEN bit ($OSCCON<7>$) is cleared and a SLEEP instruction is executed, the device enters SEC_IDLE mode. Additional details are available in **Section 4.0 "Power-Managed Modes"**.

Whenever the SOSC oscillator is providing the clock source, the SOSC System Clock Status flag, SOSCRUN ($OSCCON<2:6>$), is set. This can be used to determine the controller's current clocking mode. It can also indicate the clock source currently being used by the Fail-Safe Clock Monitor.

If the Clock Monitor is enabled and the SOSC oscillator fails while providing the clock, polling the SOSCRUN bit will indicate whether the clock is being provided by the SOSC oscillator or another source.

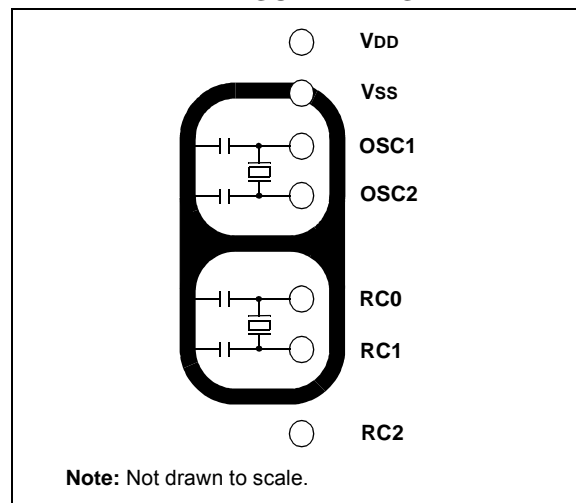
14.5.2 SOSC OSCILLATOR LAYOUT CONSIDERATIONS

The SOSC oscillator circuit draws very little power during operation. Due to the low-power nature of the oscillator, it may also be sensitive to rapidly changing signals in close proximity. This is especially true when the oscillator is configured for extremely Low-Power mode, $SOSCSEL<1:0>$ ($CONFIG1L<4:3>$) = 01.

The oscillator circuit, displayed in Figure 14-2, should be located as close as possible to the microcontroller. There should be no circuits passing within the oscillator circuit boundaries other than VSS or VDD.

If a high-speed circuit must be located near the oscillator, it may help to have a grounded guard ring around the oscillator circuit. The guard, as displayed in Figure 14-3, could be used on a single-sided PCB or in addition to a ground plane. (Examples of a high-speed circuit include the ECCP1 pin, in Output Compare or PWM mode, or the primary oscillator, using the OSC2 pin.)

FIGURE 14-3: OSCILLATOR CIRCUIT WITH GROUNDED GUARD RING



In the Low Drive Level mode, $SOSCSEL<1:0>$ = 01, it is critical that RC2 I/O pin signals be kept away from the oscillator circuit. Configuring RC2 as a digital output, and toggling it, can potentially disturb the oscillator circuit, even with a relatively good PCB layout. If possible, either leave RC2 unused or use it as an input pin with a slew rate limited signal source. If RC2 must be used as a digital output, it may be necessary to use the Higher Drive Level Oscillator mode ($SOSCSEL<1:0>$ = 11) with many PCB layouts.

Even in the Higher Drive Level mode, careful layout procedures should still be followed when designing the oscillator circuit.

In addition to dV/dt induced noise considerations, it is important to ensure that the circuit board is clean. Even a very small amount of conductive, soldering flux residue can cause PCB leakage currents that can overwhelm the oscillator circuit.

14.6 Timer1 Interrupt

The TMR1 register pair (TMR1H:TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The Timer1 interrupt, if enabled, is generated on overflow which is latched in interrupt flag bit, TMR1IF ($PIR1<0>$). This interrupt can be enabled or disabled by setting or clearing the Timer1 Interrupt Enable bit, TMR1IE ($PIE1<0>$).

