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	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	17
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 14x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16hv785t-i-so

PIC16F785/HV785

20-Pin Flash-Based 8-Bit CMOS Microcontroller

High-Performance RISC CPU:

- Only 35 Instructions to Learn:
  - All single-cycle instructions except branches
- Operating Speed:
  - DC – 20 MHz oscillator/clock input
  - DC – 200 ns instruction cycle
- Interrupt Capability
- 8-Level Seep Hardware Stack
- Direct, Indirect and Relative Addressing modes

Special Microcontroller Features:

- Precision Internal Oscillator:
  - Factory calibrated to ±1%
  - Software selectable frequency range of 8 MHz to 32 kHz
  - Software tunable
  - Two-Speed Start-up mode
  - Crystal fail detect for critical applications
  - Clock mode switching during operation for power savings
- Power-Saving Sleep mode
- Wide Operating Voltage Range (2.0V-5.5V)
- Industrial and Extended Temperature Range
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset (BOR) with Software Control Option
- Enhanced Low-Current Watchdog Timer (WDT) with on-chip Oscillator (software selectable nominal 268 seconds with full prescaler) with Software Enable
- Multiplexed Master Clear with Pull-up/Input Pin
- Programmable Code Protection
- High-Endurance Flash/EEPROM cell:
  - 100,000 write Flash endurance
  - 1,000,000 write EEPROM endurance
  - Flash/Data EEPROM retention: > 40 years

Low-Power Features:

- Standby Current:
  - 30 nA @ 2.0V, typical
- Operating Current:
  - 8.5 μA @ 32 kHz, 2.0V, typical
  - 100 μA @ 1 MHz, 2.0V, typical
- Watchdog Timer Current:
  - 1 μA @ 2.0V, typical
- Timer1 Oscillator Current:
  - 2 μA @ 32 kHz, 2.0V, typical

Peripheral Features:

- High-Speed Comparator module with:
  - Two independent analog comparators
  - Programmable on-chip voltage reference (CV_{REF}) module (% of V_{DD})
  - 1.2V band gap voltage reference
  - Comparator inputs and outputs externally accessible
  - < 40 ns propagation delay
  - 2 mv offset, typical
- Operational Amplifier module with 2 independent Op Amps:
  - 3 MHz GBWP, typical
  - All I/O pins externally accessible
- Two-Phase Asynchronous Feedback PWM module:
  - Complementary output with programmable dead band delay
  - Infinite resolution analog duty cycle
  - Sync Output/Input for multi-phase PWM
  - $F_{OSC}/2$ maximum PWM frequency
- A/D Converter:
  - 10-bit resolution and 14 channels (2 internal)
- 17 I/O pins and 1 Input-only Pin:
  - High-current source/sink for direct LED drive
  - Interrupt-on-pin change
  - Individually programmable weak pull-ups
- Timer0: 8-Bit Timer/Counter with 8-Bit Programmable Prescaler
- Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Gate Input mode
  - Option to use OSC1 and OSC2 in LP mode as Timer1 oscillator, if INTOSC mode selected
- Timer2: 8-Bit Timer/Counter with 8-Bit Period Register, Prescaler and Postscaler
- Capture, Compare, PWM module:
  - 16-bit Capture, max resolution 12.5 ns
  - Compare, max resolution 200 ns
  - 10-bit PWM with 1 output channel, max frequency 20 kHz
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Shunt Voltage Regulator (PIC16HV785 only):
  - 5 volt regulation
  - 4 mA to 50 mA shunt range

2.4 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

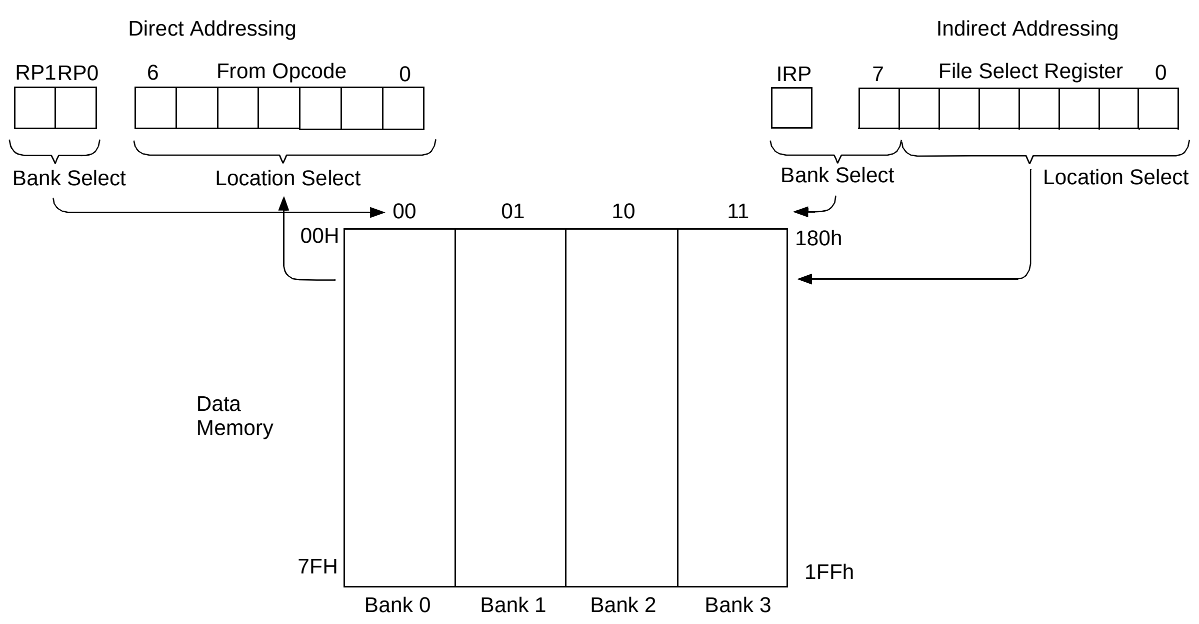
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR and the IRP bit in the STATUS Register, as shown in Figure 2-4.

