MICROCHIP

PIC12F629/675 Data Sheet

8-Pin, Flash-Based 8-Bit CMOS Microcontrollers

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 Deep Hardware Stack
- Direct, Indirect, and Relative Addressing modes

Special Microcontroller Features:

- Internal and External Oscillator Options
  - Precision Internal 4 MHz oscillator factory calibrated to ±1%
  - External Oscillator support for crystals and resonators
  - 5 µs wake-up from Sleep, 3.0V, typical
- Power-Saving Sleep mode
- Wide Operating Voltage Range – 2.0V to 5.5V
- Industrial and Extended Temperature Range
- Low-Power Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Detect (BOD)
- Watchdog Timer (WDT) with Independent Oscillator for Reliable Operation
- Multiplexed $\overline{\text{MCLR}}$/Input Pin
- Interrupt-on-Pin Change
- Individual Programmable Weak Pull-ups
- 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:
  - 1 nA @ 2.0V, typical
- Operating Current:
  - 8.5 µA @ 32 kHz, 2.0V, typical
  - 100 µA @ 1 MHz, 2.0V, typical
- Watchdog Timer Current
  - 300 nA @ 2.0V, typical
- Timer1 Oscillator Current:
  - 4 µA @ 32 kHz, 2.0V, typical

Peripheral Features:

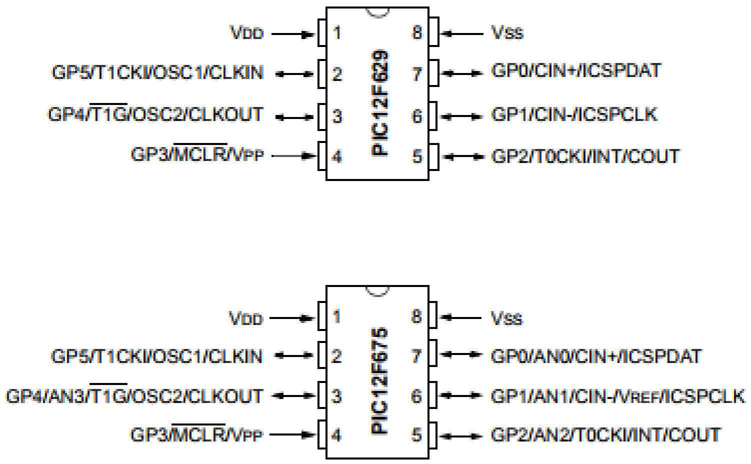
- 6 I/O Pins with Individual Direction Control
- High Current Sink/Source for Direct LED Drive
- Analog Comparator module with:
  - One analog comparator
  - Programmable on-chip comparator voltage reference (CVREF) module
  - Programmable input multiplexing from device inputs
  - Comparator output is externally accessible
- Analog-to-Digital Converter module (PIC12F675):
  - 10-bit resolution
  - Programmable 4-channel input
  - Voltage reference input
- 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
- In-Circuit Serial Programming™ (ICSP™) via two pins

Device	Program Memory	Data Memory		I/O	10-bit A/D (ch)	Comparators	Timers 8/16-bit
	Flash (words)	SRAM (bytes)	EEPROM (bytes)				
PIC12F629	1024	64	128	6	—	1	1/1
PIC12F675	1024	64	128	6	4	1	1/1

* 8-bit, 8-pin devices protected by Microchip's Low Pin Count Patent: U.S. Patent No. 5,847,450. Additional U.S. and foreign patents and applications may be issued or pending.