20.4.2 FULL-BRIDGE MODE

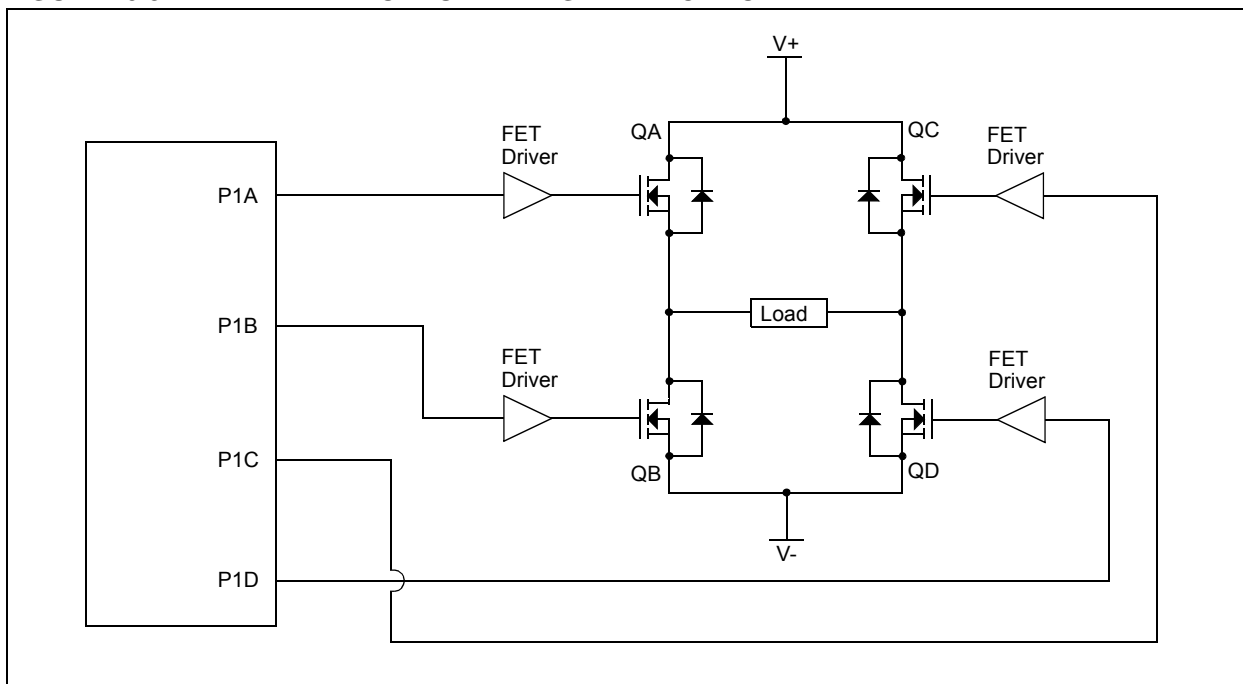
In Full-Bridge mode, all four pins are used as outputs. An example of a full-bridge application is provided in Figure 20-8.

In the Forward mode, the P1A pin is driven to its active state and the P1D pin is modulated, while the P1B and P1C pins are driven to their inactive state, as provided in Figure 20-9.

In the Reverse mode, the P1C pin is driven to its active state and the P1B pin is modulated, while the P1A and P1D pins are driven to their inactive state, as provided in Figure 20-9.

The P1A, P1B, P1C and P1D outputs are multiplexed with the port data latches. The associated TRIS bits must be cleared to configure the P1A, P1B, P1C and P1D pins as outputs.

FIGURE 20-8: EXAMPLE OF FULL-BRIDGE APPLICATION



PIC18F66K80 FAMILY

REGISTER 21-5: SSPCON2: MSSP CONTROL REGISTER 2 (I²C™ MASTER MODE)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GCEN	ACKSTAT	ACKDT ⁽¹⁾	ACKEN ⁽²⁾	RCEN ⁽²⁾	PEN ⁽²⁾	RSEN ⁽²⁾	SEN ⁽²⁾
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 **GCEN:** General Call Enable bit
Unused in Master mode.
- bit 6 **ACKSTAT:** Acknowledge Status bit (Master Transmit mode only)
1 = Acknowledge was not received from slave
0 = Acknowledge was received from slave
- bit 5 **ACKDT:** Acknowledge Data bit (Master Receive mode only)⁽¹⁾
1 = Not Acknowledged
0 = Acknowledged
- bit 4 **ACKEN:** Acknowledge Sequence Enable bit⁽²⁾
1 = Initiates Acknowledge sequence on SDA and SCL pins and transmits ACKDT data bit; automatically cleared by hardware
0 = Acknowledge sequence is Idle
- bit 3 **RCEN:** Receive Enable bit (Master Receive mode only)⁽²⁾
1 = Enables Receive mode for I²C™
0 = Receive is Idle
- bit 2 **PEN:** Stop Condition Enable bit⁽²⁾
1 = Initiates Stop condition on SDA and SCL pins; automatically cleared by hardware
0 = Stop condition is Idle
- bit 1 **RSEN:** Repeated Start Condition Enable bit⁽²⁾
1 = Initiates Repeated Start condition on SDA and SCL pins; automatically cleared by hardware
0 = Repeated Start condition Idle
- bit 0 **SEN:** Start Condition Enable bit⁽²⁾
1 = Initiates Start condition on SDA and SCL pins; automatically cleared by hardware
0 = Start condition Idle

Note 1: The value that will be transmitted when the user initiates an Acknowledge sequence at the end of a receive.

2: If the I²C module is active, these bits may not be set (no spooling) and the SSPBUF may not be written to (or writes to the SSPBUF are disabled).

21.4.4 CLOCK STRETCHING

Both 7-Bit and 10-Bit Slave modes implement automatic clock stretching during a transmit sequence.

The SEN bit (SSPCON2<0>) allows clock stretching to be enabled during receives. Setting SEN will cause the SCL pin to be held low at the end of each data receive sequence.