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: INDIRECT ADDRESSING

```
     MOVLW  0x20   ;initialize pointer
     MOVWF  FSR    ;to RAM
NEXT CLRF   INDF   ;clear INDF register
     INCF   FSR    ;increment pointer
     BTFSS  FSR,4  ;all done?
     GOTO   NEXT   ;no clear next
CONTINUE           ;yes continue
```

FIGURE 2-4: DIRECT/INDIRECT ADDRESSING PIC16F785/HV785



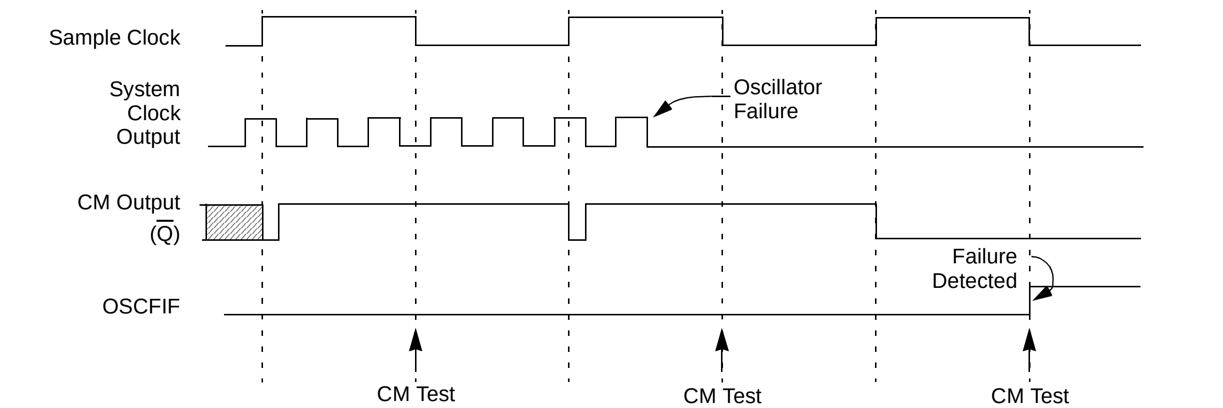
Note: For memory map detail see Figure 2-2.

3.7.1 FAIL-SAFE CONDITION CLEARING

The Fail-Safe condition is cleared after a Reset, the execution of a SLEEP instruction, or a modification of the SCS bit. While in Fail-Safe condition, the PIC16F785/HV785 uses the internal oscillator as the system clock source. The IRCF bits in the OSCCON Register can be modified to adjust the internal oscillator frequency without exiting the Fail-Safe condition.

The Fail-Safe condition must be cleared before the OSFIF flag can be cleared.

FIGURE 3-9: FSCM TIMING DIAGRAM



Note: The system clock is normally at a much higher frequency than the sample clock. The relative frequencies in this example have been chosen for clarity.

3.7.2 RESET OR WAKE-UP FROM SLEEP

The FSCM is designed to detect oscillator failure at any point after the device has exited a Reset or Sleep condition and the Oscillator Start-up Timer (OST) has expired. If the external clock is EC or RC mode, monitoring will begin immediately following these events.

For LP, XT or HS mode, the external oscillator may require a start-up time considerably longer than the FSCM sample clock time; a false clock failure may be detected (see Figure 3-9). To prevent this, the internal oscillator is automatically configured as the system clock and functions until the external clock is stable (the OST has timed out). This is identical to Two-Speed Start-up mode. Once the external oscillator is stable, the LFINTOSC returns to its role as the FSCM source.

Note:	Due to the wide range of oscillator start-up times, the Fail-Safe circuit is not active during oscillator start-up (i.e., after exiting Reset or Sleep). After an appropriate amount of time, the user should check the OSTS bit in the OSCCON Register to verify the oscillator start-up and system clock switchover has successfully completed.

4.2.3 PORTA PIN DESCRIPTIONS AND DIAGRAMS

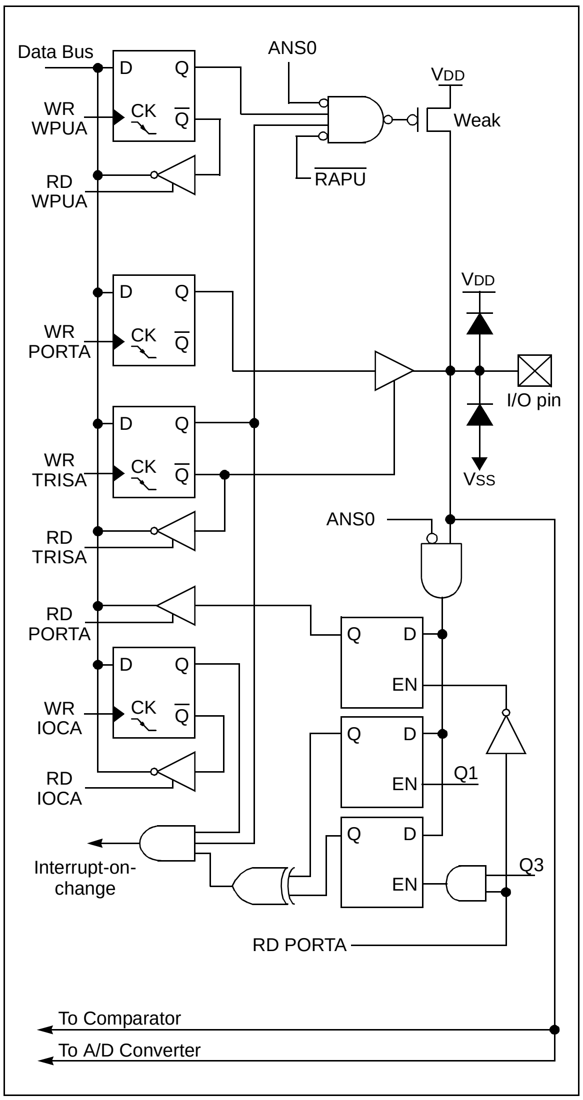
Each PORTA pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the comparator or the A/D, refer to the appropriate section in this Data Sheet.

