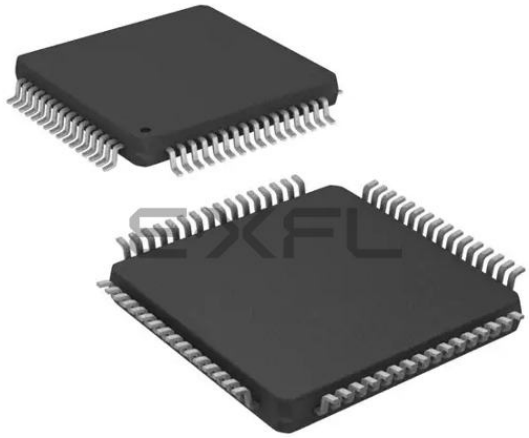


Welcome to E-XFL.COM

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	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	59
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 45x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f19195t-i-pt

1.1 Register and Bit Naming Conventions

1.1.1 REGISTER NAMES

When there are multiple instances of the same peripheral in a device, the peripheral control registers will be depicted as the concatenation of a peripheral identifier, peripheral instance, and control identifier. The control registers section will show just one instance of all the register names with an 'x' in the place of the peripheral instance number. This naming convention may also be applied to peripherals when there is only one instance of that peripheral in the device to maintain compatibility with other devices in the family that contain more than one.

1.1.2 BIT NAMES

There are two variants for bit names:

- Short name: Bit function abbreviation
- Long name: Peripheral abbreviation + short name

1.1.2.1 Short Bit Names

Short bit names are an abbreviation for the bit function. For example, some peripherals are enabled with the EN bit. The bit names shown in the registers are the short name variant.

Short bit names are useful when accessing bits in C programs. The general format for accessing bits by the short name is *RegisterName*bits.*ShortName*. For example, the enable bit, EN, in the COG1CON0 register can be set in C programs with the instruction `COG1CON0bits.EN = 1`.

Short names are generally not useful in assembly programs because the same name may be used by different peripherals in different bit positions. When this occurs, during the include file generation, all instances of that short bit name are appended with an underscore plus the name of the register in which the bit resides to avoid naming contentions.

1.1.2.2 Long Bit Names

Long bit names are constructed by adding a peripheral abbreviation prefix to the short name. The prefix is unique to the peripheral, thereby making every long bit name unique. The long bit name for the COG1 enable bit is the COG1 prefix, G1, appended with the enable bit short name, EN, resulting in the unique bit name G1EN.

Long bit names are useful in both C and assembly programs. For example, in C the COG1CON0 enable bit can be set with the `G1EN = 1` instruction. In assembly, this bit can be set with the `BSF COG1CON0,G1EN` instruction.

1.1.2.3 Bit Fields

Bit fields are two or more adjacent bits in the same register. Bit fields adhere only to the short bit naming convention. For example, the three Least Significant bits of the COG1CON0 register contain the mode control bits. The short name for this field is MD. There is no long bit name variant. Bit field access is only possible in C programs. The following example demonstrates a C program instruction for setting the COG1 to the Push-Pull mode:

```
COG1CON0bits.MD = 0x5;
```

Individual bits in a bit field can also be accessed with long and short bit names. Each bit is the field name appended with the number of the bit position within the field. For example, the Most Significant mode bit has the short bit name MD2 and the long bit name is G1MD2. The following two examples demonstrate assembly program sequences for setting the COG1 to Push-Pull mode:

Example 1:

```
MOVLW  ~(1<<G1MD1)
ANDWF  COG1CON0,F
MOVLW  1<<G1MD2 | 1<<G1MD0
IORWF  COG1CON0,F
```

Example 2:

```
BSF    COG1CON0,G1MD2
BCF    COG1CON0,G1MD1
BSF    COG1CON0,G1MD0
```

1.1.3 REGISTER AND BIT NAMING EXCEPTIONS

1.1.3.1 Status, Interrupt, and Mirror Bits

Status, interrupt enables, interrupt flags, and mirror bits are contained in registers that span more than one peripheral. In these cases, the bit name shown is unique so there is no prefix or short name variant.