21.4.4.1 Clock Stretching for 7-Bit Slave Receive Mode (SEN = 1)

In 7-Bit Slave Receive mode, on the falling edge of the ninth clock at the end of the ACK sequence, if the BF bit is set, the CKP bit in the SSPCON1 register is automatically cleared, forcing the SCL output to be held low. The CKP bit being cleared to '0' will assert the SCL line low. The CKP bit must be set in the user's ISR before reception is allowed to continue. By holding the SCL line low, the user has time to service the ISR and read the contents of the SSPBUF before the master device can initiate another receive sequence. This will prevent buffer overruns from occurring (see Figure 21-15).

Note 1: If the user reads the contents of the SSPBUF before the falling edge of the ninth clock, thus clearing the BF bit, the CKP bit will not be cleared and clock stretching will not occur.

2: The CKP bit can be set in software, regardless of the state of the BF bit. The user should be careful to clear the BF bit in the ISR before the next receive sequence in order to prevent an overflow condition.

21.4.4.2 Clock Stretching for 10-Bit Slave Receive Mode (SEN = 1)

In 10-Bit Slave Receive mode, during the address sequence, clock stretching automatically takes place but CKP is not cleared. During this time, if the UA bit is set after the ninth clock, clock stretching is initiated. The UA bit is set after receiving the upper byte of the 10-bit address and following the receive of the second byte of the 10-bit address with the R/W bit cleared to '0'. The release of the clock line occurs upon updating SSPADD. Clock stretching will occur on each data receive sequence as described in 7-bit mode.

Note: If the user polls the UA bit and clears it by updating the SSPADD register before the falling edge of the ninth clock occurs, and if the user hasn't cleared the BF bit by reading the SSPBUF register before that time, then the CKP bit will still NOT be asserted low. Clock stretching on the basis of the state of the BF bit only occurs during a data sequence, not an address sequence.

21.4.4.3 Clock Stretching for 7-Bit Slave Transmit Mode

The 7-Bit Slave Transmit mode implements clock stretching by clearing the CKP bit after the falling edge of the ninth clock if the BF bit is clear. This occurs regardless of the state of the SEN bit.

The user's ISR must set the CKP bit before transmission is allowed to continue. By holding the SCL line low, the user has time to service the ISR and load the contents of the SSPBUF before the master device can initiate another transmit sequence (see Figure 21-10).

Note 1: If the user loads the contents of SSPBUF, setting the BF bit before the falling edge of the ninth clock, the CKP bit will not be cleared and clock stretching will not occur.

2: The CKP bit can be set in software regardless of the state of the BF bit.

21.4.4.4 Clock Stretching for 10-Bit Slave Transmit Mode

In 10-Bit Slave Transmit mode, clock stretching is controlled during the first two address sequences by the state of the UA bit, just as it is in 10-Bit Slave Receive mode. The first two addresses are followed by a third address sequence, which contains the high-order bits of the 10-bit address and the R/W bit set to '1'. After the third address sequence is performed, the UA bit is not set, the module is now configured in Transmit mode and clock stretching is controlled by the BF flag as in 7-Bit Slave Transmit mode (see Figure 21-13).

22.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART)

The Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) module is one of two serial I/O modules. (Generically, the EUSART is also known as a Serial Communications Interface or SCI.)

The EUSART can be configured as a full-duplex, asynchronous system that can communicate with peripheral devices, such as CRT terminals and personal computers. It can also be configured as a half-duplex synchronous system that can communicate with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs, etc.

The Enhanced USARTx modules implement additional features, including automatic baud rate detection and calibration, automatic wake-up on Sync Break reception and 12-bit Break character transmit. These make it ideally suited for use in Local Interconnect Network bus (LIN/J2602 bus) systems.

All members of the PIC18F66K80 family are equipped with two independent EUSART modules, referred to as EUSART1 and EUSART2. They can be configured in the following modes:

- Asynchronous (full duplex) with:
 - Auto-wake-up on character reception
 - Auto-baud calibration
 - 12-bit Break character transmission
- Synchronous – Master (half duplex) with selectable clock polarity
- Synchronous – Slave (half duplex) with selectable clock polarity

The pins of EUSART1 and EUSART2 are multiplexed with the functions with the following ports, depending on the device pin count. See Table 22-1.

TABLE 22-1: CONFIGURING EUSARTx PINS⁽¹⁾

Pin Count	EUSART1		EUSART2	
	Port	Pins	Port	Pins
28-pin	PORTC	RC6/TX1/CK1 and RC7/RX1/DT1	PORTB	RB6/PGC/TX2/CK2/KBI2 and RB7/PGD/T3G/RX2/DT2/KBI3
40/44-pin	PORTC	RC6/TX1/CK1 and RC7/RX1/DT1	PORTD	RD6/TX2/CK2/P1C/PSP6 and RD7/RX2/DT2/P1D/PSP7
64-pin	PORTG	RG3/TX1/CK1 and RG0/RX1/DT1	PORTE	RE7/TX2/CK2 and RE6/RX2/DT2

Note 1: The EUSARTx control will automatically reconfigure the pin from input to output as needed.