4.2.3.1 RA0/AN0/C1IN+/ICSPDAT

Figure 4-1 shows the diagram for this pin. The RA0 pin is configurable to function as one of the following:

- General purpose I/O
- Analog input for the A/D
- Analog input to Comparator 1
- In-Circuit Serial Programming™ data

FIGURE 4-1: BLOCK DIAGRAM OF RA0

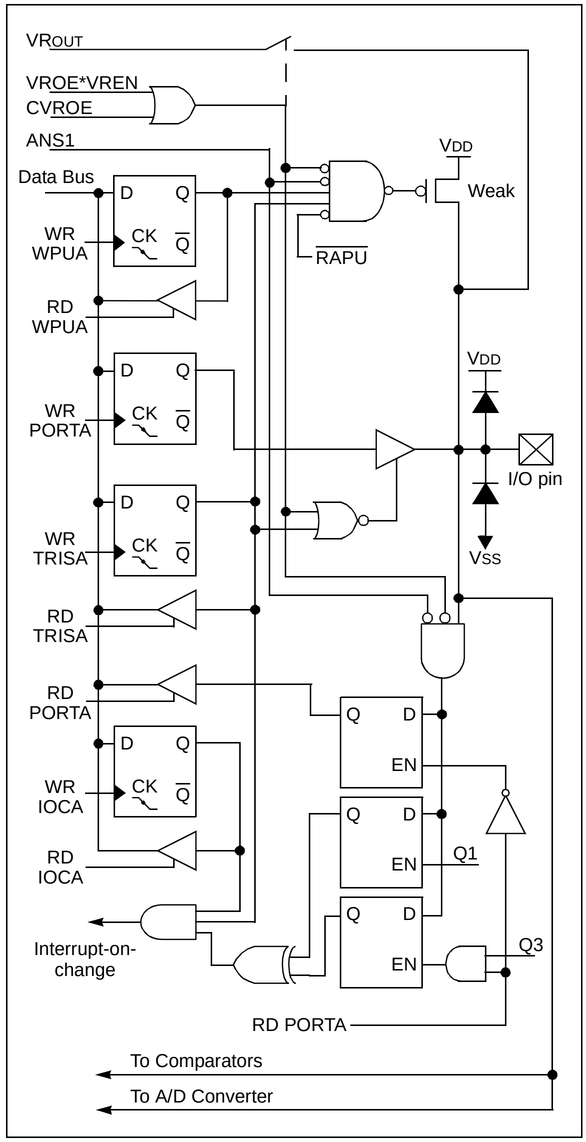


4.2.3.2 RA1/AN1/C12IN0-/VREF/ICSPCLK

Figure 4-1 shows the diagram for this pin. The RA1 pin is configurable to function as one of the following:

- General purpose I/O
- Analog input for the A/D
- Analog input to Comparators 1 and 2
- Voltage reference input for the A/D
- Buffered or unbuffered voltage reference output
- In-Circuit Serial Programming clock

FIGURE 4-2: BLOCK DIAGRAM OF RA1



4.3 PORTB and TRISB Registers

PORTB is a 4-bit wide, bidirectional port. The corresponding data direction register is TRISB (Register 4-6). Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin). Example 4-2 shows how to initialize PORTB.

Reading the PORTB register (Register 4-5) reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the port data latch.

Pin RB6 is an open drain output. All other PORTB pins have full CMOS output drivers.

The TRISB register controls the direction of the PORTB pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISB register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

> **Note:** The ANSEL1 (93h) register must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'.

EXAMPLE 4-2: INITIALIZING PORTB

```
BCF     STATUS,RP0    ;Bank 0
BCF     STATUS,RP1    ;
CLRF    PORTB         ;Init PORTB
BSF     STATUS,RP0    ;Bank 1
BCF     ANSEL1,2      ;digital I/O - RB4
BCF     ANSEL1,3      ;digital I/O - RB5
MOVLW   30h           ;Set RB<5:4> as inputs
MOVWF   TRISB         ;and set RB<7:6>
                      ;as outputs
BCF     STATUS,RP0    ;Bank 0
```

REGISTER 4-5: PORTB: PORTB REGISTER

R/W-x	R/W-x	R/W-x(1)	R/W-x(1)	U-0	U-0	U-0	U-0
RB7	RB6	RB5	RB4	—	—	—	—
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 7-4 **RB<7:4>:** PORTB General Purpose I/O Pin bits
1 = Port pin is greater than V_{IH}
0 = Port pin is less than V_{IL}

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

Note 1: Data latches are unknown after a POR, but each port bit reads '0' when the corresponding analog select bit is '1' (see Register 12-2 on page 82).

REGISTER 4-6: TRISB: PORTB TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—
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 7-4 **TRISB<7:4>:** PORTB Tri-State Control bits
1 = PORTB pin configured as an input (tri-stated)
0 = PORTB pin configured as an output

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

4.3.1 PORTB PIN DESCRIPTIONS AND DIAGRAMS

Each PORTB pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the PWM, operational amplifier, or the A/D, refer to the appropriate section in this Data Sheet.

4.3.1.1 RB4/AN10/OP2-

The RB4/AN10/OP2- pin is configurable to function as one of the following:

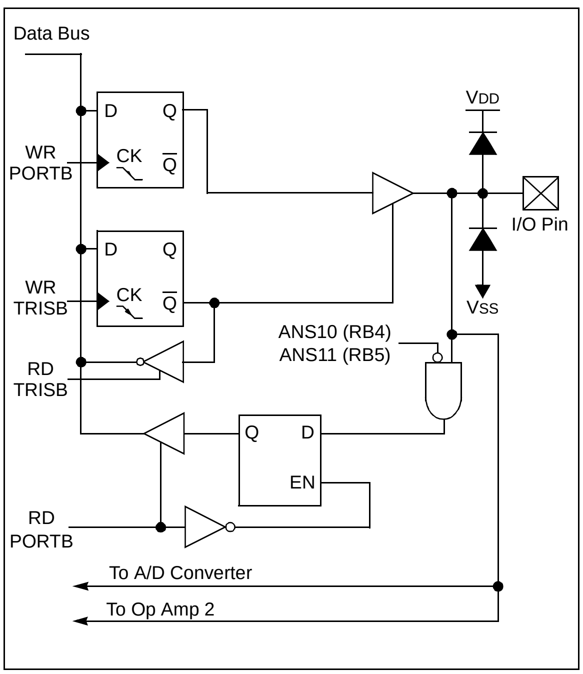
- General purpose I/O
- Analog input to the A/D
- Analog input to Op Amp 2

4.3.1.2 RB5/AN11/OP2+

The RB5/AN11/OP2+ pin is configurable to function as one of the following:

- General purpose I/O
- Analog input to the A/D
- Analog input to Op Amp 2

FIGURE 4-7: BLOCK DIAGRAM OF RB4 AND RB5

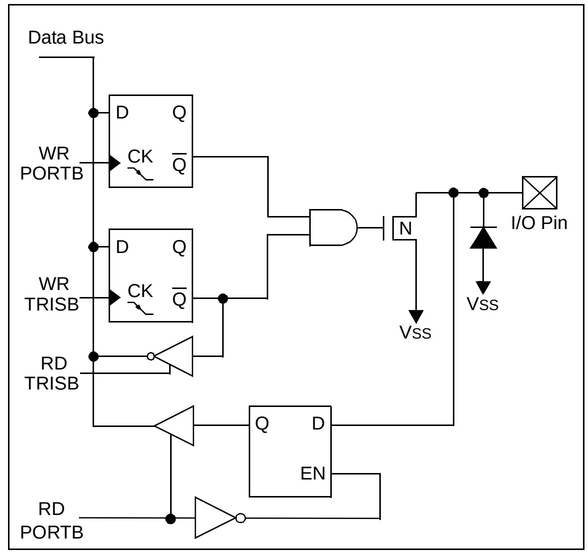


4.3.1.3 RB6

The RB6 pin is configurable to function as the following:

- Open drain general purpose I/O

FIGURE 4-8: BLOCK DIAGRAM OF RB6

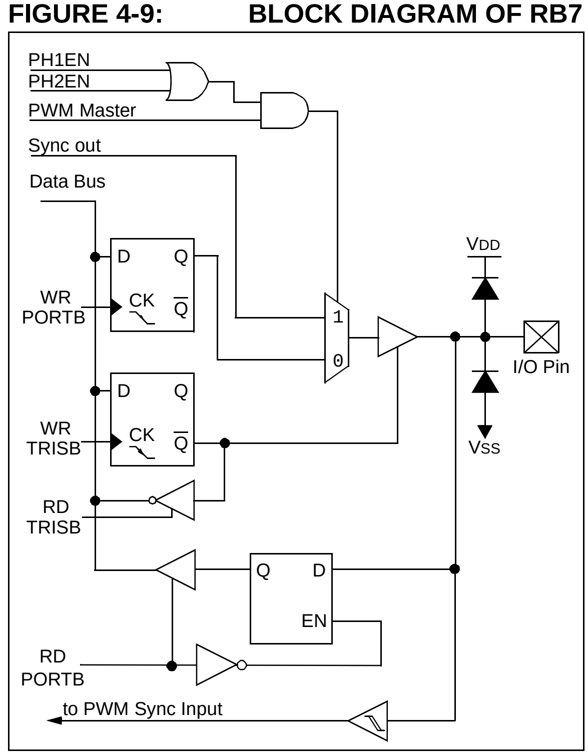


4.3.1.4 RB7/SYNC

The RB7/SYNC pin is configurable to function as one of the following:

- General purpose I/O
- PWM synchronization input and output

FIGURE 4-9: BLOCK DIAGRAM OF RB7



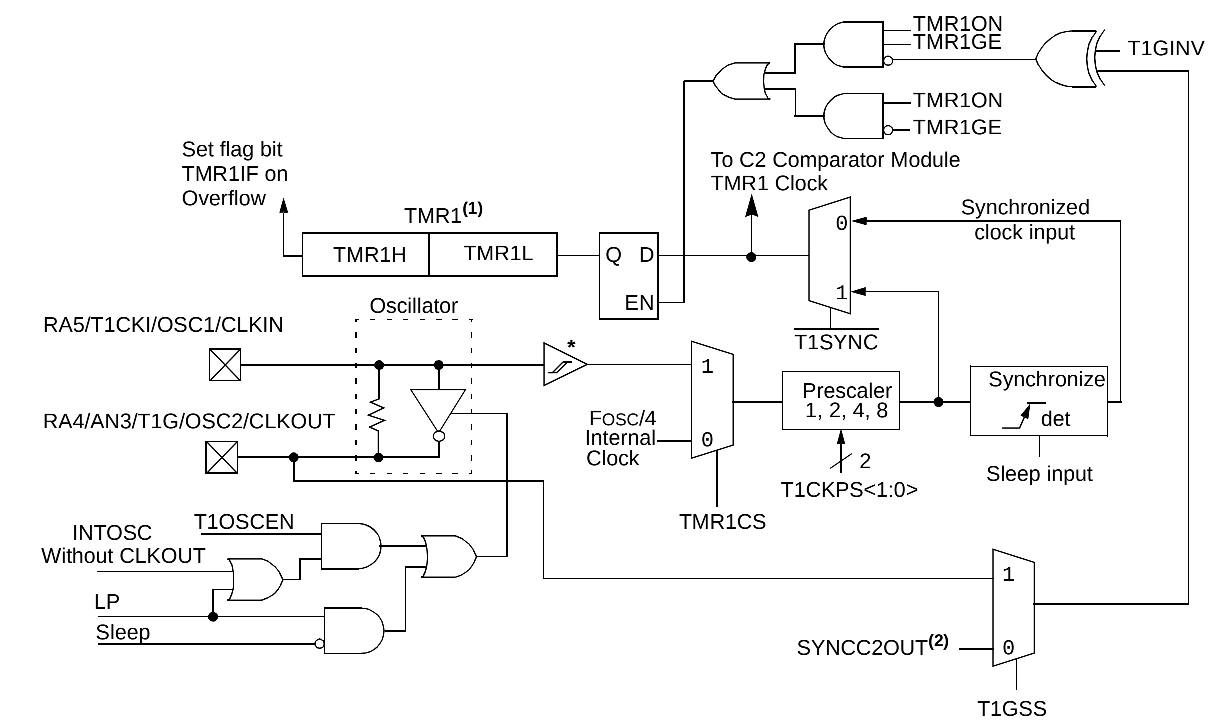
6.0 TIMER1 MODULE WITH GATE CONTROL

The Timer1 module is the 16-bit counter of the PIC16F785/HV785. Figure 6-1 shows the basic block diagram of the Timer1 module. Timer1 has the following features:

- 16-bit timer/counter (TMR1H:TMR1L)
- Readable and writable
- Internal or external clock selection
- Synchronous or asynchronous operation
- Interrupt on overflow from FFFFh to 0000h
- Wake-up upon overflow (Asynchronous mode)
- Optional external enable input:
  - Selectable gate source; T1G or C2 output (T1GSS)
  - Selectable gate polarity (T1GINV)
- Optional LP oscillator

The Timer1 Control register (T1CON), shown in Register 6-1, is used to enable/disable Timer1 and select the various features of the Timer1 module.