Pin Diagrams

8-pin PDIP, SOIC, DFN-S, DFN

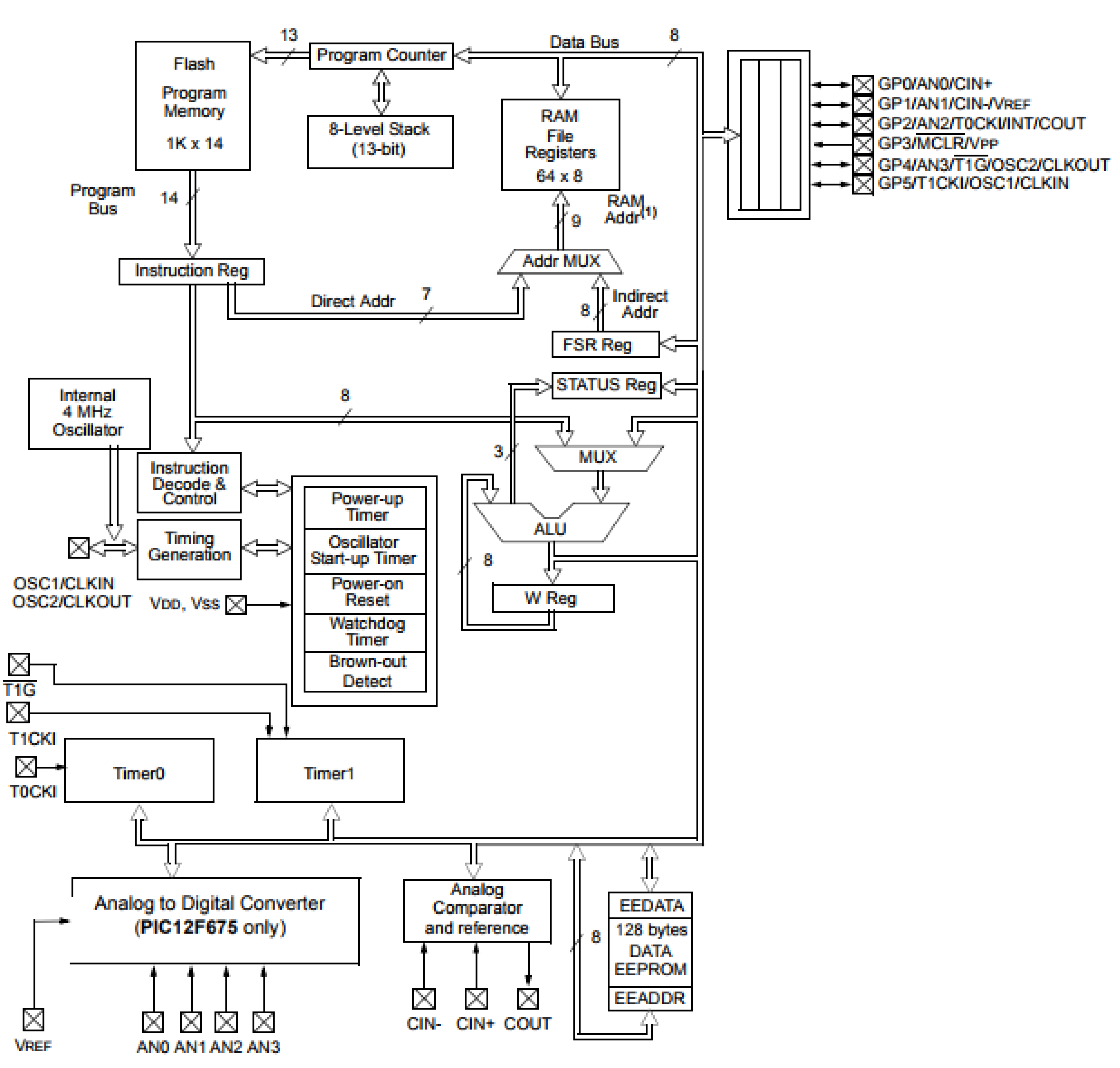


1.0 DEVICE OVERVIEW

This document contains device specific information for the PIC12F629/675. Additional information may be found in the PIC® Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this Data Sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

The PIC12F629 and PIC12F675 devices are covered by this Data Sheet. They are identical, except the PIC12F675 has a 10-bit A/D converter. They come in 8-pin PDIP, SOIC, MLF-S and DFN packages. Figure 1-1 shows a block diagram of the PIC12F629/675 devices. Table 1-1 shows the pinout description.

FIGURE 1-1: PIC12F629/675 BLOCK DIAGRAM



Note 1: Higher order bits are from STATUS register.

TABLE 1-1: PIC12F629/675 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
GP0/AN0/CIN+/ICSPDAT	GP0	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN0	AN		A/D Channel 0 input
	CIN+	AN		Comparator input
	ICSPDAT	TTL	CMOS	Serial programming I/O
GP1/AN1/CIN-/V_{REF}/ ICSPCLK	GP1	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN1	AN		A/D Channel 1 input
	CIN-	AN		Comparator input
	V_{REF}	AN		External voltage reference
	ICSPCLK	ST		Serial programming clock
GP2/AN2/T0CKI/INT/COUT	GP2	ST	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN2	AN		A/D Channel 2 input
	T0CKI	ST		TMR0 clock input
	INT	ST		External interrupt
	COUT		CMOS	Comparator output
GP3/$\overline{MCLR}$/V_{PP}	GP3	TTL		Input port w/ interrupt-on-change
	$\overline{MCLR}$	ST		Master Clear
	V_{PP}	HV		Programming voltage
GP4/AN3/$\overline{T1G}$/OSC2/ CLKOUT	GP4	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	AN3	AN		A/D Channel 3 input
	$\overline{T1G}$	ST		TMR1 gate
	OSC2		XTAL	Crystal/resonator
	CLKOUT		CMOS	$F_{OSC}/4$ output
GP5/T1CKI/OSC1/CLKIN	GP5	TTL	CMOS	Bidirectional I/O w/ programmable pull-up and interrupt-on-change
	T1CKI	ST		TMR1 clock
	OSC1	XTAL		Crystal/resonator
	CLKIN	ST		External clock input/RC oscillator connection
V_{SS}	V_{SS}	Power		Ground reference
V_{DD}	V_{DD}	Power		Positive supply

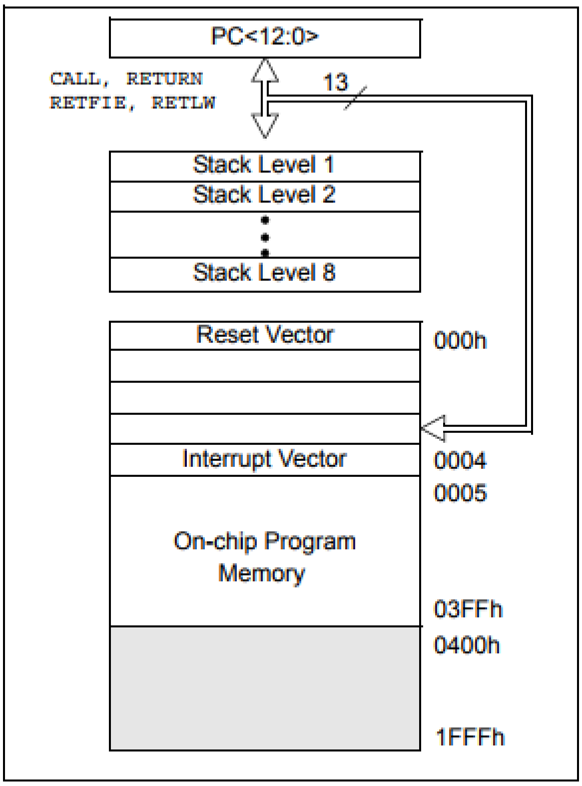
Legend: Shade = PIC12F675 only
TTL = TTL input buffer, ST = Schmitt Trigger input buffer

2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC12F629/675 devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h-03FFh) for the PIC12F629/675 devices is physically implemented. Accessing a location above these boundaries will cause a wrap-around within the first 1K x 14 space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 2-1).

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE DSTEMP/675



2.2 Data Memory Organization

The data memory (see Figure 2-2) is partitioned into two banks, which contain the General Purpose Registers and the Special Function Registers. The Special Function Registers are located in the first 32 locations of each bank. Register locations 20h-5Fh are General Purpose Registers, implemented as static RAM and are mapped across both banks. All other RAM is unimplemented and returns '0' when read. RP0 (STATUS<5>) is the bank select bit.