1.1.3.2 Legacy Peripherals

There are some peripherals that do not strictly adhere to these naming conventions. Peripherals that have existed for many years and are present in almost every device are the exceptions. These exceptions were necessary to limit the adverse impact of the new conventions on legacy code. Peripherals that do adhere to the new convention will include a table in the registers section indicating the long name prefix for each peripheral instance. Peripherals that fall into the exception category will not have this table. These peripherals include, but are not limited to, the following:

- EUSART
- MSSP

4.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- Program Memory
  - Configuration Words
  - Device ID
  - User ID
  - Program Flash Memory
  - Device Information Area (DIA)
  - Device Configuration Information (DCI)
  - Revision ID
- Data Memory
  - Core Registers
  - Special Function Registers
  - General Purpose RAM (GPR)
  - Common RAM
- Data EEPROM

The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing
- NVMREG access

4.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing 32K x 14 program memory space. Table 4-1 shows the memory sizes implemented. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 4-1).

TABLE 4-1: DEVICE SIZES AND ADDRESSES

Device	Program Flash Memory Size (Words)	Last Program Memory Address
PIC16(L)F19195	8k	1FFFh
PIC16(L)F19196	16k	3FFFh
PIC16(L)F19197	32k	7FFFh

TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19195/6/7 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 7											
CPU CORE REGISTERS; see Table 4-3 for specifics											
38Ch	—	Unimplemented								---- ----	---- ----
38Dh	—	Unimplemented								---- ----	---- ----
38Eh	—	Unimplemented								---- ----	---- ----
38Fh	—	Unimplemented								---- ----	---- ----
390h	—	Unimplemented								---- ----	---- ----
391h	—	Unimplemented								---- ----	---- ----
392h	—	Unimplemented								---- ----	---- ----
393h	—	Unimplemented								---- ----	---- ----
394h	—	Unimplemented								---- ----	---- ----
395h	—	Unimplemented								---- ----	---- ----
396h	—	Unimplemented								---- ----	---- ----
397h	—	Unimplemented								---- ----	---- ----
398h	—	Unimplemented								---- ----	---- ----
399h	—	Unimplemented								---- ----	---- ----
39Ah	—	Unimplemented								---- ----	---- ----
39Bh	—	Unimplemented								---- ----	---- ----
39Ch	—	Unimplemented								---- ----	---- ----
39Dh	—	Unimplemented								---- ----	---- ----
39Eh	—	Unimplemented								---- ----	---- ----
39Fh	—	Unimplemented								---- ----	---- ----