FIGURE 6-1: TIMER1 ON THE PIC16F785/HV785 BLOCK DIAGRAM



* ST Buffer is low power type when using LP OSC, or high-speed type when using T1CKI.

Note 1: Timer1 increments on the rising edge.

2: SYNCC2OUT is the synchronized output from Comparator 2 (See Figure 9-2 on 66).

6.1 Timer1 Modes of Operation

Timer1 can operate in one of three modes:

- 16-bit Timer with prescaler
- 16-bit Synchronous counter
- 16-bit Asynchronous counter

In Timer mode, Timer1 is incremented on every instruction cycle. In Counter mode, Timer1 is incremented on the rising edge of the external clock input T1CKI. In addition, the Counter mode clock can be synchronized to the microcontroller system clock or run asynchronously.

In Counter and Timer modules, the counter/timer clock can be gated by the Timer1 gate, which can be selected as either the T1G pin or Comparator 2 output.

If an external clock oscillator is needed (and the microcontroller is using the LP oscillator or INTOSC without CLKOUT), Timer1 can use the LP oscillator as a clock source.

> **Note:** In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions.
>
> - Timer1 enabled after POR Reset
> - Write to TMR1H or TMR1L
> - Timer1 is disabled (TMR1ON = `0`) when T1CKI is high then Timer1 is enabled (TMR1ON = `1`) when T1CKI is low. See Figure 6-2.

6.2 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 Register is set. To enable the interrupt on rollover, you must set these bits:

- Timer1 Interrupt Enable bit of the PIE1 Register
- PEIE bit of the INTCON Register
- GIE bit of the INTCON Register

The interrupt is cleared by clearing the TMR1IF in the Interrupt Service Routine.

> **Note:** The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

6.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits, of the T1CON Register, control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

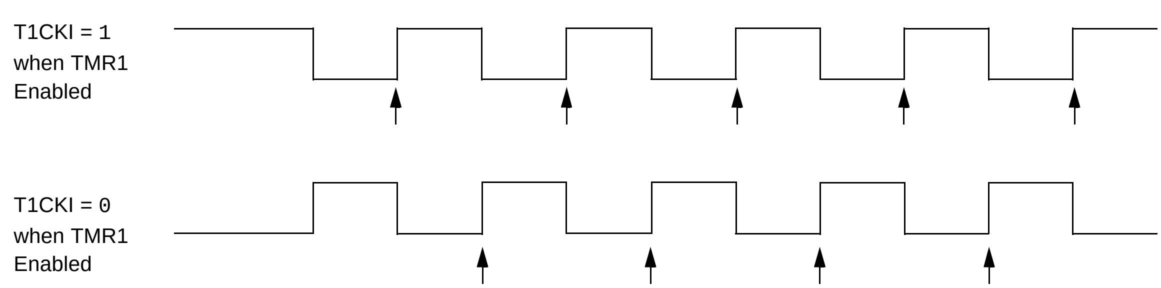
6.4 Timer1 Gate

Timer1 gate source is software configurable to be T1G pin or the output of Comparator 2. This allows the device to directly time external events using T1G or analog events using Comparator 2. See CM2CON1 (Register 9-3) for selecting the Timer1 gate source. This feature can simplify the software for a Delta-Sigma A/D Converter and many other applications. For more information on Delta-Sigma A/D Converters, see the Microchip web site (www.microchip.com).

> **Note:** TMR1GE bit, of the T1CON Register, must be set to use either T1G or C2OUT as the Timer1 gate source. See Register 9-3 for more information on selecting the Timer1 gate source.

Timer1 gate can be inverted using the T1GINV bit of the T1CON Register, whether it originates from the T1G pin or Comparator 2 output. This configures Timer1 to measure either the active high or active low time between events.

FIGURE 6-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.
2: See note box in **Section 6.1 “Timer1 Modes of Operation”**.

8.3.3 OPERATION IN SLEEP MODE

In Sleep mode, all clock sources are disabled. Timer2 will not increment and the state of the module will not change. If the RC5/CCP1 pin is driving a value, it will continue to drive that value. When the device wakes up, it will continue from this state.

8.3.3.1 OPERATION WITH FAIL-SAFE CLOCK MONITOR

If the Fail-Safe Clock Monitor is enabled, a clock failure will force the CCP to be clocked from the internal oscillator clock source, which may have a different clock frequency than the primary clock.

See **Section 3.0 “Clock Sources”** for additional details.

8.3.4 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

8.3.5 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Configure the PWM pin (RC5/CCP1) as an input by setting the TRISC<5> bit.
2. Set the PWM period by loading the PR2 register.
3. Configure the CCP module for the PWM mode by loading the CCP1CON register with the appropriate values.
4. Set the PWM duty cycle by loading the CCPR1L register and CCP1CON<5:4> bits.
5. Configure and start TMR2:
   - Clear the TMR2 interrupt flag bit by clearing the TMR2IF bit of the PIR1 Register.
   - Set the TMR2 prescale value by loading the T2CKPS bits of the T2CON Register.
   - Enable Timer2 by setting the TMR2ON bit of the T2CON Register.
6. Enable PWM output after a new PWM cycle has started:
   - Wait until TMR2 overflows (TMR2IF bit is set).
   - Enable the RC5/CCP1 pin output by clearing the TRISC<5> bit.