In order to configure the pins as an EUSARTx:

- For EUSART1:
 - SPEN (RCSTA1<7>) must be set (= 1)
 - TRISx<x> must be set (= 1)
 - For Asynchronous and Synchronous Master modes, TRISx<x> must be cleared (= 0)
 - For Synchronous Slave mode, TRISx<x> must be set (= 1)
- For EUSART2:
 - SPEN (RCSTA2<7>) must be set (= 1)
 - TRISx<x> must be set (= 1)
 - For Asynchronous and Synchronous Master modes, TRISx<x> must be cleared (= 0)
 - For Synchronous Slave mode, TRISx<x> must be set (= 1)

25.0 COMPARATOR VOLTAGE REFERENCE MODULE

The comparator voltage reference is a 32-tap resistor ladder network that provides a selectable reference voltage. Although its primary purpose is to provide a reference for the analog comparators, it may also be used independently of them.

A block diagram of the module is shown in Figure 25-1. The resistor ladder is segmented to provide a range of CVREF values and has a power-down function to conserve power when the reference is not being used. The module's supply reference can be provided from either device VDD/VSS or an external voltage reference.

25.1 Configuring the Comparator Voltage Reference

The comparator voltage reference module is controlled through the CVRCON register (Register 25-1). The comparator voltage reference provides a range of output voltage with 32 levels.

The CVR<4:0> selection bits (CVRCON<4:0>) offer a range of output voltages. Equation 25-1 shows the how the comparator voltage reference is computed.

EQUATION 25-1:

If CVRSS = 1:

$$CVREF = \left(V_{REF-} + \frac{CVR<4:0>}{32} \right) \cdot (V_{REF+} - V_{REF-})$$

If CVRSS = 0:

$$CVREF = \left(AV_{SS} + \frac{CVR<4:0>}{32} \right) \cdot (AV_{DD} - AV_{SS})$$

The comparator reference supply voltage can come from either VDD and VSS, or the external VREF+ and VREF- that are multiplexed with RA3 and RA2. The voltage source is selected by the CVRSS bit (CVRCON<5>).

The settling time of the comparator voltage reference must be considered when changing the CVREF output (see Table 31-2 in **Section 31.0 "Electrical Characteristics"**).

REGISTER 25-1: CVRCON: COMPARATOR VOLTAGE REFERENCE CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0
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	CVREN: Comparator Voltage Reference Enable bit 1 = CVREF circuit powered on 0 = CVREF circuit powered down
bit 6	CVROE: Comparator VREF Output Enable bit 1 = CVREF voltage level is output on CVREF pin 0 = CVREF voltage level is disconnected from CVREF pin
bit 5	CVRSS: Comparator VREF Source Selection bit 1 = Comparator reference source, CVRSRC = VREF+ – VREF- 0 = Comparator reference source, CVRSRC = AVDD – AVSS
bit 4-0	CVR<4:0>: Comparator VREF Value Selection $0 \leq CVR<4:0> \leq 31$ bits When CVRSS = 1: $CVREF = (V_{REF-}) + (CVR<4:0>/32) \cdot (V_{REF+} - V_{REF-})$ When CVRSS = 0: $CVREF = (AV_{SS}) + (CVR<4:0>/32) \cdot (AV_{DD} - AV_{SS})$

PIC18F66K80 FAMILY

REGISTER 27-4: COMSTAT: COMMUNICATION STATUS REGISTER

Mode 0	R/C-0	R/C-0	R-0	R-0	R-0	R-0	R-0	R-0
	RXB0OVFL	RXB1OVFL	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN
Mode 1	R/C-0	R/C-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	RXBnOVFL	TXB0	TXBP	RXBP	TXWARN	RXWARN	EWARN
Mode 2	R/C-0	R/C-0	R-0	R-0	R-0	R-0	R-0	R-0
	FIFOEMPTY	RXBnOVFL	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN
bit 7								
bit 0								

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

- bit 7 Mode 0:
RXB0OVFL: Receive Buffer 0 Overflow bit
 1 = Receive Buffer 0 has overflowed
 0 = Receive Buffer 0 has not overflowed
Mode 1:
Unimplemented: Read as '0'
Mode 2:
FIFOEMPTY: FIFO Not Empty bit
 1 = Receive FIFO is not empty
 0 = Receive FIFO is empty
- bit 6 Mode 0:
RXB1OVFL: Receive Buffer 1 Overflow bit
 1 = Receive Buffer 1 has overflowed
 0 = Receive Buffer 1 has not overflowed
Mode 1, 2:
RXBnOVFL: Receive Buffer n Overflow bit
 1 = Receive Buffer n has overflowed
 0 = Receive Buffer n has not overflowed
- bit 5 **TXBO:** Transmitter Bus-Off bit
 1 = Transmit error counter > 255
 0 = Transmit error counter ≤ 255
- bit 4 **TXBP:** Transmitter Bus Passive bit
 1 = Transmit error counter > 127
 0 = Transmit error counter ≤ 127
- bit 3 **RXBP:** Receiver Bus Passive bit
 1 = Receive error counter > 127
 0 = Receive error counter ≤ 127
- bit 2 **TXWARN:** Transmitter Warning bit
 1 = Transmit error counter > 95
 0 = Transmit error counter ≤ 95
- bit 1 **RXWARN:** Receiver Warning bit
 1 = 127 ≥ Receive error counter > 95
 0 = Receive error counter ≤ 95
- bit 0 **EWARN:** Error Warning bit
 This bit is a flag of the RXWARN and TXWARN bits.
 1 = The RXWARN or the TXWARN bits are set
 0 = Neither the RXWARN or the TXWARN bits are set