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

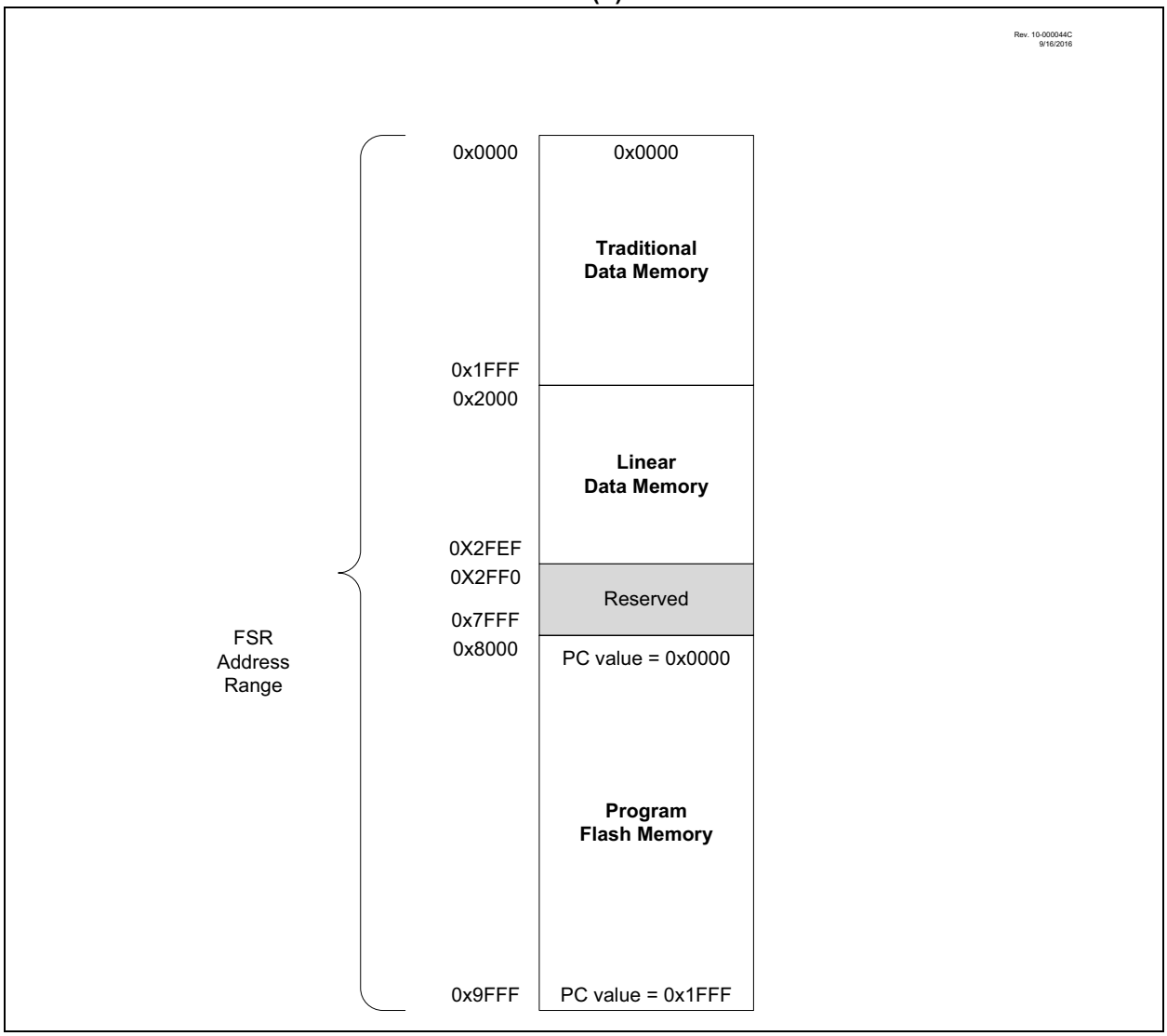
TABLE 4-12: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 PIC16(L)F19195/6/7 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 25											
CPU CORE REGISTERS; see Table 4-3 for specifics											
C8Ch	PORTG	RG7	RG6	RG5	RG4	RG3	RG2	RG1	RG0	xxxx xxxx	uuuu uuuu
C8Dh	PORTH	—	—	—	—	RH3	RH2	RH1	RH0	xxxx xxxx	uuuu uuuu
C8Eh	TRISG	TRISG7	TRISG6	—	TRISG4	TRISG3	TRISG2	TRISG1	TRISG0	1111 1111	1111 1111
C8Fh	TRISH	—	—	—	—	TRISH3	TRISH2	TRISH1	TRISH0	1111 1111	1111 1111
C90h	LATG	LATG7	LATG6	—	LATG4	LATG3	LATG2	LATG1	LATG0	1111 1111	1111 1111
C91h	LATH	—	—	—	—	LATH3	LATH2	LATH1	LATH0	1111 1111	1111 1111
C92h	—	Unimplemented								---- ----	---- ----
C93h	—	Unimplemented								---- ----	---- ----
C94h	—	Unimplemented								---- ----	---- ----
C95h	—	Unimplemented								---- ----	---- ----
C96h	—	Unimplemented								---- ----	---- ----
C97h	—	Unimplemented								---- ----	---- ----
C98h	—	Unimplemented								---- ----	---- ----
C99h	—	Unimplemented								---- ----	---- ----
C9Ah	—	Unimplemented								---- ----	---- ----
C9Bh	—	Unimplemented								---- ----	---- ----
C9Ch	—	Unimplemented								---- ----	---- ----
C9Dh	—	Unimplemented								---- ----	---- ----
C9Eh	—	Unimplemented								---- ----	---- ----
C9Fh	—	Unimplemented								---- ----	---- ----

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

FIGURE 4-8: INDIRECT ADDRESSING PIC16(L)F19195



REGISTER 9-5: OSCEN: OSCILLATOR MANUAL ENABLE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0
EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	—	—
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **EXTOEN:** External Oscillator Manual Request Enable bit
1 = EXTOSC is explicitly enabled, operating as specified by FEXTOSC
0 = EXTOSC could be enabled by another module

bit 6 **HFOEN:** HFINTOSC Oscillator Manual Request Enable bit
1 = HFINTOSC is explicitly enabled, operating as specified by OSCFRQ
0 = HFINTOSC could be enabled by another module

bit 5 **MFOEN:** MFINTOSC Oscillator Manual Request Enable bit
1 = MFINTOSC is explicitly enabled
0 = MFINTOSC could be enabled by another module

bit 4 **LFOEN:** LFINTOSC (31 kHz) Oscillator Manual Request Enable bit
1 = LFINTOSC is explicitly enabled
0 = LFINTOSC could be enabled by another module

bit 3 **SOSCEN:** Secondary (Timer1) Oscillator Manual Request bit
1 = Secondary oscillator is explicitly enabled, operating as specified by SOSCPWR
0 = Secondary oscillator could be enabled by another module

bit 2 **ADOEN:** FRC Oscillator Manual Request Enable bit
1 = FRC is explicitly enabled
0 = FRC could be enabled by another module