TABLE 8-4: REGISTERS ASSOCIATED WITH CCP AND TIMER2

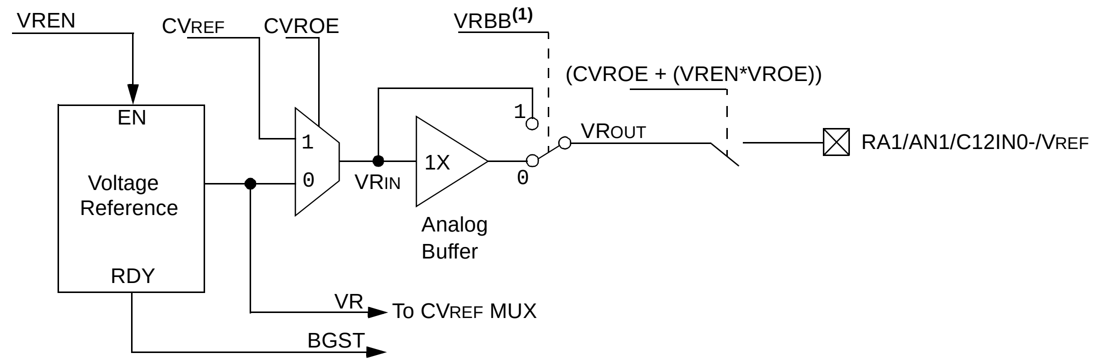
Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	uuuu uuuu
CCPR1H	Capture/Compare/PWM Register 1 High Byte								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RAIE	T0IF	INTF	RAIF	0000 0000	0000 0000
PIE1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 0000	0000 0000
PR2	Timer2 Module Period Register								1111 1111	1111 1111
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TMR2	Timer2 Module Register								0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	--11 1111	--11 1111

Legend: – = Unimplemented locations, read as ‘0’, u = unchanged, x = unknown. Shaded cells are not used by the CCP or Timer2 modules.

10.2.1 VR STABILIZATION PERIOD

When the Voltage Reference module is enabled, it will require some time for the reference and its amplifier circuits to stabilize. The user program must include a small delay routine to allow the module to settle. See **Section 19.0 "Electrical Specifications"** for the minimum delay requirement.

FIGURE 10-2: VR REFERENCE BLOCK DIAGRAM



Note 1: Buffered output requires $V_{RIN} = (V_{DD} - 1.4)V$.
2: VREN is fixed high for PIC16HV785 device.

TABLE 10-1: REGISTERS ASSOCIATED WITH COMPARATOR AND VOLTAGE REFERENCE MODULES

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
ANSEL0	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CM1CON0	C1ON	C1OUT	C1OE	C1POL	C1SP	C1R	C1CH1	C1CH0	0000 0000	0000 0000
CM2CON0	C2ON	C2OUT	C2OE	C2POL	C2SP	C2R	C2CH1	C2CH0	0000 0000	0000 0000
CM2CON1	MC1OUT	MC2OUT	—	—	—	—	T1GSS	C2SYNC	00-- --10	00-- --10
PIE1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 ---0	0000 ---0
PIR1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 ---0	0000 ---0
PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
REFCON	—	—	BGST	VRBB	VREN	VROE	CVROE	—	--00 000-	--00 000-
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
VRCON	C1VREN	C2VREN	VRR	—	VR3	VR2	VR1	VR0	000- 0000	000- 0000

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

12.1.7 CONFIGURING THE A/D

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as inputs.

To determine sample time, see Table 19-16 and Table 19-17. After this sample time has elapsed, the A/D conversion can be started.

These steps should be followed for an A/D conversion:

1. Configure the A/D module:
   - Configure analog/digital I/O (ANSx)
   - Select A/D conversion clock in the ADCON1 Register
   - Configure voltage reference in the ADCON0 Register
   - Select A/D input channel in the ADCON0 Register
   - Select result format in the ADCON0 Register
   - Turn on A/D module in the ADCON0 Register
2. Configure A/D interrupt (if desired):
   - Clear ADIF bit of the PIR1 Register
   - Set ADIE bit of the PIE1 Register
   - Set PEIE and GIE bits of the INTCON Register
3. Wait the required acquisition time.
4. Start conversion:
   - Set GO/$\overline{\text{DONE}}$ bit (ADCON0<1>)
5. Wait for A/D conversion to complete, by either:
   - Polling for the GO/$\overline{\text{DONE}}$ bit to be cleared (with interrupts disabled); OR
   - Waiting for the A/D interrupt
6. Read A/D Result register pair (ADRESH:ADRESL), clear bit ADIF if required.
7. For next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as T_{AD}. A minimum wait of 2 T_{AD} is required before the next acquisition starts.

EXAMPLE 12-1: A/D CONVERSION

```
;This code block configures the A/D
;for polling, Vdd reference, R/C clock
;and RA0 input.
;
;Conversion start and wait for complete
;polling code included.
;
 BCF    STATUS,RP1  ;Bank 1
 BSF    STATUS,RP0  ;
 MOVLW  B'01110000' ;A/D RC clock
 MOVWF  ADCON1
 BSF    TRISA,0     ;Set RA0 to input
 BSF    ANSEL0,0    ;Set RA0 to analog
 BCF    STATUS,RP0  ;Bank 0
 MOVLW  B'10000001' ;Right, Vdd Vref, AN0
 MOVWF  ADCON0
 CALL   SampleTime  ;Wait min sample time
 BSF    ADCON0,GO   ;Start conversion
 BTFSC  ADCON0,GO   ;Is conversion done?
 GOTO   $-1         ;No, test again
 MOVF   ADRESH,W    ;Read upper 2 bits
 MOVWF  RESULTHI
 BSF    STATUS,RP0  ;Bank 1
 MOVF   ADRESL,W    ;Read lower 8 bits
 BCF    STATUS,RP0  ;Bank 0
 MOVWF  RESULTLO
```

12.4 Effects of Reset

A device Reset forces all registers to their Reset state. Thus, the A/D module is turned off and any pending conversion is aborted. The ADRESH:ADRESL registers are unchanged.

12.5 Use of the CCP Trigger

An A/D conversion can be started by the “special event trigger” of the CCP module. This requires that the CCP1M3:CCP1M0 bits of the CCP1CON Register be programmed as ‘1011’ and that the A/D module is enabled (ADON bit is set). When the trigger occurs, the GO/$\overline{\text{DONE}}$ bit will be set, starting the A/D conversion and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D acquisition period with minimal software overhead (moving the ADRESH:ADRESL to the desired location).

The appropriate analog input channel must be selected and the minimum acquisition done before the “special event trigger” sets the GO/$\overline{\text{DONE}}$ bit (starts a conversion).

If the A/D module is not enabled (ADON is cleared), then the “special event trigger” will be ignored by the A/D module, but will still reset the Timer1 counter. See **Section 8.0 “Capture/Compare/PWM (CCP) Module”** for more information.