PIC18F66K80 FAMILY

REGISTER 27-45: RXFCONn: RECEIVE FILTER CONTROL REGISTER 'n' [0 ≤ n ≤ 1]⁽¹⁾

RXFCON0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXF7EN	RXF6EN	RXF5EN	RXF4EN	RXF3EN	RXF2EN	RXF1EN	RXF0EN

RXFCON1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	RXF15EN	RXF14EN	RXF13EN	RXF12EN	RXF11EN	RXF10EN	RXF9EN	RXF8EN
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 7-0 **RXF<7:0>EN:** Receive Filter n Enable bits
 0 = Filter is disabled
 1 = Filter is enabled

Note 1: This register is available in Mode 1 and 2 only.

Note: Register 27-46 through Register 27-51 are writable in Configuration mode only.

REGISTER 27-46: SDFLC: STANDARD DATA BYTES FILTER LENGTH COUNT REGISTER⁽¹⁾

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	FLC4	FLC3	FLC2	FLC1	FLC0
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **FLC<4:0>:** Filter Length Count bits

Mode 0:

Not used; forced to '00000'.

00000-10010 = 0

18 bits are available for standard data byte filter. Actual number of bits used depends on the DLC<3:0> bits (RXBnDLC<3:0> or BnDLC<3:0> if configured as RX buffer) of the message being received.

If DLC<3:0> = 0000

No bits will be compared with incoming data bits.

If DLC<3:0> = 0001

Up to 8 data bits of RXFnEID<7:0>, as determined by FLC<2:0>, will be compared with the corresponding number of data bits of the incoming message.

If DLC<3:0> = 0010

Up to 16 data bits of RXFnEID<15:0>, as determined by FLC<3:0>, will be compared with the corresponding number of data bits of the incoming message.

If DLC<3:0> = 0011

Up to 18 data bits of RXFnEID<17:0>, as determined by FLC<4:0>, will be compared with the corresponding number of data bits of the incoming message.

Note 1: This register is available in Mode 1 and 2 only.

PIC18F66K80 FAMILY

REGISTER 28-11: CONFIG7L: CONFIGURATION REGISTER 7 LOW (BYTE ADDRESS 30000Ch)

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	EBTR3	EBTR2	EBTR1	EBTR0
bit 7				bit 0			

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

- bit 7-4 **Unimplemented:** Read as '0'
- bit 3 **EBTR3:** Table Read Protection bit
1 = Block 3 is not protected from table reads executed in other blocks⁽¹⁾
0 = Block 3 is protected from table reads executed in other blocks⁽¹⁾
- bit 2 **EBTR2:** Table Read Protection bit
1 = Block 2 is not protected from table reads executed in other blocks⁽¹⁾
0 = Block 2 is protected from table reads executed in other blocks⁽¹⁾
- bit 1 **EBTR1:** Table Read Protection bit
1 = Block 1 is not protected from table reads executed in other blocks⁽¹⁾
0 = Block 1 is protected from table reads executed in other blocks⁽¹⁾
- bit 0 **EBTR0:** Table Read Protection bit
1 = Block 0 is not protected from table reads executed in other blocks⁽¹⁾
0 = Block 0 is protected from table reads executed in other blocks⁽¹⁾

Note 1: For the memory size of the blocks, see Figure 28-6.