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

14.9 Register Definitions: PORTD

REGISTER 14-24: PORTD: PORTD REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **RD<7:0>**: PORTD I/O Value bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTD are actually written to corresponding LATD register. Reads from PORTD register is return of actual I/O pin values.

REGISTER 14-25: TRISD: PORTD TRI-STATE REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **TRISD<7:0>:** PORTD Tri-State Control bit
1 = PORTD pin configured as an input (tri-stated)
0 = PORTD pin configured as an output

14.11 Register Definitions: PORTE

REGISTER 14-32: PORTE: PORTE REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u
RE7	RE6	RE5	RE4	RE3	—	RE1	RE0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-3 **RE<7:3>**: PORTE I/O Value bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

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

bit 1-0 **RE<1:0>**: PORTE I/O Value bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTE are actually written to corresponding LATE register. Reads from PORTE register is return of actual I/O pin values.

REGISTER 14-33: TRISE: PORTE TRI-STATE REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1
TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	—	TRISE1	TRISE0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-3 **TRISE<7:3>:** PORTE Tri-State Control bit
1 = PORTE pin configured as an input (tri-stated)
0 = PORTE pin configured as an output

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

bit 1-0 **TRISE<1:0>:** PORTE Tri-State Control bit
1 = PORTE pin configured as an input (tri-stated)
0 = PORTE pin configured as an output

REGISTER 14-45: ODCONF: PORTF OPEN-DRAIN CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ODCF7	ODCF6	ODCF5	ODCF4	ODCF3	ODCF2	ODCF1	ODCF0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **ODCF<7:0>:** PORTF Open-Drain Enable bits
For RF<7:0> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

REGISTER 14-46: SLRCONF: PORTF SLEW RATE CONTROL REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
SLRF7	SLRF6	SLRF5	SLRF4	SLRF3	SLRF2	SLRF1	SLRF0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **SLRF<7:0>:** PORTF Slew Rate Enable bits
For RF<7:0> pins, respectively
1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

REGISTER 14-47: INLVLF: PORTF INPUT LEVEL CONTROL REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
INLVLF7	INLVLF6	INLVLF5	INLVLF4	INLVLF3	INLVLF2	INLVLF1	INLVLF0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **INLVLF<7:0>:** PORTF Input Level Select bits
For RF<7:0> pins, respectively
1 = ST input used for PORT reads and interrupt-on-change
0 = TTL input used for PORT reads and interrupt-on-change