TABLE 12-3: SUMMARY OF A/D REGISTERS

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	ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	0000 0000	0000 0000
ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	—	—	-000 ----	-000 ----
ADRESH	Most Significant 8 bits of the left justified A/D result or 2 bits of the right justified result								xxxx xxxx	uuuu uuuu
ADRESL	Least Significant 2 bits of the left justified A/D result or 8 bits of the right justified result								xxxx xxxx	uuuu uuuu
ANSEL0	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
ANSEL1	—	—	—	—	ANS11	ANS10	ANS9	ANS8	---- 1111	---- 1111
INTCON	GIE	PEIE	T0IE	INTE	RAIE	T0IF	INTF	RAIF	0000 0000	0000 0000
PIE1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 0000	0000 0000
PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
PORTB	RB7	RB6	RB5	RB4	—	—	—	—	xxxx ----	uuuu ----
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	1111 ----	1111 ----
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented read as ‘0’. Shaded cells are not used for A/D module.

14.0 DATA EEPROM MEMORY

The EEPROM data memory is readable and writable during normal operation (full V_{DD} range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers. There are four SFRs used to read and write this memory:

- EECON1
- EECON2 (not a physically implemented register)
- EEDAT
- EEADR

EEDAT holds the 8-bit data for read/write, and EEADR holds the address of the EEPROM location being accessed. The PIC16F785/HV785 has 256 bytes of data EEPROM with an address range from 0h to FFh.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write). The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature, as well as from chip-to-chip. Please refer to AC Specifications in **Section 19.0 "Electrical Specifications"** for exact limits.

When the data memory is code-protected, the CPU may continue to read and write the data EEPROM memory. The device programmer can no longer access the data EEPROM data and will read zeroes.

REGISTER 14-1: EEDAT: EEPROM DATA 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
EEDAT7	EEDAT6	EEDAT5	EEDAT4	EEDAT3	EEDAT2	EEDAT1	EEDAT0
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 **EEDATn**: Byte Value to Write to or Read From Data EEPROM bits

REGISTER 14-2: EEADR: EEPROM ADDRESS 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
EEADR7	EEADR6	EEADR5	EEADR4	EEADR3	EEADR2	EEADR1	EEADR0
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 **EEADR**: Specifies one of 256 locations for EEPROM Read/Write Operation bits

TABLE 15-5: INITIALIZATION CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---1 --0x
$\overline{\text{MCLR}}$ Reset during normal operation	000h	000u uuuu	---u --uu
$\overline{\text{MCLR}}$ Reset during Sleep	000h	0001 0uuu	---u --uu
WDT Reset	000h	0000 uuuu	---u --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---u --uu
Brown-out Reset	000h	0001 1uuu	---1 --u0
Interrupt Wake-up from Sleep	PC + 1(1)	uuu1 0uuu	---u --uu

Legend: u = unchanged, x = unknown, – = unimplemented bit, reads as ‘0’.

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

BTFSC	**Bit Test f, Skip if Clear**
Syntax:	[*label*] BTFSC f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	skip if (f<b>) = `0`
Status Affected:	None
Description:	If bit ‘b’ in register ‘f’ is ‘`1`’, the next instruction is executed. If bit ‘b’, in register ‘f’, is ‘`0`’, the next instruction is discarded, and a `NOP` is executed instead, making this a two-cycle instruction.

CLRF	**Clear f**
Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register ‘f’ are cleared and the Z bit is set.

BTFSS	**Bit Test f, Skip if Set**
Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = `1`
Status Affected:	None
Description:	If bit ‘b’ in register ‘f’ is ‘`0`’, the next instruction is executed. If bit ‘b’ is ‘`1`’, then the next instruction is discarded and a `NOP` is executed instead, making this a two-cycle instruction.

CLRW	**Clear W**
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CALL	**Call Subroutine**
Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. `CALL` is a two-cycle instruction.

CLRWDT	**Clear Watchdog Timer**
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	`CLRWDT` instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

18.2 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for all PIC MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

18.3 MPLAB C18 and MPLAB C30 C Compilers

The MPLAB C18 and MPLAB C30 Code Development Systems are complete ANSI C compilers for Microchip's PIC18 and PIC24 families of microcontrollers and the dsPIC30 and dsPIC33 family of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

18.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

18.5 MPLAB ASM30 Assembler, Linker and Librarian

MPLAB ASM30 Assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire dsPIC30F instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

18.6 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C18 and MPLAB C30 C Compilers, and the MPASM and MPLAB ASM30 Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

FIGURE 20-40: TYPICAL VP6 REFERENCE VOLTAGE DISTRIBUTION (5V, 25°C)

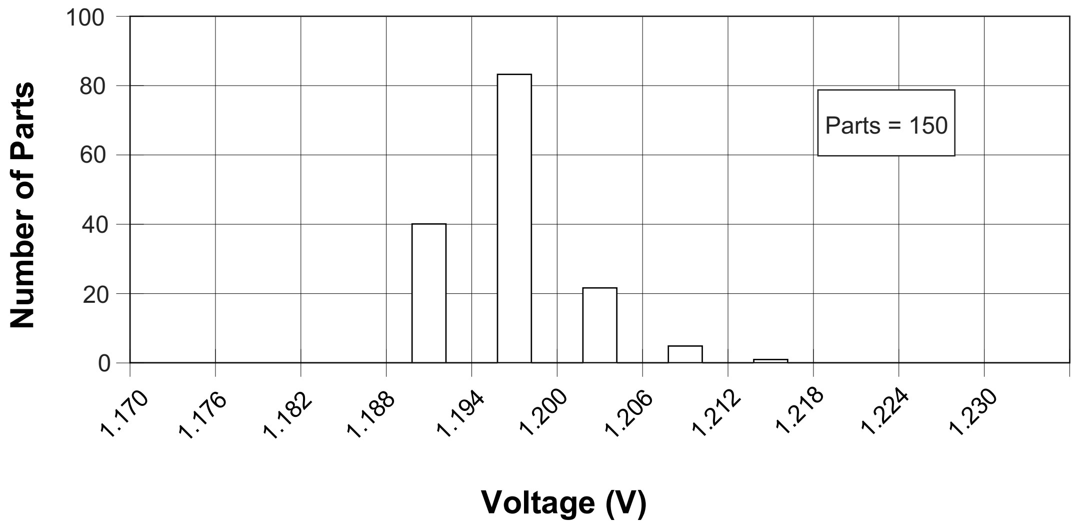


FIGURE 20-41: TYPICAL VP6 REFERENCE VOLTAGE DISTRIBUTION (5V, 85°C)