- RP0 = 0 Bank 0 is selected
- RP0 = 1 Bank 1 is selected

Note:	The IRP and RP1 bits STATUS<7:6> are reserved and should always be maintained as '0's.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the PIC12F629/675 devices. Each register is accessed, either directly or indirectly, through the File Select Register FSR (see **Section 2.4 "Indirect Addressing, INDF and FSR Registers"**).

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Table 2-1). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function Registers associated with the "core" are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

FIGURE 2-2: DATA MEMORY MAP OF THE PIC12F629/675

Bank 0	File Address	Bank 1	File Address
Indirect addr.(1)	00h	Indirect addr.(1)	80h
TMR0	01h	OPTION_REG	81h
PCL	02h	PCL	82h
STATUS	03h	STATUS	83h
FSR	04h	FSR	84h
GPIO	05h	TRISIO	85h
	06h		86h
	07h		87h
	08h		88h
	09h		89h
PCLATH	0Ah	PCLATH	8Ah
INTCON	0Bh	INTCON	8Bh
PIR1	0Ch	PIE1	8Ch
	0Dh		8Dh
TMR1L	0Eh	PCON	8Eh
TMR1H	0Fh		8Fh
T1CON	10h	OSCCAL	90h
	11h		91h
	12h		92h
	13h		93h
	14h		94h
	15h	WPU	95h
	16h	IOC	96h
	17h		97h
	18h		98h
CMCON	19h	VRCON	99h
	1Ah	EEDATA	9Ah
	1Bh	EEADR	9Bh
	1Ch	EECON1	9Ch
	1Dh	EECON2(1)	9Dh
ADRESH(2)	1Eh	ADRESL(2)	9Eh
ADCON0(2)	1Fh	ANSEL(2)	9Fh
General Purpose Registers 64 Bytes	20h–5Fh	accesses 20h-5Fh	A0h–DFh
	60h–7Fh		E0h–FFh

Bank 0 Bank 1

Unimplemented data memory locations, read as '0'.
1: Not a physical register.
2: PIC12F675 only.

TABLE 2-1: SPECIAL FUNCTION REGISTERS SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOD	Page
Bank 0											
00h	INDF(1)	Addressing this Location uses Contents of FSR to Address Data Memory								0000 0000	20,61
01h	TMR0	Timer0 Module's Register								xxxx xxxx	29
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	19
03h	STATUS	IRP(2)	RP1(2)	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	14
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	20
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxx	21
06h	—	Unimplemented								—	—
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter					---0 0000	19
0Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	15
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	17
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit Timer1								xxxx xxxx	32
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit Timer1								xxxx xxxx	32
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	-000 0000	35
11h	—	Unimplemented								—	—
12h	—	Unimplemented								—	—
13h	—	Unimplemented								—	—
14h	—	Unimplemented								—	—
15h	—	Unimplemented								—	—
16h	—	Unimplemented								—	—
17h	—	Unimplemented								—	—
18h	—	Unimplemented								—	—
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	38
1Ah	—	Unimplemented								—	—
1Bh	—	Unimplemented								—	—
1Ch	—	Unimplemented								—	—
1Dh	—	Unimplemented								—	—
1Eh	ADRESH(3)	Most Significant 8 bits of the Left Shifted A/D Result or 2 bits of the Right Shifted Result								xxxx xxxx	44
1Fh	ADCON0(3)	ADFM	VCFG	—	—	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	00-- 0000	45,61

Legend: — = unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: This is not a physical register.
2: These bits are reserved and should always be maintained as '0'.
3: PIC12F675 only.

TABLE 2-1: SPECIAL FUNCTION REGISTERS SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOD	Page
Bank 1											
80h	INDF[1]	Addressing this Location uses Contents of FSR to Address Data Memory								`0000 0000`	20,61
81h	OPTION_REG	$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	`1111 1111`	14,31
82h	PCL	Program Counter's (PC) Least Significant Byte								`0000 0000`	19
83h	STATUS	IRP[2]	RP1[2]	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	`0001 1xxx`	14
84h	FSR	Indirect Data Memory Address Pointer								`xxxx xxxx`	20
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	`--11 1111`	21
86h	—	Unimplemented								—	—
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah	PCLATH	—	—	—	Write Buffer for Upper 5 bits of Program Counter					`---0 0000`	19
8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	`0000 0000`	15
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	`00-- 0--0`	16
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$	`---- --0x`	18
8Fh	—	Unimplemented								—	—
90h	OSCCAL	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	`1000 00--`	18
91h	—	Unimplemented								—	—
92h	—	Unimplemented								—	—
93h	—	Unimplemented								—	—
94h	—	Unimplemented								—	—
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	`--11 -111`	21
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	`--00 0000`	23
97h	—	Unimplemented								—	—
98h	—	Unimplemented								—	—
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	`0-0- 0000`	42
9Ah	EEDATA	Data EEPROM Data Register								`0000 0000`	49
9Bh	EEADR	—	Data EEPROM Address Register							`-000 0000`	49
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	`---- x000`	50
9Dh	EECON2[1]	EEPROM Control Register 2								`---- ----`	50
9Eh	ADRESL[3]	Least Significant 2 bits of the Left Shifted A/D Result of 8 bits or the Right Shifted Result								`xxxx xxxx`	44
9Fh	ANSEL[3]	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	`-000 1111`	46,61

Legend: — = unimplemented locations read as '0', `u` = unchanged, `x` = unknown, `q` = value depends on condition, shaded = unimplemented

Note 1: This is not a physical register.
2: These bits are reserved and should always be maintained as '0'.
3: PIC12F675 only.

2.2.2.1 STATUS Register

The STATUS register, shown in Register 2-1, contains:

- the arithmetic status of the ALU
- the Reset status
- the bank select bits for data memory (SRAM)

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where `u` = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits, see the "Instruction Set Summary".

