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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	EBI/EMI, I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	69
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 24x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f86k22-i-pt

PIC18F87K22 FAMILY

64/80-Pin, High-Performance, 1-Mbit Enhanced Flash MCUs with 12-Bit A/D and nanoWatt XLP Technology

Low-Power Features:

- Power-Managed modes:
  - Run: CPU on, peripherals on
  - Idle: CPU off, peripherals on
  - Sleep: CPU off, peripherals off
- Two-Speed Oscillator Start-up
- Fail-Safe Clock Monitor
- Power-Saving Peripheral Module Disable (PMD)
- Ultra Low-Power Wake-up
- Fast Wake-up, 1 μs Typical
- Low-Power WDT, 300 nA Typical
- Ultra Low 50 nA Input Leakage
- Run mode Currents Down to 5.5 μA, Typical
- Idle mode Currents Down to 1.7 μA Typical
- Sleep mode Currents Down to Very Low 20 nA, Typical
- RTCC Current Downs to Very Low 700 nA, Typical

Special Microcontroller Features:

- Operating Voltage Range: 1.8V to 5.5V
- On-Chip 3.3V Regulator
- Operating Speed up to 64 MHz
- Up to 128 Kbytes On-Chip Flash Program Memory
- Data EEPROM of 1,024 Bytes
- 4K x 8 General Purpose Registers (SRAM)
- 10,000 Erase/Write Cycle Flash Program Memory, Minimum
- 1,000,000 Erase/write Cycle Data EEPROM Memory, Typical
- Flash Retention: 40 Years, Minimum
- Three Internal Oscillators: LF-INTRC (31 kHz), MF-INTOSC (500 kHz) and HF-INTOSC (16 MHz)
- Self-Programmable under Software Control
- Priority Levels for Interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
  - Programmable period from 4 ms to 4,194s (about 70 minutes)
- In-Circuit Serial Programming™ (ICSP™) via Two Pins
- In-Circuit Debug via Two Pins
- Programmable:
  - BOR
  - LVD

Device	Program Memory		Data Memory		I/O	12-Bit A/D (ch)	CCP/ ECCP (PWM)	MSSP			EUSART	Comparators	Timers 8/16-Bit	External Bus	CTMU	RTCC
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I^2C™						
PIC18F65K22	32K	16,383	2K	1K	53	16	5/3	2	Y	Y	2	3	4/4	N	Y	Y
PIC18F66K22	64K	32,768	4K	1K	53	16	7/3	2	Y	Y	2	3	6/5	N	Y	Y
PIC18F67K22	128K	65,536	4K	1K	53	16	7/3	2	Y	Y	2	3	6/5	N	Y	Y
PIC18F85K22	32K	16,383	2K	1K	69	24	5/3	2	Y	Y	2	3	4/4	Y	Y	Y
PIC18F86K22	64K	32,768	4K	1K	69	24	7/3	2	Y	Y	2	3	6/5	Y	Y	Y
PIC18F87K22	128K	65,536	4K	1K	69	24	7/3	2	Y	Y	2	3	6/5	Y	Y	Y

TABLE 1-3: PIC18F6XK22 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	QFN/TQFP			
				PORTE is a bidirectional I/O port.
RE0/$\overline{RD}$/P2D	2			
RE0		I/O	ST	Digital I/O.
$\overline{RD}$		I	TTL	Parallel Slave Port read strobe.
P2D		O	—	EECP2 PWM Output D.
RE1/$\overline{WR}$/P2C	1			
RE1		I/O	ST	Digital I/O.
$\overline{WR}$		I	TTL	Parallel Slave Port write strobe.
P2C		O	—	ECCP2 PWM Output C.
RE2/$\overline{CS}$/P2B/CCP10	64			
RE2		I/O	ST	Digital I/O.
$\overline{CS}$		I	TTL	Parallel Slave Port chip select.
P2B		O	—	ECCP2 PWM Output B.
CCP10(3)		I/O	S/T	Capture 10 input/Compare 10 output/PWM10 output.
RE3/P3C/CCP9/REFO	63			
RE3		I/O	ST	Digital I/O.
P3C		O	—	ECCP3 PWM Output C.
CCP9(3,4)		I/O	S/T	Capture 9 input/Compare 9 output/PWM9 output.
REFO		O	—	Reference clock out.
RE4/P3B/CCP8	62			
RE4		I/O	ST	Digital I/O.
P3B		O	—	ECCP3 PWM Output B.
CCP8(4)		I/O	S/T	Capture 8 input/Compare 8 output/PWM8 output.
RE5/P1C/CCP7	61			
RE5		I/O	ST	Digital I/O.
P1C		O	—	ECCP1 PWM Output C.
CCP7(4)		I/O	S/T	Capture 7 input/Compare 7 output/PWM7 output.
RE6/P1B/CCP6	60			
RE6		I/O	ST	Digital I/O.
P1B		O	—	ECCP1 PWM Output B.
CCP6(4		I/O	S/T	Capture 6 input/Compare 6 output/PWM6 output.
RE7/ECCP2/P2A	59			
RE7		I/O	ST	Digital I/O.
ECCP2(2)		I/O	ST	Capture 2 input/Compare 2 output/PWM2 output.
P2A		O	—	ECCP2 PWM Output A.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; Analog = Analog input; I = Input; O = Output; P = Power; OD = Open-Drain (no P diode to V_{DD}); I^2C = I^2C™/SMBus

Note 1: Default assignment for ECCP2 when the CCP2MX Configuration bit is set.

2: Alternate assignment for ECCP2 when the CCP2MX Configuration bit is cleared.

3: Not available on PIC18F65K22 and PIC18F85K22 devices.

4: The CC6, CCP7, CCP8 and CCP9 pin placement depends on the setting of the ECCPMX Configuration bit (CONFIG3H<1>).

TABLE 5-2: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices		Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets, WDT Reset, RESET Instruction, Stack Resets, CM Resets	Wake-up via WDT or Interrupt
RTCCFG	PIC18F6XK22	PIC18F8XK22	0-00 0000	u-uu uuuu	u-uu uuuu
RTCCAL	PIC18F6XK22	PIC18F8XK22	0000 0000	uuuu uuuu	uuuu uuuu
RTCVALH	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
RTCVALL	PIC18F6XK22	PIC18F8XK22	0000 0000	uuuu uuuu	uuuu uuuu
ALRMCFG	PIC18F6XK22	PIC18F8XK22	0000 0000	uuuu uuuu	uuuu uuuu
ALRMRPT	PIC18F6XK22	PIC18F8XK22	0000 0000	uuuu uuuu	uuuu uuuu
ALRMVALH	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
ALRMVALL	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CTMUCONH	PIC18F6XK22	PIC18F8XK22	0-00 0000	0-00 0000	u-uu uuuu
CTMUCONL	PIC18F6XK22	PIC18F8XK22	0000 00xx	0000 00xx	uuuu uuuu
CTMUICONH	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
CM1CON	PIC18F6XK22	PIC18F8XK22	0001 1111	0001 1111	uuuu uuuu
PADCFG1	PIC18F6XK22	PIC18F8XK22	00-- -00-	uu-- -uu-	uu-- -uu-
PADCFG1	PIC18F6XK22	PIC18F8XK22	000- -00-	uuu- -uu-	uuu- -uu-
ECCP2AS	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
ECCP2DEL	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
CCPR2H	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR2L	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP2CON	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
ECCP3AS	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
ECCP3DEL	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
CCPR3H	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR3L	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP3CON	PIC18F6XK22	PIC18F8XK22	0000 0000	0000 0000	uuuu uuuu
CCPR8H	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR8L	PIC18F6XK22	PIC18F8XK22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP8CON	PIC18F6XK22	PIC18F8XK22	--00 0000	--00 0000	--uu uuuu
CCPR9H	PIC18F66K22 PIC18F67K22	PIC18F86K22 PIC18F87K22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR9L	PIC18F66K22 PIC18F67K22	PIC18F86K22 PIC18F87K22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP9CON	PIC18F66K22 PIC18F67K22	PIC18F86K22 PIC18F87K22	--00 0000	--00 0000	--uu uuuu
CCPR10H	PIC18F66K22 PIC18F67K22	PIC18F86K22 PIC18F87K22	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR10L	PIC18F66K22 PIC18F67K22	PIC18F86K22 PIC18F87K22	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: 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.