PIC18F66K80 FAMILY

TABLE 29-2: PIC18F66K80 FAMILY INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
				MSb		LSb			
BIT-ORIENTED OPERATIONS									
BCF	f, b, a	Bit Clear f	1	1001	bbba	ffff	ffff	None	1, 2
BSF	f, b, a	Bit Set f	1	1000	bbba	ffff	ffff	None	1, 2
BTFSC	f, b, a	Bit Test f, Skip if Clear	1 (2 or 3)	1011	bbba	ffff	ffff	None	3, 4
BTFSS	f, b, a	Bit Test f, Skip if Set	1 (2 or 3)	1010	bbba	ffff	ffff	None	3, 4
BTG	f, b, a	Bit Toggle f	1	0111	bbba	ffff	ffff	None	1, 2
CONTROL OPERATIONS									
BC	n	Branch if Carry	1 (2)	1110	0010	nnnn	nnnn	None	4
BN	n	Branch if Negative	1 (2)	1110	0110	nnnn	nnnn	None	
BNC	n	Branch if Not Carry	1 (2)	1110	0011	nnnn	nnnn	None	
BNN	n	Branch if Not Negative	1 (2)	1110	0111	nnnn	nnnn	None	
BNOV	n	Branch if Not Overflow	1 (2)	1110	0101	nnnn	nnnn	None	
BNZ	n	Branch if Not Zero	1 (2)	1110	0001	nnnn	nnnn	None	
BOV	n	Branch if Overflow	1 (2)	1110	0100	nnnn	nnnn	None	
BRA	n	Branch Unconditionally	2	1101	0nnn	nnnn	nnnn	None	
BZ	n	Branch if Zero	1 (2)	1110	0000	nnnn	nnnn	None	
CALL	n, s	Call Subroutine 1st word 2nd word	2	1110	110s	kkkk	kkkk	None	
				1111	kkkk	kkkk	kkkk		
CLRWDT	—	Clear Watchdog Timer	1	0000	0000	0000	0100	$\overline{\text{TO}}$, $\overline{\text{PD}}$	
DAW	—	Decimal Adjust WREG	1	0000	0000	0000	0111	C	
GOTO	n	Go to Address 1st word 2nd word	2	1110	1111	kkkk	kkkk	None	
				1111	kkkk	kkkk	kkkk		
NOP	—	No Operation	1	0000	0000	0000	0000	None	
NOP	—	No Operation	1	1111	xxxx	xxxx	xxxx	None	
POP	—	Pop Top of Return Stack (TOS)	1	0000	0000	0000	0110	None	
PUSH	—	Push Top of Return Stack (TOS)	1	0000	0000	0000	0101	None	
RCALL	n	Relative Call	2	1101	1nnn	nnnn	nnnn	None	
RESET		Software Device Reset	1	0000	0000	1111	1111	All	
RETFIE	s	Return from Interrupt Enable	2	0000	0000	0001	000s	GIE/GIEH, PEIE/GIEL	
RETLW	k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None	
RETURN	s	Return from Subroutine	2	0000	0000	0001	001s	None	
SLEEP	—	Go into Standby mode	1	0000	0000	0000	0011	$\overline{\text{TO}}$, $\overline{\text{PD}}$	

Note 1: When a PORT register is modified as a function of itself (e.g., `MOVF PORTB, 1, 0`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as an input and is driven low by an external device, the data will be written back with a '0'.

- 2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned.
- 3: If the Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.
- 4: Some instructions are two-word instructions. The second word of these instructions will be executed as a `NOP` unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

PIC18F66K80 FAMILY

LFSR Load FSR

Syntax: LFSR f, k

Operands: $0 \leq f \leq 2$
 $0 \leq k \leq 4095$

Operation: $k \rightarrow \text{FSRf}$

Status Affected: None

Encoding:

1110	1110	00ff	k ₁₁ kkk
1111	0000	k ₇ kkk	kkkk

Description: The 12-bit literal 'k' is loaded into the file select register pointed to by 'f'.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k' MSB	Process Data	Write literal 'k' MSB to FSRfH
Decode	Read literal 'k' LSB	Process Data	Write literal 'k' to FSRfL

Example: LFSR 2, 3ABh

After Instruction

FSR2H = 03h
 FSR2L = ABh

MOVF Move f

Syntax: MOVF f {,d {,a}}

Operands: $0 \leq f \leq 255$
 $d \in [0,1]$
 $a \in [0,1]$

Operation: $f \rightarrow \text{dest}$

Status Affected: N, Z

Encoding:

0101	00da	ffff	ffff
------	------	------	------

Description: The contents of register 'f' are moved to a destination dependent upon the status of 'd'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f' (default). Location 'f' can be anywhere in the 256-byte bank.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \leq 95$ (5Fh). See **Section 29.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write W

Example: MOVF REG, 0, 0

Before Instruction

REG = 22h
 W = FFh

After Instruction

REG = 22h
 W = 22h

29.2 Extended Instruction Set

In addition to the standard 75 instructions of the PIC18 instruction set, the PIC18F66K80 family of devices also provides an optional extension to the core CPU functionality. The added features include eight additional instructions that augment Indirect and Indexed Addressing operations and the implementation of Indexed Literal Offset Addressing for many of the standard PIC18 instructions.

The additional features of the extended instruction set are enabled by default on unprogrammed devices. Users must properly set or clear the XINST Configuration bit during programming to enable or disable these features.

The instructions in the extended set can all be classified as literal operations, which either manipulate the File Select Registers, or use them for Indexed Addressing. Two of the instructions, ADDFSR and SUBFSR, each have an additional special instantiation for using FSR2. These versions (ADDULNK and SUBULNK) allow for automatic return after execution.