> **Note 1:** Bits IRP and RP1 (STATUS<7:6>) are not used by the PIC12F629/675 and should be maintained as clear. Use of these bits is not recommended, since this may affect upward compatibility with future products.
>
> **2:** The C and DC bits operate as a $\overline{\text{Borrow}}$ and $\overline{\text{Digit Borrow}}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

REGISTER 2-1: STATUS: STATUS REGISTER (ADDRESS: 03h OR 83h)

Reserved	Reserved	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
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 **IRP:** This bit is reserved and should be maintained as '0'

bit 6 **RP1:** This bit is reserved and should be maintained as '0'

bit 5 **RP0:** Register Bank Select bit (used for direct addressing)
0 = Bank 0 (00h - 7Fh)
1 = Bank 1 (80h - FFh)

bit 4 **$\overline{\text{TO}}$:** Time-out bit
1 = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
0 = A WDT Time-out occurred

bit 3 **$\overline{\text{PD}}$**: Power-down bit
1 = After power-up or by the `CLRWDT` instruction
0 = By execution of the `SLEEP` instruction

bit 2 **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)
For $\overline{\text{borrow}}$, the polarity is reversed.
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

bit 0 **C**: Carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note: For $\overline{\text{borrow}}$ the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high or low order bit of the source register.

2.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure:

- TMR0/WDT prescaler
- External GP2/INT interrupt
- TMR0
- Weak pull-ups on GPIO

Note: To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT by setting PSA bit to '1' (OPTION<3>). See **Section 4.4 "Prescaler"**.

REGISTER 2-2: OPTION_REG: OPTION REGISTER (ADDRESS: 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
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 **$\overline{GPPU}$:** GPIO Pull-up Enable bit
1 = GPIO pull-ups are disabled
0 = GPIO pull-ups are enabled by individual PORT latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of GP2/INT pin
0 = Interrupt on falling edge of GP2/INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
1 = Transition on GP2/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on GP2/T0CKI pin
0 = Increment on low-to-high transition on GP2/T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the TIMER0 module

bit 2-0 **PS2:PS0:** Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

2.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, GPIO port change and external GP2/INT pin interrupts.

> **Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON: INTERRUPT CONTROL REGISTER (ADDRESS: 0Bh OR 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF
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 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** GP2/INT External Interrupt Enable bit
1 = Enables the GP2/INT external interrupt
0 = Disables the GP2/INT external interrupt

bit 3 **GPIE:** Port Change Interrupt Enable bit(1)
1 = Enables the GPIO port change interrupt
0 = Disables the GPIO port change interrupt

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit(2)
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** GP2/INT External Interrupt Flag bit
1 = The GP2/INT external interrupt occurred (must be cleared in software)
0 = The GP2/INT external interrupt did not occur

bit 0 **GPIF:** Port Change Interrupt Flag bit
1 = When at least one of the GP5:GP0 pins changed state (must be cleared in software)
0 = None of the GP5:GP0 pins have changed state

Note 1: IOC register must also be enabled to enable an interrupt-on-change.
2: T0IF bit is set when TIMER0 rolls over. TIMER0 is unchanged on Reset and should be initialized before clearing T0IF bit.

2.2.2.4 PIE1 Register

The PIE1 register contains the interrupt enable bits, as shown in Register 2-4.

> **Note:** Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1 (ADDRESS: 8Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIE	ADIE	—	—	CMIE	—	—	TMR1IE
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 **EEIE:** EE Write Complete Interrupt Enable bit
1 = Enables the EE write complete interrupt
0 = Disables the EE write complete interrupt

bit 6 **ADIE:** A/D Converter Interrupt Enable bit (PIC12F675 only)
1 = Enables the A/D converter interrupt
0 = Disables the A/D converter interrupt

bit 5-4 **Unimplemented:** Read as '0'

bit 3 **CMIE:** Comparator Interrupt Enable bit
1 = Enables the comparator interrupt
0 = Disables the comparator interrupt

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **TMR1IE:** TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

2.2.2.5 PIR1 Register

The PIR1 register contains the interrupt flag bits, as shown in Register 2-5.

> **Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-5: PIR1: PERIPHERAL INTERRUPT REGISTER 1 (ADDRESS: 0Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIF	ADIF	—	—	CMIF	—	—	TMR1IF
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 **EEIF:** EEPROM Write Operation Interrupt Flag bit
1 = The write operation completed (must be cleared in software)
0 = The write operation has not completed or has not been started

bit 6 **ADIF:** A/D Converter Interrupt Flag bit (PIC12F675 only)
1 = The A/D conversion is complete (must be cleared in software)
0 = The A/D conversion is not complete

bit 5-4 **Unimplemented:** Read as ‘0’

bit 3 **CMIF**: Comparator Interrupt Flag bit
1 = Comparator input has changed (must be cleared in software)
0 = Comparator input has not changed

bit 2-1 **Unimplemented:** Read as ‘0’

bit 0 **TMR1IF**: TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

2.2.2.6 PCON Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Detect (BOD)
- Watchdog Timer Reset (WDT)
- External $\overline{\text{MCLR}}$ Reset

The PCON Register bits are shown in Register 2-6.

REGISTER 2-6: PCON: POWER CONTROL REGISTER (ADDRESS: 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$
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 7-2 **Unimplemented:** Read as ‘0’

bit 1 **$\overline{\text{POR}}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{\text{BOD}}$:** Brown-out Detect Status bit
1 = No Brown-out Detect occurred
0 = A Brown-out Detect occurred (must be set in software after a Brown-out Detect occurs)

2.2.2.7 OSCCAL Register

The Oscillator Calibration register (OSCCAL) is used to calibrate the internal 4 MHz oscillator. It contains 6 bits to adjust the frequency up or down to achieve 4 MHz.

The OSCCAL register bits are shown in Register 2-7.