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: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

4: See Table 5-1 for Reset value for specific condition.

TABLE 6-2: PIC18F87K22 FAMILY REGISTER FILE SUMMARY (CONTINUED)

Address	File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
FE6h	POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented (not a physical register)								---- ----
FE5h	POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented (not a physical register)								---- ----
FE4h	PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register)								---- ----
FE3h	PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register) – value of FSR1 offset by W								---- ----
FE2h	FSR1H	—	—	—	—	Indirect Data Memory Address Pointer 1 High				---- xxxx
FE1h	FSR1L	Indirect Data Memory Address Pointer 1 Low Byte								xxxx xxxx
FE0h	BSR	—	—	—	—	Bank Select Register				---- 0000
FDFh	INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed (not a physical register)								---- ----
FDEh	POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented (not a physical register)								---- ----
FDDh	POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented (not a physical register)								---- ----
FDCh	PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register)								---- ----
FDBh	PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register) – value of FSR2 offset by W								---- ----
FDAh	FSR2H	—	—	—	—	Indirect Data Memory Address Pointer 2 High				---- xxxx
FD9h	FSR2L	Indirect Data Memory Address Pointer 2 Low Byte								xxxx xxxx
FD8h	STATUS	—	—	—	N	OV	Z	DC	C	---x xxxx
FD7h	TMR0H	Timer0 Register High Byte								0000 0000
FD6h	TMR0L	Timer0 Register Low Byte								xxxx xxxx
FD5h	T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	TOPS2	TOPS1	TOPS0	1111 1111
FD4h	SPBRGH1	USART1 Baud Rate Generator High Byte								0000 0000
FD3h	OSCCON	IDLEN	IRCF2	IRCF1	IRCF0	OSTS	HFIOFS	SCS1	SCS0	0110 q000
FD2h	IPR5	TMR7GIP(3)	TMR12IP(3)	TMR10IP(3)	TMR8IP	TMR7IP(3)	TMR6IP	TMR5IP	TMR4IP	1111 1111
FD1h	WDTCON	REGSLP	—	ULPLVL	SRETEN	—	ULPEN	ULPSINK	SWDTEN	0-x0 -000
FD0h	RCON	IPEN	SBOREN	$\overline{CM}$	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	0111 11qq
FCFh	TMR1H	Timer1 Register High Byte								xxxx xxxx
FCEh	TMR1L	Timer1 Register Low Byte								xxxx xxxx
FCDh	T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	SOSCEN	$\overline{T1SYNC}$	RD16	TMR1ON	0000 0000
FCCh	TMR2	Timer2 Register								0000 0000
FCBh	PR2	Timer2 Period Register								1111 1111
FCAh	T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000
FC9h	SSP1BUF	MSSP Receive Buffer/Transmit Register								xxxx xxxx
FC8h	SSP1ADD	MSSP Address Register in I^2C™ Slave Mode. SSP1 Baud Rate Reload Register in I^2C Master Mode.								0000 0000
FC7h	SSP1STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000
FC6h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000
FC5h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000
FC4h	ADRESH	A/D Result Register High Byte								xxxx xxxx
FC3h	ADRESL	A/D Result Register Low Byte								xxxx xxxx
FC2h	ADCON0	—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	-000 0000
FC1h	ADCON1	TRIGSEL1	TRIGSEL0	VCFG1	VCFG0	VNCFG	CHSN2	CHSN1	CHSN0	0000 0000
FC0h	ADCON2	ADFM	—	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0	0-00 0000
FBFh	ECCP1AS	ECCP1ASE	ECCP1AS2	ECCP1AS1	ECCP1AS0	PSS1AC1	PSS1AC0	PSS1BD1	PSS1BD0	0000 0000
FBEh	ECCP1DEL	P1RSEN	P1DC6	P1DC5	P1DC4	P1DC3	P1DC2	P1DC1	P1DC0	0000 0000
FBDh	CCPR1H	Capture/Compare/PWM Register1 High Byte								xxxx xxxx
FBCh	CCPR1L	Capture/Compare/PWM Register1 Low Byte								xxxx xxxx
FBBh	CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000
FBAh	PIR5	TMR7GIF(3)	TMR12IF(3)	TMR10IF(3)	TMR8IF	TMR7IF(3)	TMR6IF	TMR5IF	TMR4IF	0000 0000
FB9h	PIE5	TMR7GIE(3)	TMR12IE(3)	TMR10IE(3)	TMR8IE	TMR7IE(3)	TMR6IE	TMR5IE	TMR4IE	0000 0000
FB8h	IPR4	CCP10IP(3)	CCP9IP(3)	CCP8IP	CCP7IP	CCP6IP	CCP5IP	CCP4IP	CCP3IP	1111 1111
FB7h	PIR4	CCP10IF(3)	CCP9IF(3)	CCP8IF	CCP7IF	CCP6IF	CCP5IF	CCP4IF	CCP3IF	0000 0000

Note 1: This bit is available when Master Clear is disabled (MCLRE = 0). When MCLRE is set, the bit is unimplemented.
2: Unimplemented on 64-pin devices (PIC18F6XK22), read as '0'.
3: Unimplemented on devices with a program memory of 32 Kbytes (PIC18FX5K22).