The extended instructions are specifically implemented to optimize re-entrant program code (that is, code that is recursive or that uses a software stack) written in high-level languages, particularly C. Among other things, they allow users working in high-level languages to perform certain operations on data structures more efficiently. These include:

- Dynamic allocation and deallocation of software stack space when entering and leaving subroutines
- Function Pointer invocation
- Software Stack Pointer manipulation
- Manipulation of variables located in a software stack

A summary of the instructions in the extended instruction set is provided in Table 29-3. Detailed descriptions are provided in **Section 29.2.2 “Extended Instruction Set”**. The opcode field descriptions in Table 29-1 (page 484) apply to both the standard and extended PIC18 instruction sets.

Note: The instruction set extension and the Indexed Literal Offset Addressing mode were designed for optimizing applications written in C; the user may likely never use these instructions directly in assembler. The syntax for these commands is provided as a reference for users who may be reviewing code that has been generated by a compiler.

29.2.1 EXTENDED INSTRUCTION SYNTAX

Most of the extended instructions use indexed arguments, using one of the File Select Registers and some offset to specify a source or destination register. When an argument for an instruction serves as part of Indexed Addressing, it is enclosed in square brackets (“[]”). This is done to indicate that the argument is used as an index or offset. The MPASM™ Assembler will flag an error if it determines that an index or offset value is not bracketed.

When the extended instruction set is enabled, brackets are also used to indicate index arguments in byte-oriented and bit-oriented instructions. This is in addition to other changes in their syntax. For more details, see **Section 29.2.3.1 “Extended Instruction Syntax with Standard PIC18 Commands”**.

Note: In the past, square brackets have been used to denote optional arguments in the PIC18 and earlier instruction sets. In this text and going forward, optional arguments are denoted by braces (“{}”).

TABLE 29-3: EXTENSIONS TO THE PIC18 INSTRUCTION SET

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected
			MSb		LSb		
ADDFSR f, k	Add Literal to FSR	1	1110	1000	ffkk	kkkk	None
ADDULNK k	Add Literal to FSR2 and Return	2	1110	1000	11kk	kkkk	None
CALLW	Call Subroutine using WREG	2	0000	0000	0001	0100	None
MOVSF z _s , f _d	Move z _s (source) to 1st word f _d (destination) 2nd word	2	1110	1011	0zzz	zzzz	None
MOVSS z _s , z _d	Move z _s (source) to 1st word z _d (destination) 2nd word	2	1110	1011	1zzz	zzzz	None
PUSHL k	Store Literal at FSR2, Decrement FSR2	1	1110	1010	kkkk	kkkk	None
SUBFSR f, k	Subtract Literal from FSR	1	1110	1001	ffkk	kkkk	None
SUBULNK k	Subtract Literal from FSR2 and return	2	1110	1001	11kk	kkkk	None

PIC18F66K80 FAMILY

MOVSS Move Indexed to Indexed

Syntax: MOVSS [z_s], [z_d]

Operands: 0 ≤ z_s ≤ 127
0 ≤ z_d ≤ 127

Operation: ((FSR2) + z_s) → ((FSR2) + z_d)

Status Affected: None

Encoding:

1110	1011	1zzz	zzzz _s
1111	xxxx	xzzz	zzzz _d

1st word (source)

2nd word (dest.)

Description

The contents of the source register are moved to the destination register. The addresses of the source and destination registers are determined by adding the 7-bit literal offsets, 'z_s' or 'z_d', respectively, to the value of FSR2. Both registers can be located anywhere in the 4096-byte data memory space (000h to FFFh).

The MOVSS instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

If the resultant source address points to an Indirect Addressing register, the value returned will be 00h. If the resultant destination address points to an Indirect Addressing register, the instruction will execute as a NOP.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	Determine dest addr	Determine dest addr	Write to dest reg

Example: MOVSS [05h], [06h]

Before Instruction

FSR2 = 80h

Contents of 85h = 33h

Contents of 86h = 11h

After Instruction

FSR2 = 80h

Contents of 85h = 33h

Contents of 86h = 33h

PUSHL Store Literal at FSR2, Decrement FSR2

Syntax: PUSHL k

Operands: 0 ≤ k ≤ 255

Operation: k → (FSR2),
FSR2 – 1 → FSR2

Status Affected: None

Encoding:

1111	1010	kkkk	kkkk
------	------	------	------

Description:

The 8-bit literal 'k' is written to the data memory address specified by FSR2. FSR2 is decremented by 1 after the operation.