REGISTER 2-7: OSCCAL: OSCILLATOR CALIBRATION REGISTER (ADDRESS: 90h)

R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—
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

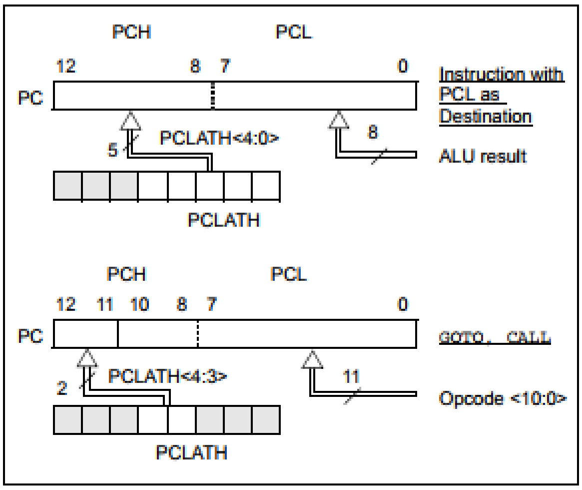
bit 7-2 **CAL5:CAL0:** 6-bit Signed Oscillator Calibration bits
`111111` = Maximum frequency
`100000` = Center frequency
`000000` = Minimum frequency

bit 1-0 **Unimplemented:** Read as '0'

2.3 PCL and PCLATH

The Program Counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-3 shows the two situations for the loading of the PC. The upper example in Figure 2-3 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-3 shows how the PC is loaded during a `CALL` or `GOTO` instruction (PCLATH<4:3> → PCH).

FIGURE 2-3: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 COMPUTED GOTO

A computed `GOTO` is accomplished by adding an offset to the PC (`ADDWF PCL`). When performing a table read using a computed `GOTO` method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note, *"Implementing a Table Read"* (AN556).

2.3.2 STACK

The PIC12F629/675 family has an 8-level deep x 13-bit wide hardware stack (see Figure 2-1). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a `CALL` instruction is executed, or an interrupt causes a branch. The stack is POPed in the event of a `RETURN`, `RETLW` or a `RETFIE` instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1: There are no Status bits to indicate Stack Overflow or Stack Underflow conditions.

2: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the `CALL`, `RETURN`, `RETLW` and `RETFIE` instructions, or the vectoring to an interrupt address.

2.4 Indirect Addressing, INDF and FSR Registers

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

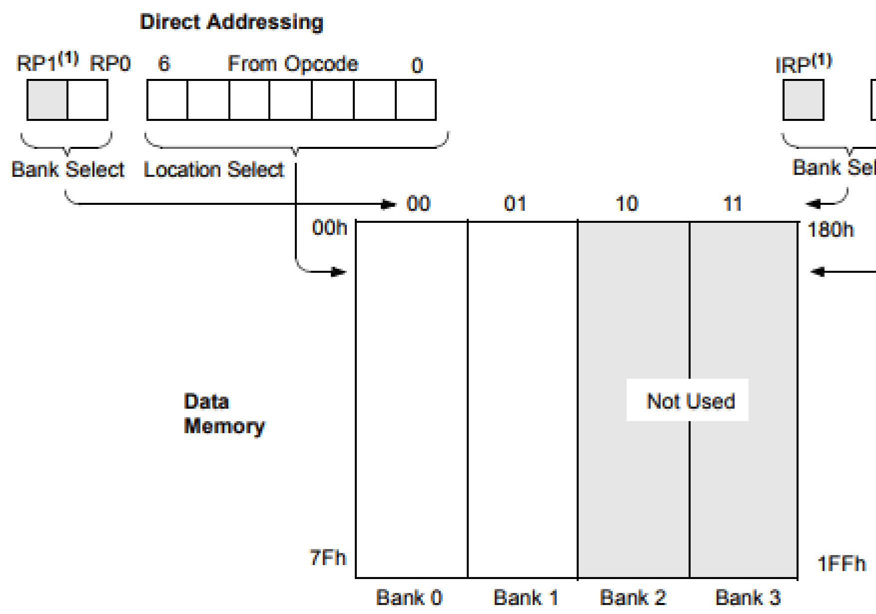
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-2.

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     ;inc pointer
          BTFSS   FSR,4   ;all done?
          GOTO    NEXT    ;no clear next
CONTINUE                  ;yes continue
```

FIGURE 2-2: DIRECT/INDIRECT ADDRESSING PIC12F629/675



For memory map detail see Figure 2-2.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

3.0 GPIO PORT

There are as many as six general purpose I/O pins available. Depending on which peripherals are enabled, some or all of the pins may not be available as general purpose I/O. In general, when a peripheral is enabled, the associated pin may not be used as a general purpose I/O pin.

Note:	Additional information on I/O ports may be found in the PIC® Mid-Range Reference Manual, (DS33023).

3.1 GPIO and the TRISIO Registers

GPIO is an 6-bit wide, bidirectional port. The corresponding data direction register is TRISIO. Setting a TRISIO bit (= 1) will make the corresponding GPIO pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISIO bit (= 0) will make the corresponding GPIO pin an output (i.e., put the contents of the output latch on the selected pin). The exception is GP3, which is input-only and its TRISIO bit will always read as '1'. Example 3-1 shows how to initialize GPIO.

Reading the GPIO register 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. GP3 reads '0' when MCLREN = 1.

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

Note:	The ANSEL (9Fh) and CMCON (19h) registers (9Fh) must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

EXAMPLE 3-1: INITIALIZING GPIO