6.4.3.1 FSR Registers and the INDF Operand

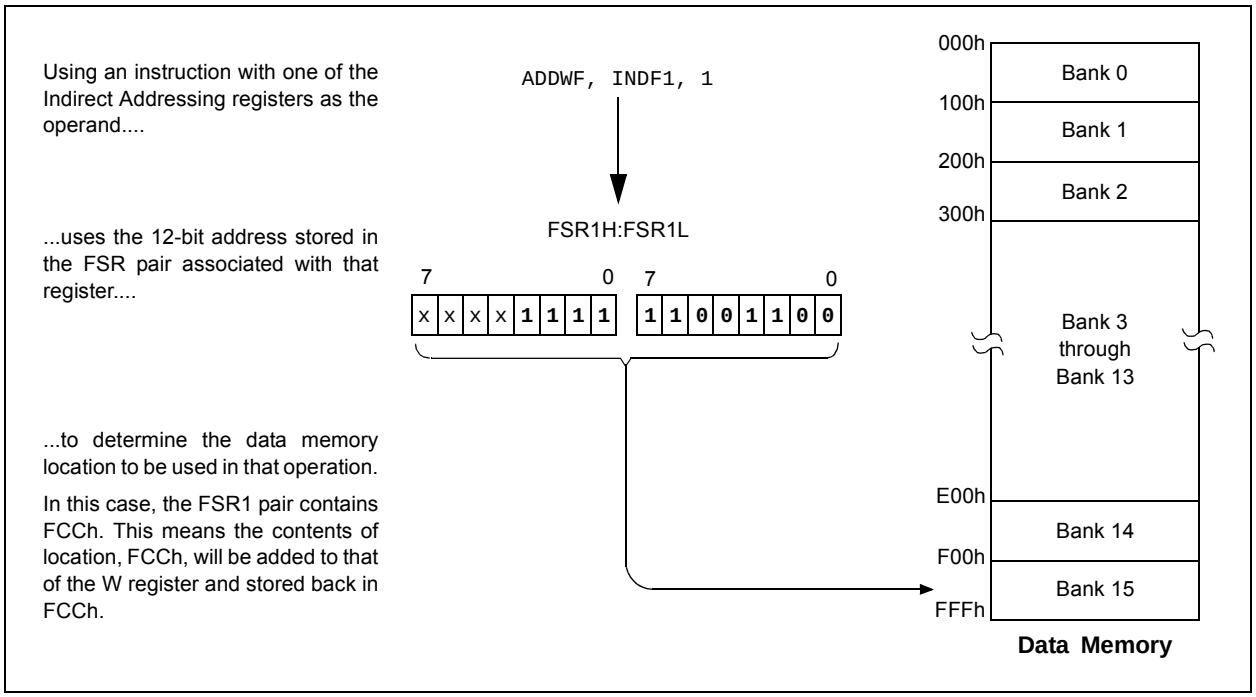
At the core of Indirect Addressing are three sets of registers: FSR0, FSR1 and FSR2. Each represents a pair of 8-bit registers: FSRnH and FSRnL. The four upper bits of the FSRnH register are not used, so each FSR pair holds a 12-bit value. This represents a value that can address the entire range of the data memory in a linear fashion. The FSR register pairs, then, serve as pointers to data memory locations.

Indirect Addressing is accomplished with a set of Indirect File Operands, INDF0 through INDF2. These can be thought of as “virtual” registers. The operands are mapped in the SFR space, but are not physically implemented. Reading or writing to a particular INDF register actually accesses its corresponding FSR register pair. A read from INDF1, for example, reads the data at the address indicated by FSR1H:FSR1L.

Instructions that use the INDF registers as operands actually use the contents of their corresponding FSR as a pointer to the instruction’s target. The INDF operand is just a convenient way of using the pointer.

Because Indirect Addressing uses a full 12-bit address, data RAM banking is not necessary. Thus, the current contents of the BSR and the Access RAM bit have no effect on determining the target address.

FIGURE 6-8: INDIRECT ADDRESSING



REGISTER 18-8: MONTH: MONTH VALUE REGISTER(1)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	MTHTEN0	MTHONE3	MTHONE2	MTHONE1	MTHONE0
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 **MTHTEN0:** Binary Coded Decimal Value of Month's Tens Digit bit
Contains a value of 0 or 1.

bit 3-0 **MTHONE<3:0>:** Binary Coded Decimal Value of Month's Ones Digit bits
Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 18-9: DAY: DAY VALUE REGISTER(1)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
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 **Unimplemented:** Read as '0'

bit 5-4 **DAYTEN<1:0>:** Binary Coded Decimal value of Day's Tens Digit bits
Contains a value from 0 to 3.

bit 3-0 **DAYONE<3:0>:** Binary Coded Decimal Value of Day's Ones Digit bits
Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 18-10: WEEKDAY: WEEKDAY VALUE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 7							bit 0

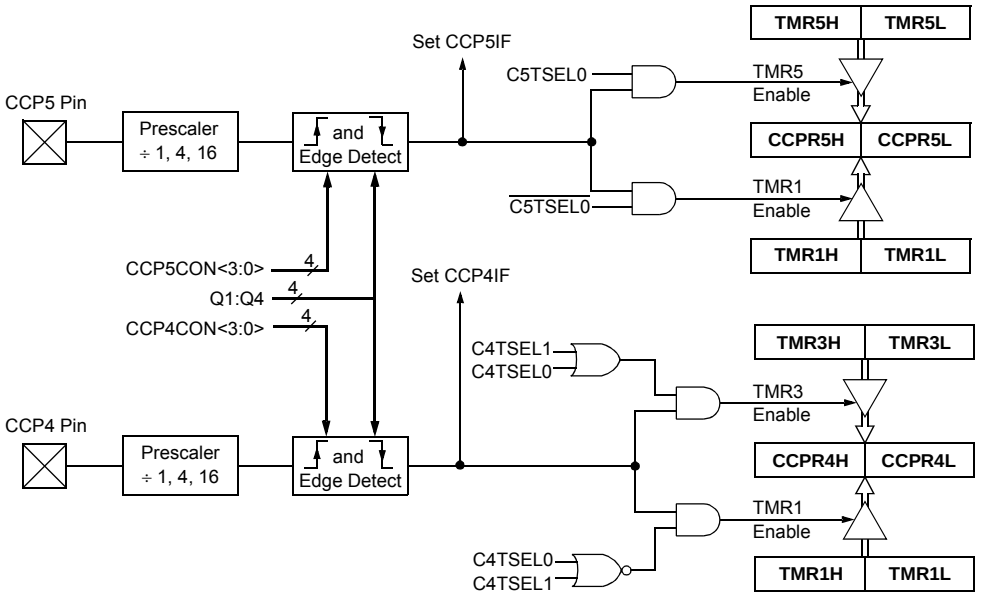
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-3 **Unimplemented:** Read as '0'

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit bits
Contains a value from 0 to 6.

Note 1: A write to this register is only allowed when RTCWREN = 1.

FIGURE 19-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



Note: This block diagram uses CCP4 and CCP5, and their appropriate timers as an example. For details on all of the CCP modules and their timer assignments, see Table 19-2 and Table 19-3.

19.2.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCP4IE bit (PIE4<1>) clear to avoid false interrupts and should clear the flag bit, CCP4IF, following any such change in operating mode.

19.2.4 CCP PRESCALER

There are four prescaler settings in Capture mode. They are specified as part of the operating mode selected by the mode select bits (CCP4M<3:0>). Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. This means that any Reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Doing that will also not clear the prescaler counter – meaning the first capture may be from a non-zero prescaler.