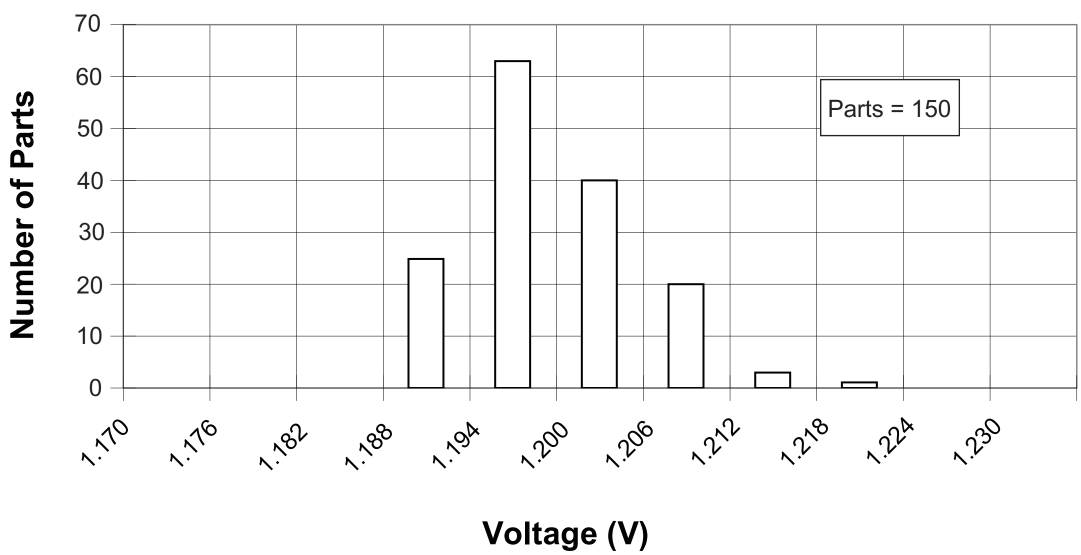


FIGURE 20-46: TYPICAL VP6 REFERENCE VOLTAGE DISTRIBUTION (3V, 125°C)

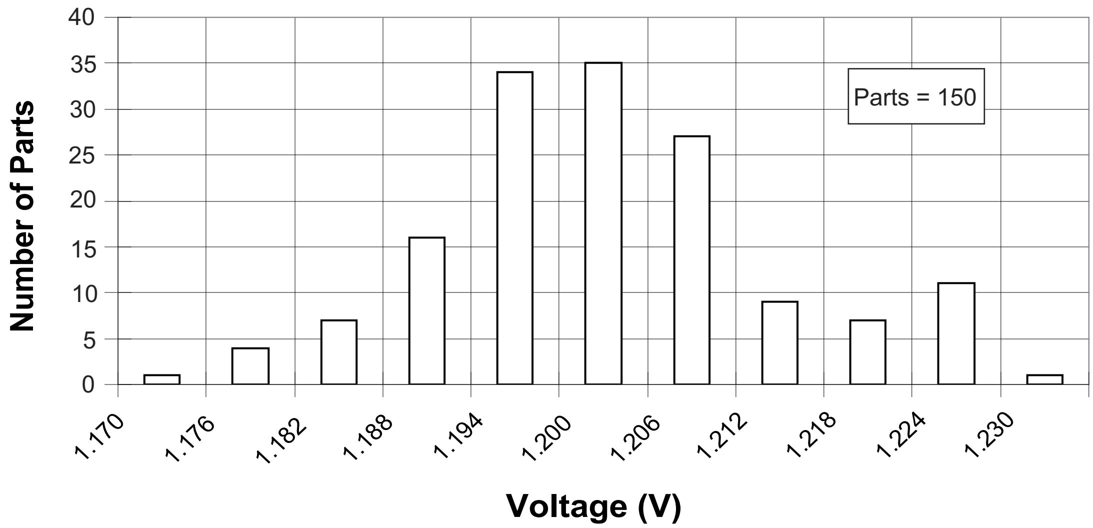
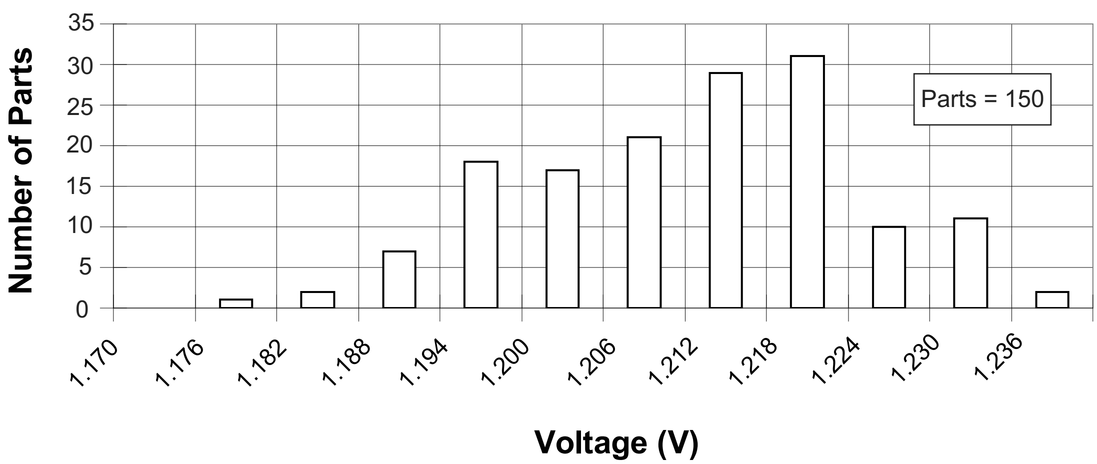


FIGURE 20-47: TYPICAL VP6 REFERENCE VOLTAGE DISTRIBUTION (3V, -40°C)



V

W

X