```
BCF     STATUS,RP0    ;Bank 0
CLRF    GPIO          ;Init GPIO
MOVLW   07h           ;Set GP<2:0> to
MOVWF   CMCON         ;digital IO
BSF     STATUS,RP0    ;Bank 1
CLRF    ANSEL         ;Digital I/O
MOVLW   0Ch           ;Set GP<3:2> as inputs
MOVWF   TRISIO        ;and set GP<5:4,1:0>
                      ;as outputs
```

3.2 Additional Pin Functions

Every GPIO pin on the PIC12F629/675 has an interrupt-on-change option and every GPIO pin, except GP3, has a weak pull-up option. The next two sections describe these functions.

3.2.1 WEAK PULL-UP

Each of the GPIO pins, except GP3, has an individually configurable weak internal pull-up. Control bits WPUx enable or disable each pull-up. Refer to Register 3-3. Each weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset by the $\overline{\text{GPPU}}$ bit (OPTION<7>).

REGISTER 3-1: GPIO: GPIO REGISTER (ADDRESS: 05h)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
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 5-0 **GPIO<5:0>**: General Purpose I/O pin
1 = Port pin is >V_{IH}
0 = Port pin is <V_{IL}

REGISTER 3-2: TRISIO: GPIO TRI-STATE REGISTER (ADDRESS: 85h)

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0
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 5-0 **TRISIO<5:0>**: General Purpose I/O Tri-State Control bit
1 = GPIO pin configured as an input (tri-stated)
0 = GPIO pin configured as an output

Note: TRISIO<3> always reads '1'.

REGISTER 3-3: WPU: WEAK PULL-UP REGISTER (ADDRESS: 95h)

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0
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 **WPU<5:4>:** Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

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

bit 2-0 **WPU<2:0>:** Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: Global $\overline{\text{GPPU}}$ must be enabled for individual pull-ups to be enabled.

2: The weak pull-up device is automatically disabled if the pin is in Output mode (TRISIO = 0).

3.2.2 INTERRUPT-ON-CHANGE

Each of the GPIO pins is individually configurable as an interrupt-on-change pin. Control bits IOC enable or disable the interrupt function for each pin. Refer to Register 3-4. The interrupt-on-change is disabled on a Power-on Reset.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of GPIO. The 'mismatch' outputs of the last read are OR'd together to set, the GP Port Change Interrupt flag bit (GPIF) in the INTCON register.

This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

a) Any read or write of GPIO. This will end the mismatch condition.
b) Clear the flag bit GPIF.

A mismatch condition will continue to set flag bit GPIF. Reading GPIO will end the mismatch condition and allow flag bit GPIF to be cleared.

Note: If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the GPIF interrupt flag may not get set.

REGISTER 3-4: IOC: INTERRUPT-ON-CHANGE GPIO REGISTER (ADDRESS: 96h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
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 5-0 **IOC<5:0>:** Interrupt-on-Change GPIO Control bits
1 = Interrupt-on-change enabled
0 = Interrupt-on-change disabled

Note 1: Global Interrupt Enable (GIE) must be enabled for individual interrupts to be recognized.

3.3 Pin Descriptions and Diagrams

Each GPIO 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.

3.3.1 GP0/AN0/CIN+

Figure 3-1 shows the diagram for this pin. The GP0 pin is configurable to function as one of the following:

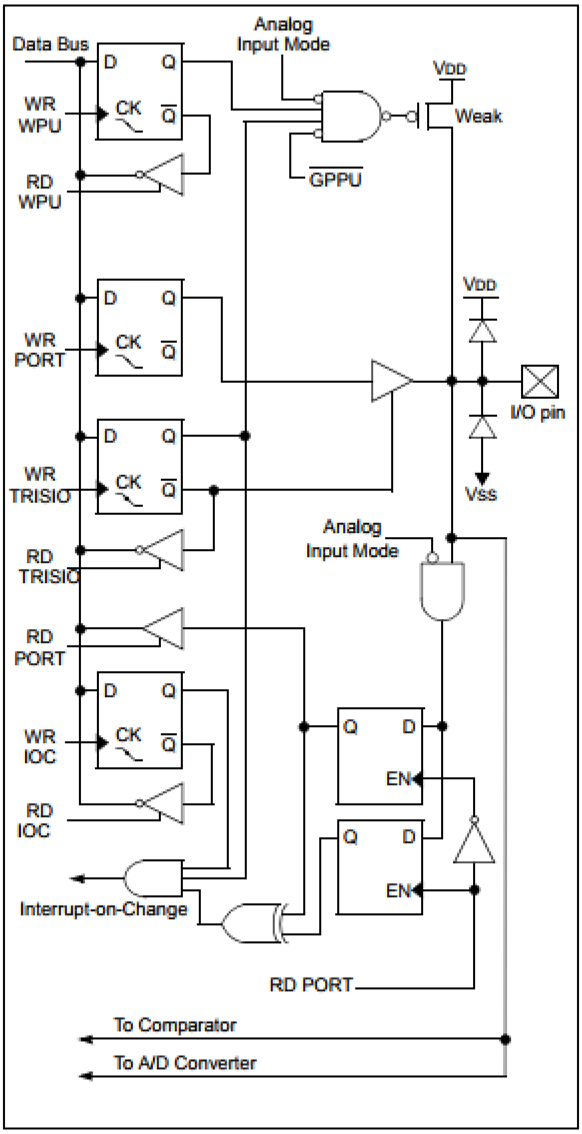
- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- an analog input to the comparator

3.3.2 GP1/AN1/CIN-/VREF

Figure 3-1 shows the diagram for this pin. The GP1 pin is configurable to function as one of the following:

- as a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- an analog input to the comparator
- a voltage reference input for the A/D (PIC12F675 only)

FIGURE 3-1: BLOCK DIAGRAM OF GP0 AND GP1 PINS



3.3.3 GP2/AN2/T0CKI/INT/COUT

Figure 3-2 shows the diagram for this pin. The GP2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)

3.3.4 GP3/MCLR/VPP

Figure 3-3 shows the diagram for this pin. The GP3 pin is configurable to function as one of the following:

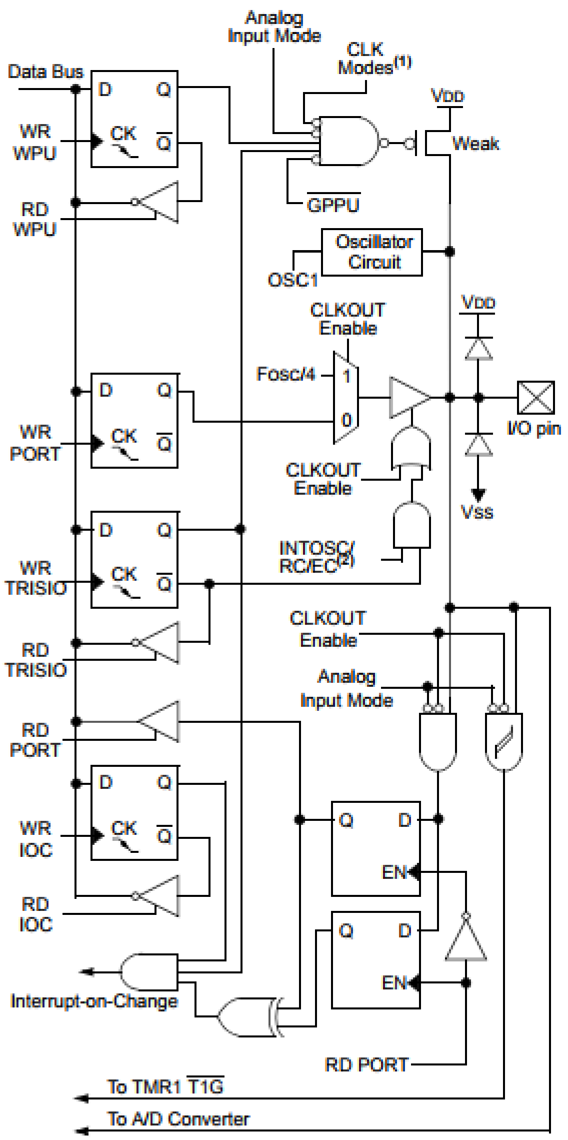
- a general purpose input
- as Master Clear Reset

3.3.5 GP4/AN3/$\overline{\text{T1G}}$/OSC2/CLKOUT

Figure 3-4 shows the diagram for this pin. The GP4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the A/D (PIC12F675 only)
- a TMR1 gate input
- a crystal/resonator connection
- a clock output

FIGURE 3-4: BLOCK DIAGRAM OF GP4



Note 1: CLK modes are XT, HS, LP, LPTMR1 and CLKOUT Enable.