Example 19-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

EXAMPLE 19-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF   CCP4CON      ; Turn CCP module off
MOVLW  NEW_CAPT_PS  ; Load WREG with the
                    ; new prescaler mode
                    ; value and CCP ON
MOVWF  CCP4CON      ; Load CCP4CON with
                    ; this value
```

21.4.3.4 7-Bit Address Masking Mode

Unlike 5-bit masking, 7-Bit Address Masking mode uses a mask of up to 8 bits (in 10-bit addressing) to define a range of addresses that can be Acknowledged, using the lowest bits of the incoming address. This allows the module to Acknowledge up to 127 different addresses with 7-bit addressing, or 255 with 10-bit addressing (see Example 21-3). This mode is the default configuration of the module, which is selected when MSSPMSK is unprogrammed ('1').

The address mask for 7-Bit Address Masking mode is stored in the SSPxMSK register, instead of the SSPxCON2 register. SSPxMSK is a separate hardware register within the module, but it is not directly addressable. Instead, it shares an address in the SFR space with the SSPxADD register. To access the SSPxMSK register, it is necessary to select MSSP mode, '1001' (SSPxCON1<3:0> = 1001) and then read or write to the location of SSPxADD.

To use 7-Bit Address Masking mode, it is necessary to initialize SSPxMSK with a value before selecting the I^2C Slave Addressing mode. Thus, the required sequence of events is:

1. Select SSPxMSK Access mode (SSPxCON2<3:0> = 1001).
2. Write the mask value to the appropriate SSPADD register address (FC8h for MSSP1, F6Eh for MSSP2).
3. Set the appropriate I^2C Slave mode (SSPxCON2<3:0> = 0111 for 10-bit addressing, '0110' for 7-bit addressing).

Setting or clearing mask bits in SSPxMSK behaves in the opposite manner of the ADMSK bits in 5-Bit Address Masking mode. That is, clearing a bit in SSPxMSK causes the corresponding address bit to be masked; setting the bit requires a match in that position. SSPxMSK resets to all '1's upon any Reset condition and, therefore, has no effect on the standard MSSP operation until written with a mask value.

With 7-bit addressing, SSPxMSK<7:1> bits mask the corresponding address bits in the SSPxADD register. For any SSPxMSK bits that are active (SSPxMSK<n> = 0), the corresponding SSPxADD address bit is ignored (SSPxADD<n> = x). For the module to issue an address Acknowledge, it is sufficient to match only on addresses that do not have an active address mask.

With 10-bit addressing, SSPxMSK<7:0> bits mask the corresponding address bits in the SSPxADD register. For any SSPxMSK bits that are active (= 0), the corresponding SSPxADD address bit is ignored (SSPxADD<n> = x).

Note:	The two Most Significant bits of the address are not affected by address masking.

EXAMPLE 21-3: ADDRESS MASKING EXAMPLES IN 7-BIT MASKING MODE

7-Bit Addressing:

SSPxADD<7:1> = 1010 000

SSPxMSK<7:1> = 1111 001

Addresses Acknowledged = ACh, A8h, A4h, A0h

10-Bit Addressing:

SSPxADD<7:0> = 1010 0000 (The two MSb are ignored in this example since they are not affected)

SSPxMSK<5:1> = 1111 0011

Addresses Acknowledged = ACh, A8h, A4h, A0h

FIGURE 21-13: I²C™ SLAVE MODE TIMING (TRANSMISSION, 10-BIT ADDRESS)

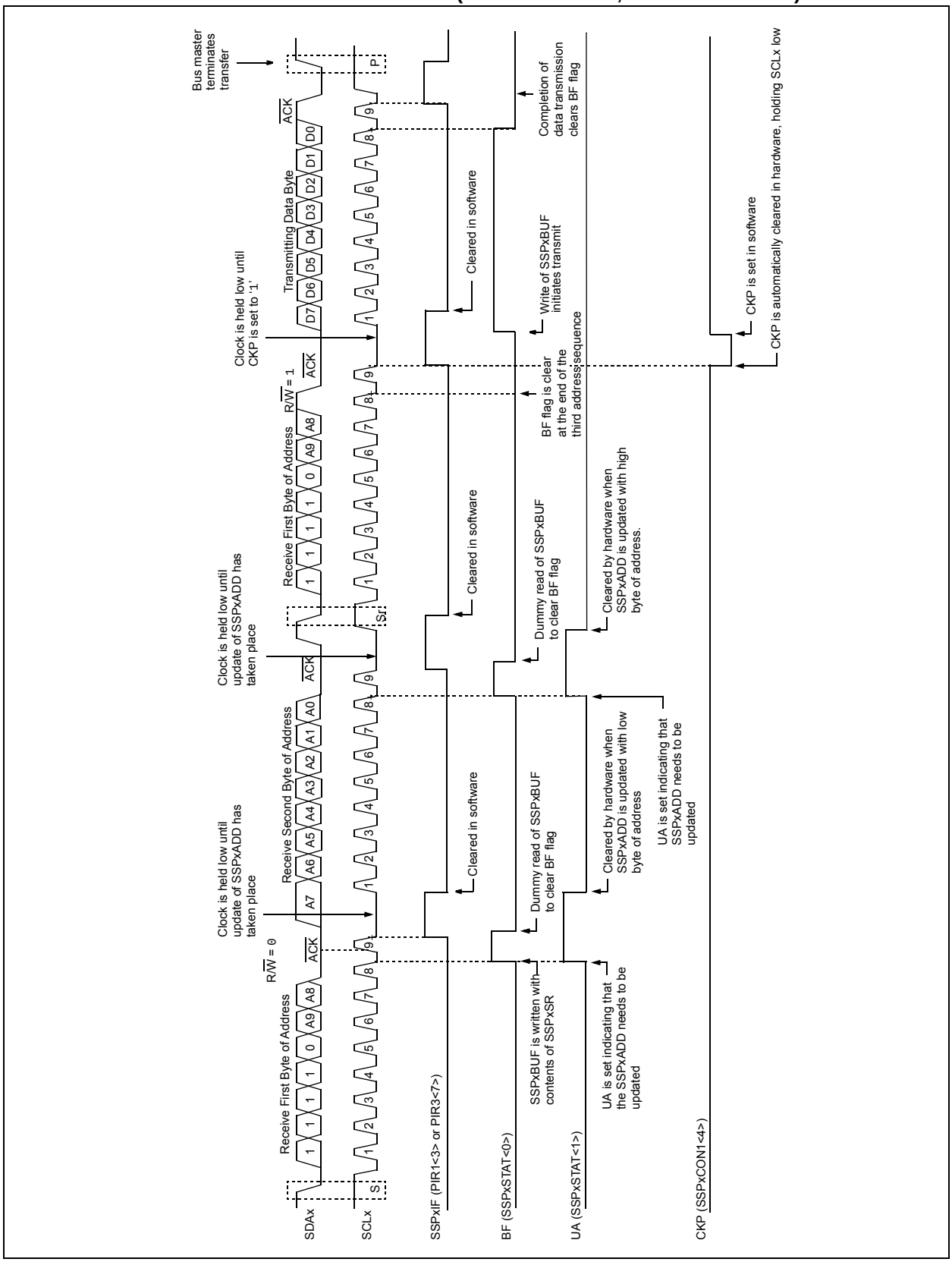
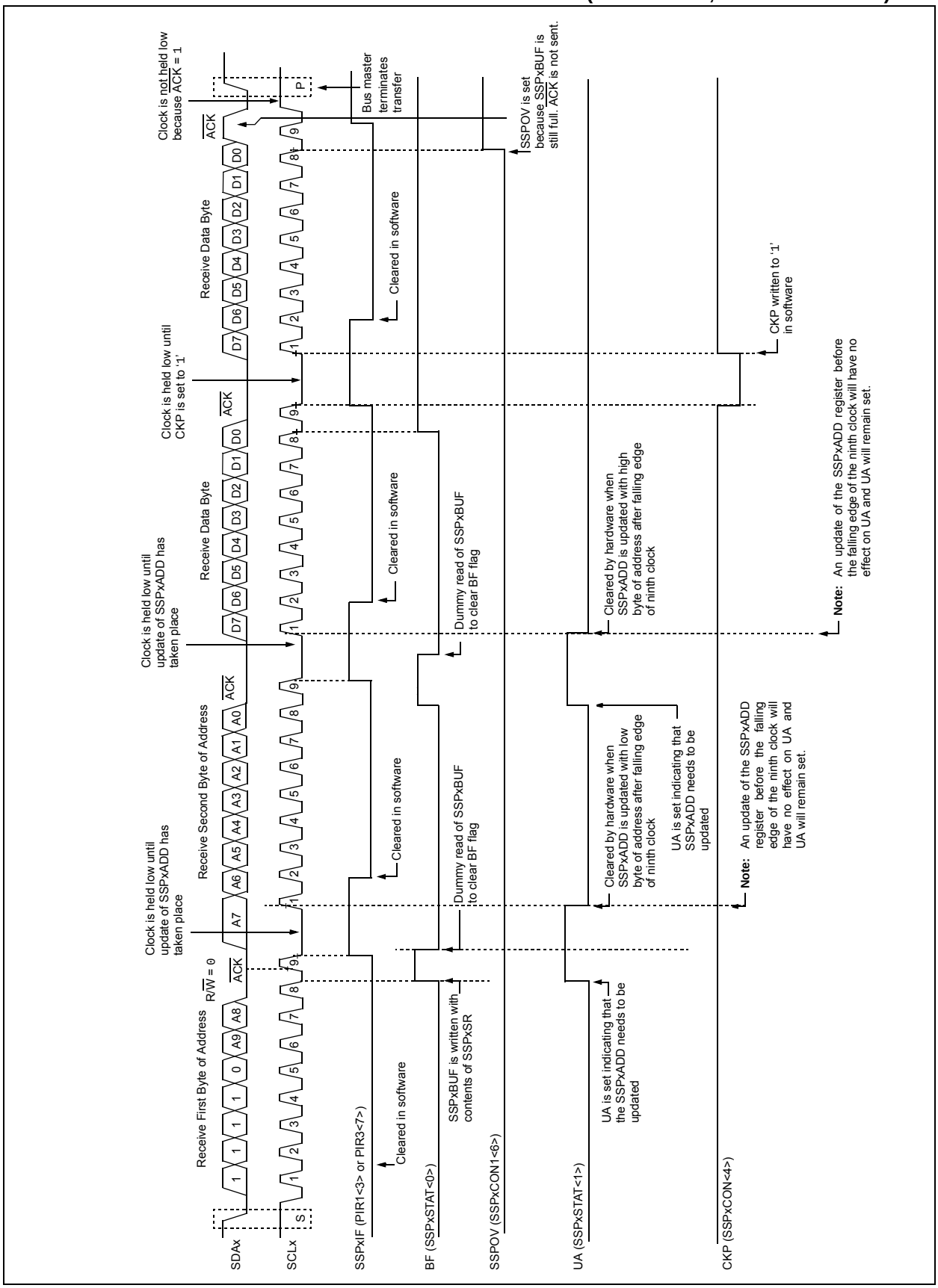


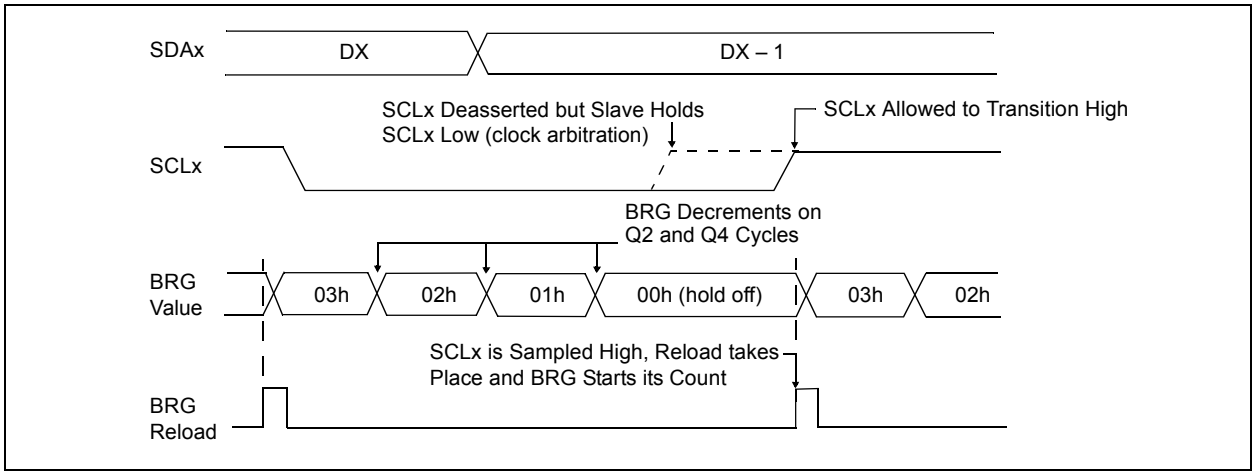
FIGURE 21-16: I²C™ SLAVE MODE TIMING WITH SEN = 1 (RECEPTION, 10-BIT ADDRESS)



21.4.7.2 Clock Arbitration

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, deasserts the SCLx pin (SCLx allowed to float high). When the SCLx pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCLx pin is actually sampled high. When the SCLx pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<6:0> and begins counting. This ensures that the SCLx high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 21-20).