This instruction allows users to push values onto a software stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read 'k'	Process data	Write to destination

Example: PUSHL 08h

Before Instruction

FSR2H:FSR2L = 01ECh

Memory (01ECh) = 00h

After Instruction

FSR2H:FSR2L = 01EBh

Memory (01ECh) = 08h

PIC18F66K80 FAMILY

31.2 DC Characteristics: Power-Down and Supply Current PIC18F66K80 Family (Industrial/Extended) (Continued)

PIC18F66K80 Family (Industrial/Extended)		Standard Operating Conditions (unless otherwise stated)					
		Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	Device	Typ	Max	Units	Conditions		
	Supply Current (IDD) Cont. ^(2,3)						
	PIC18LFXXK80	20	70	μA	-40°C	VDD = 1.8V ⁽⁴⁾ Regulator disabled	FOSC = 1 MHz (PRI_IDLE mode, EC oscillator)
		20	70	μA	+25°C		
		20	70	μA	+60°C		
		25	80	μA	+85°C		
		30	100	μA	+125°C		
	PIC18LFXXK80	37	120	μA	-40°C	VDD = 3.3V ⁽⁴⁾ Regulator disabled	
		37	120	μA	+25°C		
		37	120	μA	+60°C		
		40	130	μA	+85°C		
		45	150	μA	+125°C		
	PIC18FXXK80	85	140	μA	-40°C	VDD = 3.3V ⁽⁵⁾ Regulator enabled	
		100	140	μA	+25°C		
		105	140	μA	+60°C		
		110	150	μA	+85°C		
		120	170	μA	+125°C		
	PIC18FXXK80	110	225	μA	-40°C	VDD = 5V ⁽⁵⁾ Regulator enabled	
		110	225	μA	+25°C		
		110	225	μA	+60°C		
		120	230	μA	+85°C		
130		250	μA	+125°C			

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;

MCLR = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: For LF devices, RETEN (CONFIG1L<0>) = 1.

5: For F devices, SRETEN (WDTCON<4>) = 1 and RETEN (CONFIG1L<0>) = 0.

PIC18F66K80 FAMILY

31.3 DC Characteristics: PIC18F66K80 Family (Industrial) (Continued)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended			
Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
D090	VOH	Output High Voltage⁽¹⁾ I/O Ports: PORTA, PORTB, PORTC	$V_{DD} - 0.7$	—	V	IOH = -3 mA, VDD = 5.5V, -40°C to +125°C IOH = -2 mA, VDD = 5.5V, -40°C to +125°C IOH = -1 mA, VDD = 5.5V, -40°C to +125°C
		PORTD, PORTE, PORTF, PORTG	$V_{DD} - 0.7$	—	V	
D092		OSC2/CLKO (INTOSC, EC modes)	$V_{DD} - 0.7$	—	V	
D100 ⁽⁴⁾	COSC2	Capacitive Loading Specs on Output Pins OSC2 Pin	—	20	pF	In HS mode when external clock is used to drive OSC1 To meet the AC Timing Specifications I ² C™ Specification
D101	CIO	All I/O Pins and OSC2	—	50	pF	
D102	CB	SCL, SDA	—	400	pF	

Note 1: Negative current is defined as current sourced by the pin.

PIC18F66K80 FAMILY

TABLE 31-1: MEMORY PROGRAMMING REQUIREMENTS

DC CHARACTERISTICS			Standard Operating Conditions Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for Industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for Extended				
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
Internal Program Memory Programming Specifications⁽¹⁾							
D110	VPP	Voltage on $\overline{\text{MCLR}}$ /VPP/RE5 pin	VDD + 1.5	—	10	V	(Note 3, Note 4)
D113	IDDP	Supply Current during Programming	—	—	10	mA	
Data EEPROM Memory							
D120	ED	Byte Endurance	100K	1000K	—	E/W	(Note 2) -40°C to +125°C
D121	VDRW	VDD for Read/Write	1.8	—	5.5	V	
			1.8	—	3.6	V	Using EECON to read/write PIC18FXXKXX devices
D122	TDEW	Erase/Write Cycle Time	—	4	—	ms	Using EECON to read/write PIC18LFXXKXX devices
D123	TRETD	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated
D124	TREF	Number of Total Erase/Write Cycles before Refresh ⁽²⁾	1M	10M	—	E/W	-40°C to +125°C
Program Flash Memory							
D130	EP	Cell Endurance	1K	10K	—	E/W	-40°C to +125°C
D131	VPR	VDD for Read	1.8	—	5.5	V	
			1.8	—	3.6	V	PIC18FXXKXX devices
D132B	VPEW	Voltage for Self-Timed Erase or Write Operations VDD	1.8	—	5.5	V	PIC18LFXXKXX devices
D133A	TIW	Self-Timed Write Cycle Time	—	2	—	ms	PIC18FXXKXX devices
D134	TRETD	Characteristic Retention	20	—	—	Year	Provided no other specifications are violated
D135	IDDP	Supply Current during Programming	—	—	10	mA	
D140	TWE	Writes per Erase Cycle	—	—	1		For each physical address

† Data in “Typ” column is at 3.3V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: These specifications are for programming the on-chip program memory through the use of table write instructions.

2: Refer to **Section 8.8 “Using the Data EEPROM”** for a more detailed discussion on data EEPROM endurance.

3: Required only if Single-Supply Programming is disabled.

4: The MPLAB® ICD 2 does not support variable VPP output. Circuitry to limit the ICD2 VPP voltage must be placed between the ICD 2 and target system when programming or debugging with the ICD2.