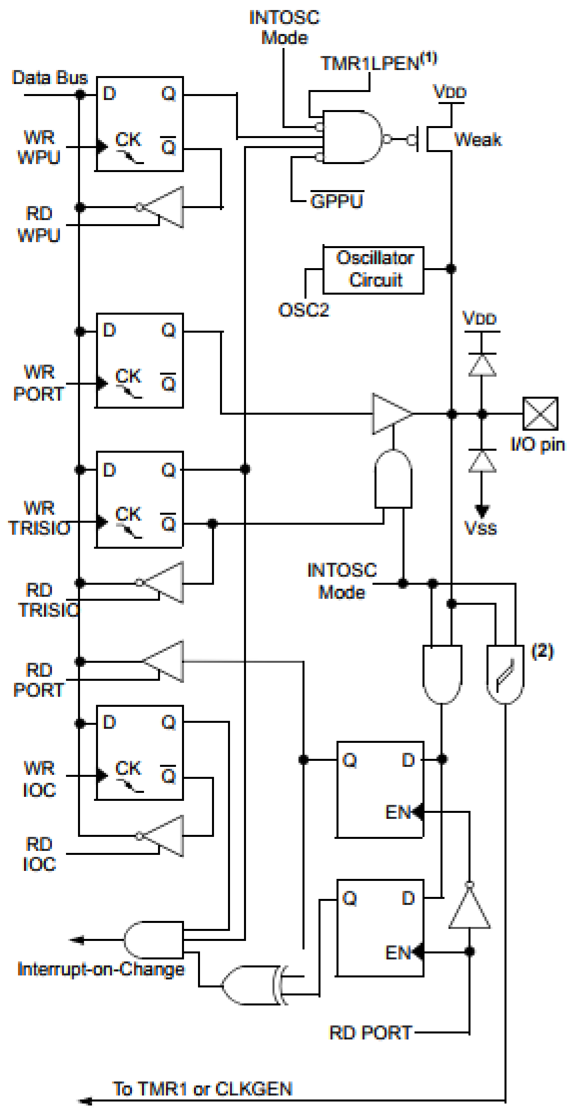
2: With CLKOUT option.

3.3.6 GP5/T1CKI/OSC1/CLKIN

Figure 3-5 shows the diagram for this pin. The GP5 pin is configurable to function as one of the following:

- a general purpose I/O
- a TMR1 clock input
- a crystal/resonator connection
- a clock input

FIGURE 3-5: BLOCK DIAGRAM OF GP5



Note 1: Timer1 LP Oscillator enabled

2: When using Timer1 with LP oscillator, the Schmitt Trigger is by-passed.

TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH GPIO

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOD	Value on all other Resets
05h	GPIO	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--xx xxxx	--uu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	--11 -111
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	--00 0000	--00 0000
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	-000 1111

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

4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 4-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

> **Note:** Additional information on the Timer0 module is available in the PIC® Mid-Range Reference Manual, (DS33023).

4.1 Timer0 Operation

Timer mode is selected by clearing the T0CS bit (OPTION_REG<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

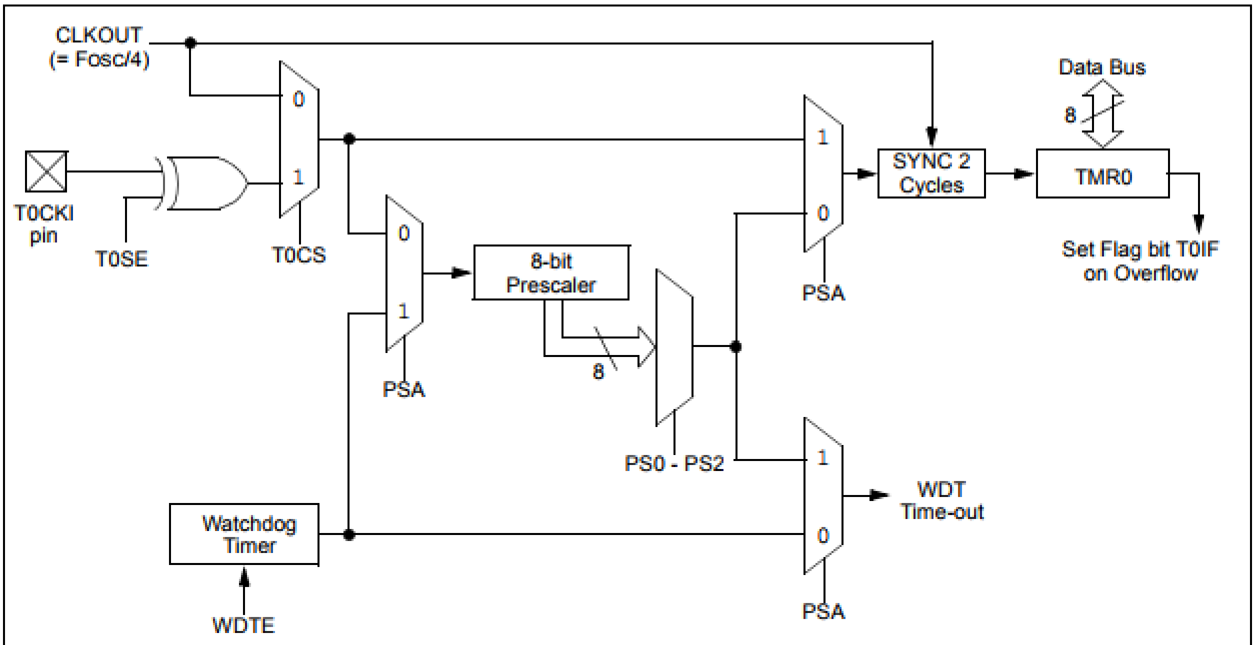
Counter mode is selected by setting the T0CS bit (OPTION_REG<5>). In this mode, the Timer0 module will increment either on every rising or falling edge of pin GP2/T0CKI. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION_REG<4>). Clearing the T0SE bit selects the rising edge.

> **Note:** Counter mode has specific external clock requirements. Additional information on these requirements is available in the PIC® Mid-Range Reference Manual, (DS33023).

4.2 Timer0 Interrupt

A Timer0 interrupt is generated when the TMR0 register timer/counter overflows from FFh to 00h. This overflow sets the T0IF bit. The interrupt can be masked by clearing the T0IE bit (INTCON<5>). The T0IF bit (INTCON<2>) must be cleared in software by the Timer0 module Interrupt Service Routine before re-enabling this interrupt. The Timer0 interrupt cannot wake the processor from Sleep since the timer is shut-off during Sleep.

FIGURE 4-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the OPTION register.

4.3 Using Timer0 with an External Clock

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI, with the internal phase clocks, is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least $2T_{OSC}$ (and a small RC delay of 20 ns) and low for at least $2T_{OSC}$ (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

> **Note:** The ANSEL (9Fh) and CMCON (19h) registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. The ANSEL register is defined for the PIC12F675.

REGISTER 4-1: OPTION_REG: OPTION REGISTER (ADDRESS: 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
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 **$\overline{\text{GPPU}}$:** GPIO Pull-up Enable bit
- 1 = GPIO pull-ups are disabled
- 0 = GPIO pull-ups are enabled by individual PORT latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
- 1 = Interrupt on rising edge of GP2/INT pin
- 0 = Interrupt on falling edge of GP2/INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
- 1 = Transition on GP2/T0CK pin
- 0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE:** TMR0 Source Edge Select bit
- 1 = Increment on high-to-low transition on GP2/T0CKI pin
- 0 = Increment on low-to-high transition on GP2/T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
- 1 = Prescaler is assigned to the WDT
- 0 = Prescaler is assigned to the TIMER0 module

bit 2-0 **PS2:PS0:** Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

4.4 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer. For simplicity, this counter will be referred to as "prescaler" throughout this Data Sheet. The prescaler assignment is controlled in software by the control bit PSA (OPTION_REG<3>). Clearing the PSA bit will assign the prescaler to Timer0. Prescale values are selectable via the PS2:PS0 bits (OPTION_REG<2:0>).

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF 1, MOVWF 1, BSF 1, x....etc.`) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer.

4.4.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control (i.e., it can be changed "on the fly" during program execution). To avoid an unintended device Reset, the following instruction sequence (Example 4-1) must be executed when changing the prescaler assignment from Timer0 to WDT.

EXAMPLE 4-1: CHANGING PRESCALER (TIMER0→WDT)