FIGURE 21-20: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



23.0 12-BIT ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The Analog-to-Digital (A/D) Converter module in the PIC18F87K22 family of devices has 16 inputs for the 64-pin devices and 24 inputs for the 80-pin devices. This module allows conversion of an analog input signal to a corresponding 12-bit digital number.

The module has these registers:

- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)
- A/D Control Register 2 (ADCON2)
- A/D Port Configuration Register 0 (ANCON0)
- A/D Port Configuration Register 1 (ANCON1)
- A/D Port Configuration Register 2 (ANCON2)
- ADRESH (the upper, A/D Results register)
- ADRESL (the lower, A/D Results register)

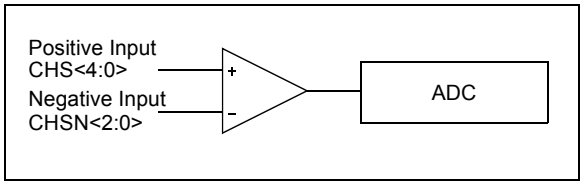
The ADCON0 register, shown in Register 23-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 23-2, configures the voltage reference and special trigger selection. The ADCON2 register, shown in Register 23-3, configures the A/D clock source and programmed acquisition time and justification.

23.1 Differential A/D Converter

The converter in PIC18F87K22 family devices is implemented as a differential A/D where the differential voltage between two channels is measured and converted to digital values (see Figure 23-1).

The converter can also be configured to measure a voltage from a single input by clearing the CHSN bits (ADCON1<2:0>). With this configuration, the negative channel input is connected internally to AVSS (see Figure 23-2).

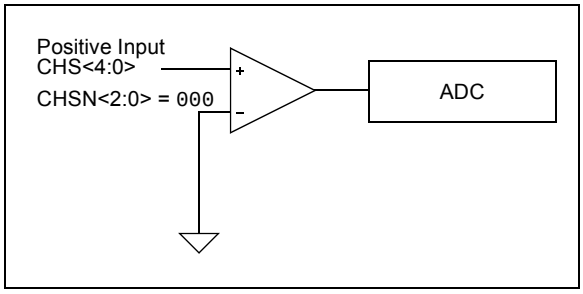
FIGURE 23-1: DIFFERENTIAL CHANNEL MEASUREMENT



Differential conversion feeds the two input channels to a unity gain differential amplifier. The positive channel input is selected using the CHS bits (ADCON0<6:2>) and the negative channel input is selected using the CHSN bits (ADCON1<2:0>).

The output from the amplifier is fed to the A/D Converter, as shown in Figure 23-1. The 12-bit result is available on the ADRESH and ADRESL registers. An additional bit indicates if the 12-bit result is a positive or negative value.

FIGURE 23-2: SINGLE CHANNEL MEASUREMENT



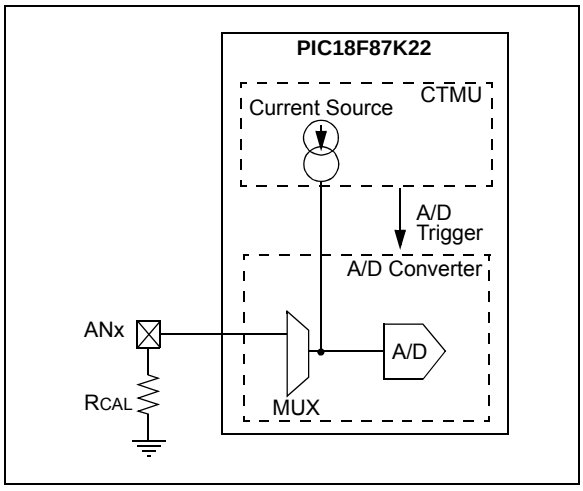
In the Single Channel Measurement mode, the negative input is connected to AVSS by clearing the CHSN bits (ADCON1<2:0>).

The CTMU current source may be trimmed with the trim bits in CTMUICON using an iterative process to get the exact current desired. Alternatively, the nominal value without adjustment may be used. That value may be stored by software for use in all subsequent capacitive or time measurements.

To calculate the value for RCAL, the nominal current must be chosen. Then, the resistance can be calculated.

For example, if the A/D Converter reference voltage is 3.3V, use 70% of full scale (or 2.31V) as the desired approximate voltage to be read by the A/D Converter. If the range of the CTMU current source is selected to be 0.55 μA, the resistor value needed is calculated as RCAL = 2.31V/0.55 μA, for a value of 4.2 MΩ. Similarly, if the current source is chosen to be 5.5 μA, RCAL would be 420,000Ω, and 42,000Ω if the current source is set to 55 μA.

FIGURE 27-2: CTMU CURRENT SOURCE CALIBRATION CIRCUIT



A value of 70% of full-scale voltage is chosen to make sure that the A/D Converter was in a range that is well above the noise floor. If an exact current is chosen to incorporate the trimming bits from CTMUICON, the resistor value of RCAL may need to be adjusted accordingly. RCAL may also be adjusted to allow for available resistor values. RCAL should be of the highest precision available, in light of the precision needed for the circuit that the CTMU will be measuring. A recommended minimum would be 0.1% tolerance.

The following examples show a typical method for performing a CTMU current calibration.

- Example 27-1 demonstrates how to initialize the A/D Converter and the CTMU.

  This routine is typical for applications using both modules.

- Example 27-2 demonstrates one method for the actual calibration routine.

  This method manually triggers the A/D Converter to demonstrate the entire step-wise process. It is also possible to automatically trigger the conversion by setting the CTMU's CTTRIG bit (CTMUCONH<0>).

REGISTER 28-14: DEVID1: DEVICE ID REGISTER 1 FOR THE PIC18F87K22 FAMILY

R	R	R	R	R	R	R	R
DEV2	DEV1	DEV0	REV4	REV3	REV2	REV1	REV0
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 **DEV<2:0>:** Device ID bits

Devices with DEV<10:3> of '0101 0010' (see DEVID2):
010 = PIC18F65K22
000 = PIC18F66K22
101 = PIC18F85K22
011 = PIC18F86K22

Devices with DEV<10:3> of '0101 0001':
000 = PIC18F67K22
010 = PIC18F87K22

bit 4-0 **REV<4:0>:** Revision ID bits

These bits are used to indicate the device revision.

REGISTER 28-15: DEVID2: DEVICE ID REGISTER 2 FOR THE PIC18F87K22 FAMILY

R	R	R	R	R	R	R	R
DEV10(1)	DEV9(1)	DEV8(1)	DEV7(1)	DEV6(1)	DEV5(1)	DEV4(1)	DEV3(1)
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 **DEV<10:3>:** Device ID bits(1)

These bits are used with the DEV<2:0> bits in the Device ID Register 1 to identify the part number.
0101 0010 = PIC18F65K22, PIC18F66K22, PIC18F85K22 and PIC18F86K22
0101 0001 = PIC18F67K22 and PIC18F87K22

Note 1: These values for DEV<10:3> may be shared with other devices. The specific device is always identified by using the entire DEV<10:0> bit sequence.

28.3 On-Chip Voltage Regulator

All of the PIC18F87K22 family devices power their core digital logic at a nominal 3.3V. For designs that are required to operate at a higher typical voltage, such as 5V, all family devices incorporate two on-chip regulators that allow the device to run its core logic from VDD. Those regulators are:

- Normal On-Chip Regulator
- Ultra Low-Power On-Chip Regulator

The hardware configuration of these regulators is the same and is explained in **Section 28.3.1 "Regulator Enable/Disable By Hardware"**. The regulators' only differences relate to when the device enters Sleep, as explained in **Section 28.3.2 "Operation of Regulator in Sleep"**.

28.3.1 REGULATOR ENABLE/DISABLE BY HARDWARE

The regulator can be enabled or disabled only by hardware. The regulator is controlled by the ENVREG pin and the VDDCORE/VCAP pin.

28.3.1.1 Regulator Enable Mode

Tying VDD to the pin enables the regulator, which in turn, provides power to the core from the other VDD pins.

When the regulator is enabled, a low-ESR filter capacitor must be connected to the VDDCORE/VCAP pin (see Figure 28-2). This helps maintain the regulator's stability. The recommended value for the filter capacitor is given in **Section 31.2 "DC Characteristics: Power-Down and Supply Current PIC18F87K22 Family (Industrial/Extended)"**.

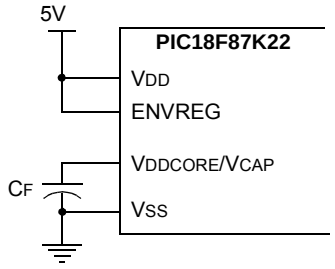
28.3.1.2 Regulator Disable Mode

If the regulator is disabled by connecting VSS to the ENVREG pin, the power to the core is supplied directly by VDD. The voltage levels for VDD must not exceed the specified VDDCORE levels. In Regulator Disabled mode, a 0.1 µF capacitor should be connected to the VDDCORE/VCAP pin (see Figure 28-2).

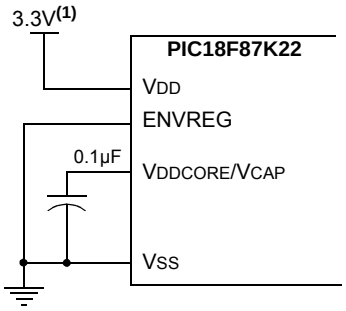
When the regulator is being used, the overall voltage budget is very tight. The regulator should operate the device down to 1.8V. When VDD drops below 3.3V, the regulator no longer regulates, but the output voltage follows the input until VDD reaches 1.8V. Below this voltage, the output of the regulator output may drop to 0V.

FIGURE 28-2: CONNECTIONS FOR THE ON-CHIP REGULATOR

Regulator Enabled (ENVREG tied to VDD):



Regulator Disabled (ENVREG tied to VSS):



Note 1: These are typical operating voltages. For the full operating ranges of VDD and VDDCORE, see **Section 31.2 "DC Characteristics: Power-Down and Supply Current PIC18F87K22 Family (Industrial/Extended)"**.