```
BCF     STATUS,RP0    ;Bank 0
CLRWDT                ;Clear WDT
CLRF    TMR0          ;Clear TMR0 and
                      ; prescaler
BSF     STATUS,RP0    ;Bank 1

MOVLW   b'00101111'   ;Required if desired
MOVWF   OPTION_REG    ; PS2:PS0 is
CLRWDT                ; 000 or 001
                      ;
MOVLW   b'00101xxx'   ;Set postscaler to
MOVWF   OPTION_REG    ; desired WDT rate
BCF     STATUS,RP0    ;Bank 0
```

To change prescaler from the WDT to the TMR0 module, use the sequence shown in Example 4-2. This precaution must be taken even if the WDT is disabled.

EXAMPLE 4-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT                ;Clear WDT and
                      ; postscaler
BSF     STATUS,RP0    ;Bank 1

MOVLW   b'xxxx0xxx'   ;Select TMR0,
                      ; prescale, and
                      ; clock source
MOVWF   OPTION_REG    ;
BCF     STATUS,RP0    ;Bank 0
```

TABLE 4-1: REGISTERS ASSOCIATED WITH TIMER0

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOD	Value on all other Resets
01h	TMR0	Timer0 Module Register								xxxx xxxx	uuuu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
81h	OPTION_REG	$\overline{\text{GPPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111

Legend: — = Unimplemented locations, read as '0', u = unchanged, x = unknown.
Shaded cells are not used by the Timer0 module.