TABLE 28-4: SUMMARY OF CODE PROTECTION REGISTERS

File Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
300008h	CONFIG5L	CP7(1)	CP6(1)	CP5(1)	CP4(1)	CP3	CP2	CP1	CP0
300009h	CONFIG5H	CPD	CPB	—	—	—	—	—	—
30000Ah	CONFIG6L	WRT7(1)	WRT6(1)	WRT5(1)	WRT4(1)	WRT3	WRT2	WRT1	WRT0
30000Bh	CONFIG6H	WRTD	WRTB	WRTC	—	—	—	—	—
30000Ch	CONFIG7L	EBTR7(1)	EBTR6(1)	EBTR5(1)	EBTR4(1)	EBTR3	EBTR2	EBTR1	EBTR0
30000Dh	CONFIG7H	—	EBTRB	—	—	—	—	—	—

Legend: Shaded cells are unimplemented.

Note 1: This bit is available only on the PIC18F67K22 and PIC18F87K22 devices.

28.6.1 PROGRAM MEMORY CODE PROTECTION

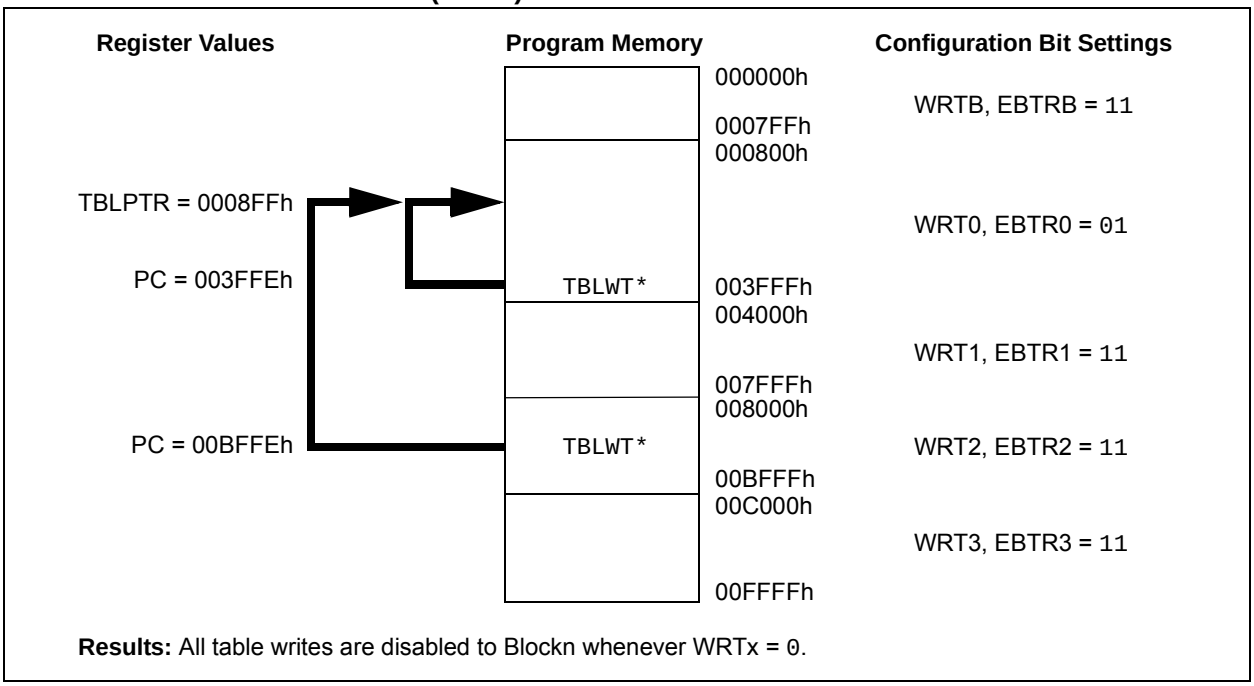
The program memory may be read to, or written from, any location using the table read and table write instructions. The Device ID may be read with table reads. The Configuration registers may be read and written with the table read and table write instructions.

In normal Execution mode, the CPx bits have no direct effect. CPx bits inhibit external reads and writes. A block of user memory may be protected from table writes if the WRTx Configuration bit is '0'.

The EBTRx bits control table reads. For a block of user memory, with the EBTRx bit set to '0', a table read instruction that executes from within that block is allowed to read. A table read instruction that executes from a location outside of that block is not allowed to read and will result in reading '0's. Figures 28-7 through 28-9 illustrate table write and table read protection.

Note: Code protection bits may only be written to a '0' from a '1' state. It is not possible to write a '1' to a bit in the '0' state. Code protection bits are only set to '1' by a full chip erase or block erase function. The full chip erase and block erase functions can only be initiated via ICSP or an external programmer. Refer to the device programming specification for more information.

FIGURE 28-7: TABLE WRITE (WRTx) DISALLOWED



Results: All table writes are disabled to Blockn whenever WRTx = 0.

BRA **Unconditional Branch**

Syntax: BRA n

Operands: $-1024 \le n \le 1023$

Operation: $(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding:

1101	0nnn	nnnn	nnnn

Description: Add the 2's complement number '2n' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is a two-cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

Example: HERE BRA Jump

Before Instruction
PC = address (HERE)
After Instruction
PC = address (Jump)

BSF **Bit Set f**

Syntax: BSF f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b \le 7$
$a \in [0,1]$

Operation: $1 \rightarrow f<b>$

Status Affected: None

Encoding:

1000	bbba	ffff	ffff

Description: Bit 'b' in register 'f' is set.

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 $f \le 95$ (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 register 'f'

Example: BSF FLAG_REG, 7, 1

Before Instruction
FLAG_REG = 0Ah
After Instruction
FLAG_REG = 8Ah

BTFSC **Bit Test File, Skip if Clear**

Syntax: BTFSC f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b \le 7$
$a \in [0,1]$

Operation: skip if (f<b>) = 0

Status Affected: None

Encoding:

1011	bbba	ffff	ffff

Description: If bit 'b' in register 'f' is '0', then the next instruction is skipped. If bit 'b' is '0', then the next instruction fetched during the current instruction execution is discarded and a NOP is executed instead, making this a two-cycle instruction.

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 $f \le 95$ (5Fh). See **Section 29.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1(2)
Note: 3 cycles if skip and followed by a 2-word instruction.

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE    BTFSC   FLAG, 1, 0
FALSE   :
TRUE    :
```
Before Instruction
PC = address (HERE)
After Instruction
If FLAG<1> = 0;
PC = address (TRUE)
If FLAG<1> = 1;
PC = address (FALSE)

BTFSS **Bit Test File, Skip if Set**

Syntax: BTFSS f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b < 7$
$a \in [0,1]$

Operation: skip if (f<b>) = 1

Status Affected: None

Encoding:

1010	bbba	ffff	ffff

Description: If bit 'b' in register 'f' is '1', then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution is discarded and a NOP is executed instead, making this a two-cycle instruction.

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 $f \le 95$ (5Fh). See **Section 29.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1(2)
Note: 3 cycles if skip and followed by a 2-word instruction.

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

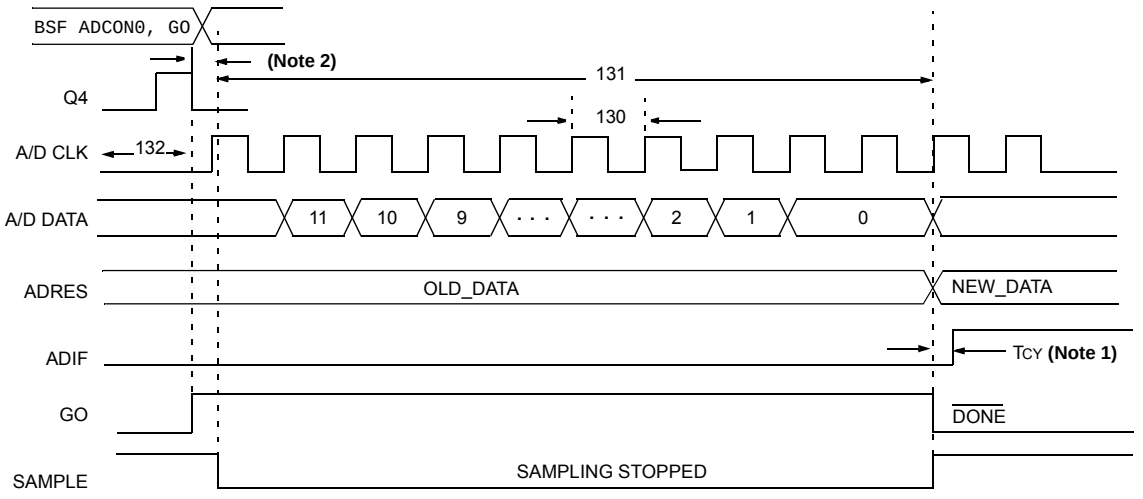
Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE    BTFSS   FLAG, 1, 0
FALSE   :
TRUE    :
```
Before Instruction
PC = address (HERE)
After Instruction
If FLAG<1> = 0;
PC = address (FALSE)
If FLAG<1> = 1;
PC = address (TRUE)

FIGURE 31-24: A/D CONVERSION TIMING



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.

2: This is a minimal RC delay (typically 100 ns), which also disconnects the holding capacitor from the analog input.

TABLE 31-28: A/D CONVERSION REQUIREMENTS

Param No.	Symbol	Characteristic	Min	Max	Units	Conditions
130	TAD	A/D Clock Period	0.8	12.5(1)	µs	TOSC-based, VREF ≥ 3.0V
			1.4	25(1)	µs	VDD = 3.0V; TOSC-based, VREF full range
			—	**1**	µs	A/D RC mode
			—	3	µs	VDD = 3.0V; A/D RC mode
131	TCNV	Conversion Time (not including acquisition time)(2)	14	15	TAD	
132	TACQ	Acquisition Time(3)	1.4	—	µs	-40°C to +125°C
135	TSWC	Switching Time from Convert → Sample	—	**(Note 4)**		
137	TDIS	Discharge Time	0.2	—	µs	-40°C to +125°C

Note 1: The time of the A/D clock period is dependent on the device frequency and the TAD clock divider.

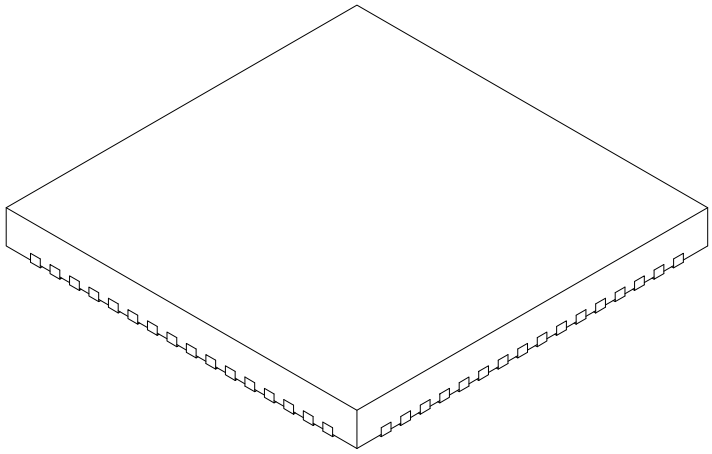
2: ADRES registers may be read on the following TCY cycle.

3: The time for the holding capacitor to acquire the "New" input voltage when the voltage changes full scale after the conversion (VDD to VSS or VSS to VDD). The source impedance (*RS*) on the input channels is 50Ω.

4: On the following cycle of the device clock.

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		64	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		9.00 BSC	
Exposed Pad Width	E2	7.05	7.15	7.50
Overall Length	D		9.00 BSC	
Exposed Pad Length	D2	7.05	7.15	7.50
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

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

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-149B Sheet 2 of 2