REGISTER 17-12: IOCGF: INTERRUPT-ON-CHANGE PORTG FLAG REGISTER

U-0	U-0	R/W/HS-0/0	U-0	U-0	U-0	U-0	U-0
—	—	IOCGF5	—	—	—	—	—
bit 7							bit 0

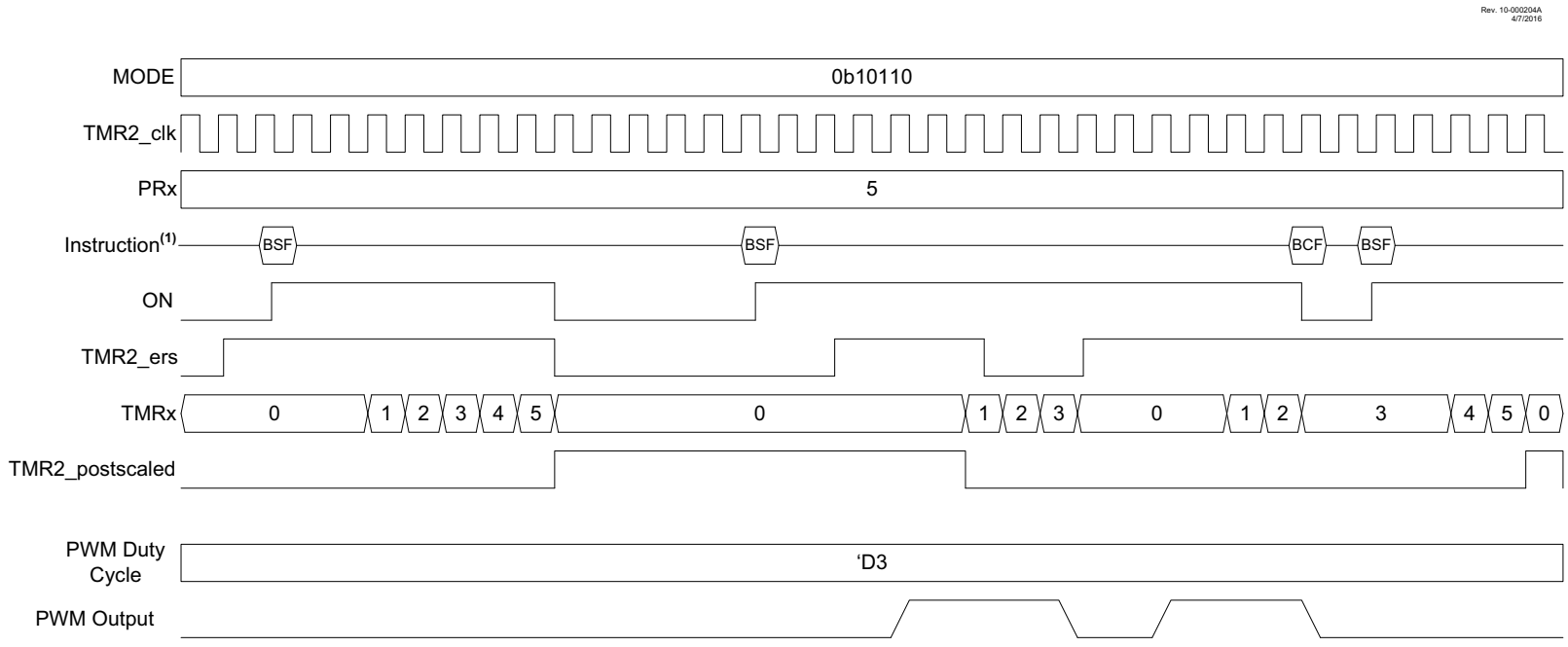
Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HS - Bit is set in hardware

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **IOCGF5:** Interrupt-on-Change PORTG Flag bit
1 = An enabled change was detected on the associated pin
Set when IOCGPx = 1 and a rising edge was detected on RGx, or when IOCGNx = 1 and a falling edge was detected on RGx.
0 = No change was detected, or the user cleared the detected change

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

FIGURE 27-13: LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODE TIMING DIAGRAM (MODE = 10110)



Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

28.7.8 CAPTURE MODE

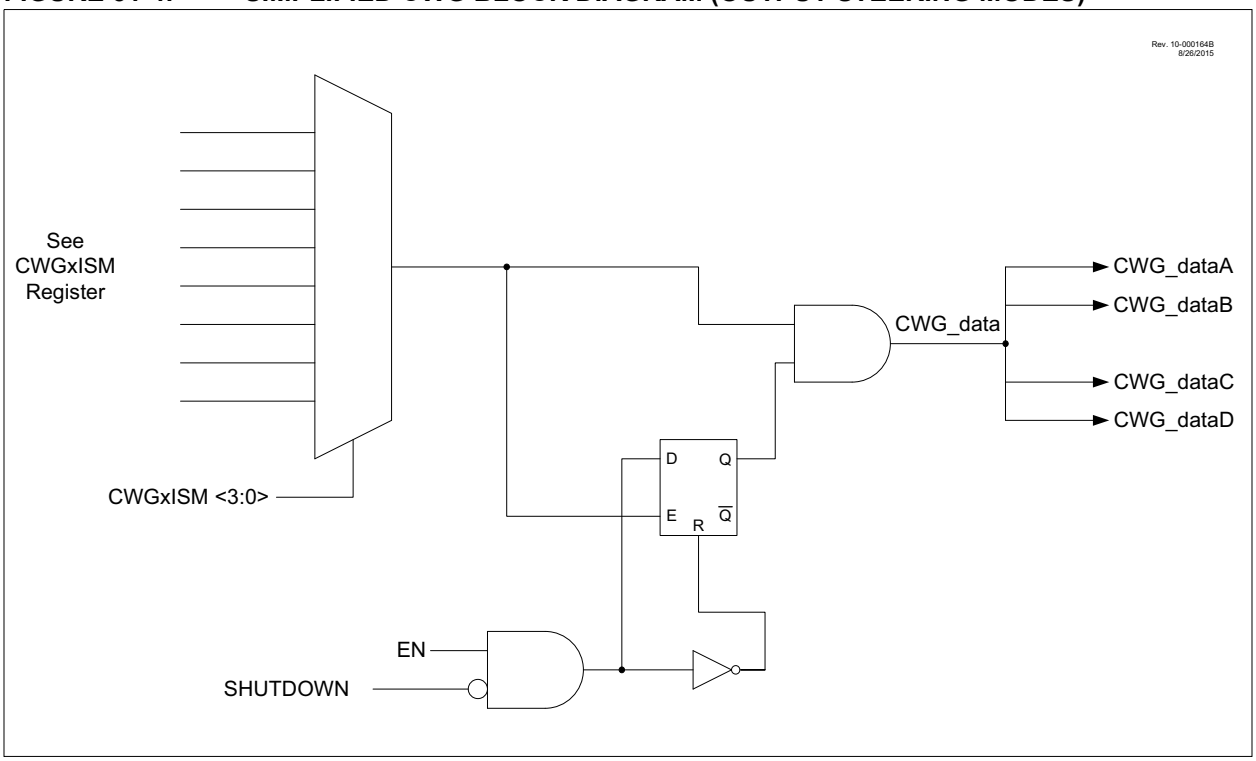
Capture mode captures the Timer value based on a rising or falling edge on the SMTWINx input and triggers an interrupt. This mimics the capture feature of a CCP module. The timer begins incrementing upon the SMTxGO bit being set, and updates the value of the SMTxCPR register on each rising edge of SMTWINx, and updates the value of the CPW register on each falling edge of the SMTWINx. The timer is not reset by any hardware conditions in this mode and must be reset by software, if desired. See Figure 28-16 and Figure 28-17.

31.1.4 STEERING MODES

In Steering modes, the data input can be steered to any or all of the four CWG output pins. In Synchronous Steering mode, changes to steering selection registers take effect on the next rising input.

In Non-Synchronous mode, steering takes effect on the next instruction cycle. Additional details are provided in **Section 31.9 "CWG Steering Mode"**.

FIGURE 31-4: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)



31.2 Clock Source

The CWG module allows the following clock sources to be selected:

- Fosc (system clock)
- HFINTOSC (16 MHz only)

The clock sources are selected using the CS bit of the CWG1CLKCON register.

REGISTER 31-8: CWG1CLK: CWG1 CLOCK SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	CS
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-1 **Unimplemented:** Read as '0'

bit 0 **CS:** CWG1 Clock Selection bit
1 = HFINTOSC 16 MHz is selected
0 = FOSC is selected

REGISTER 31-9: CWG1ISM: CWG1 INPUT SELECTION REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	IS<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **IS<3:0>:** CWG1 Input Selection bits
1011-1111 = Reserved. No channel connected.
1010 = LC4_out
1001 = LC3_out
1000 = LC2_out
0111 = LC1_out
0110 = Comparator C2 out
0101 = Comparator C1 out
0100 = PWM4_out
0011 = PWM3_out
0010 = CCP2_out
0001 = CCP1_out
0000 = CWG11CLK

33.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULES

33.1 MSSP Module Overview

The Master Synchronous Serial Port (MSSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module can operate in one of two modes:

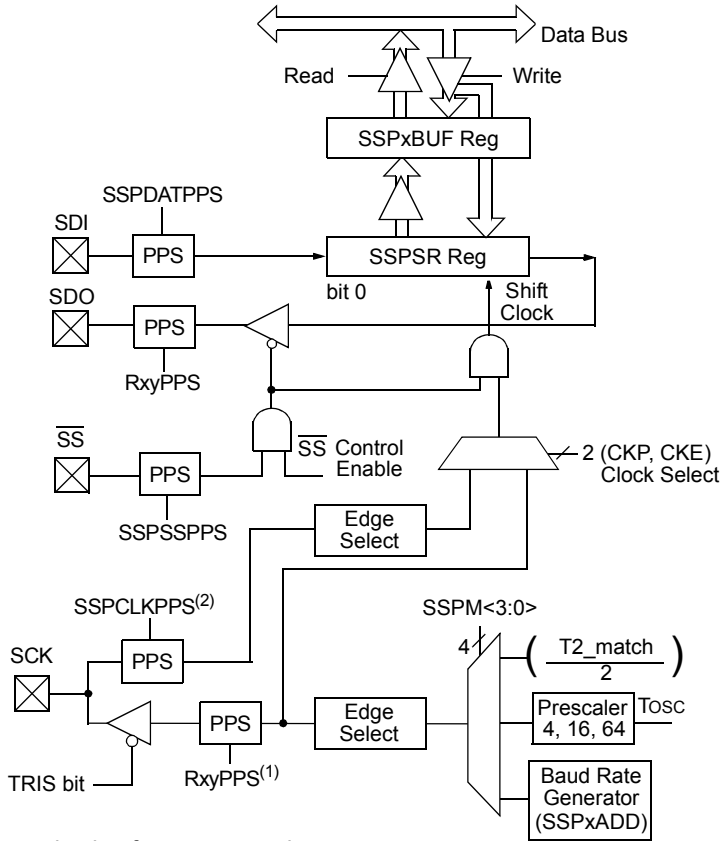
- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C)

The SPI interface supports the following modes and features:

- Master mode
- Slave mode
- Clock Polarity
- Slave Select Synchronization (Slave mode only)
- Daisy-chain connection of slave devices

Figure 33-1 is a block diagram of the SPI interface module.

FIGURE 33-1: MSSP BLOCK DIAGRAM (SPI MODE)



Note 1: Output selection for master mode

2: Input selection for slave mode

FIGURE 33-34: BUS COLLISION DURING START CONDITION (SCL = 0)

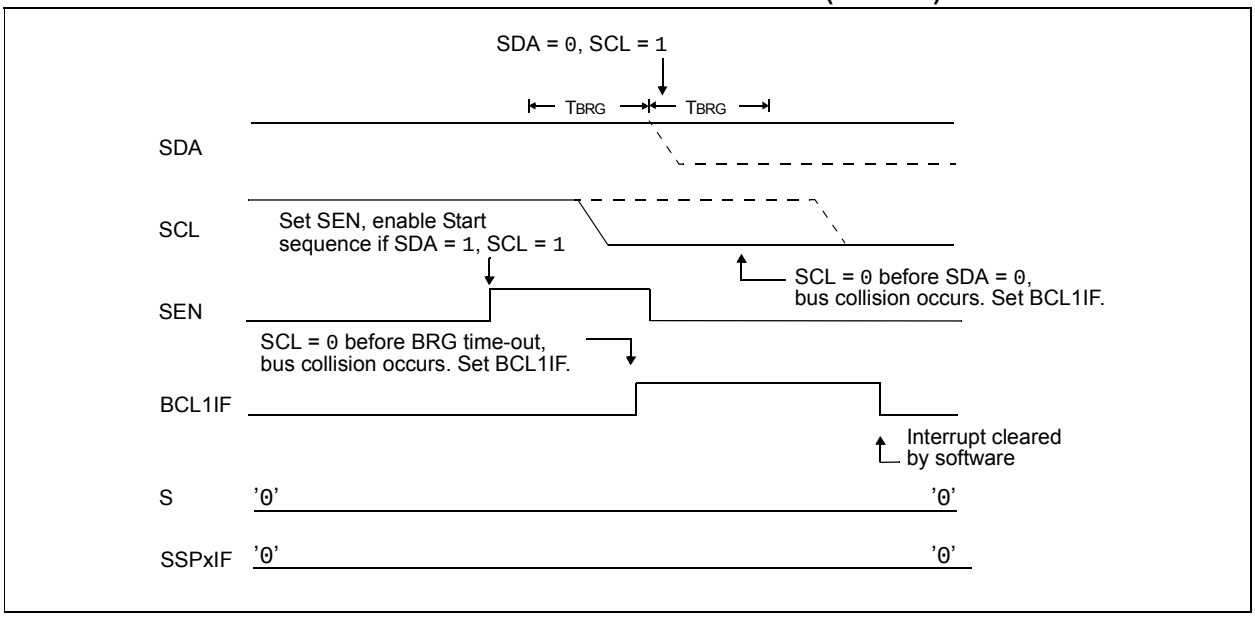
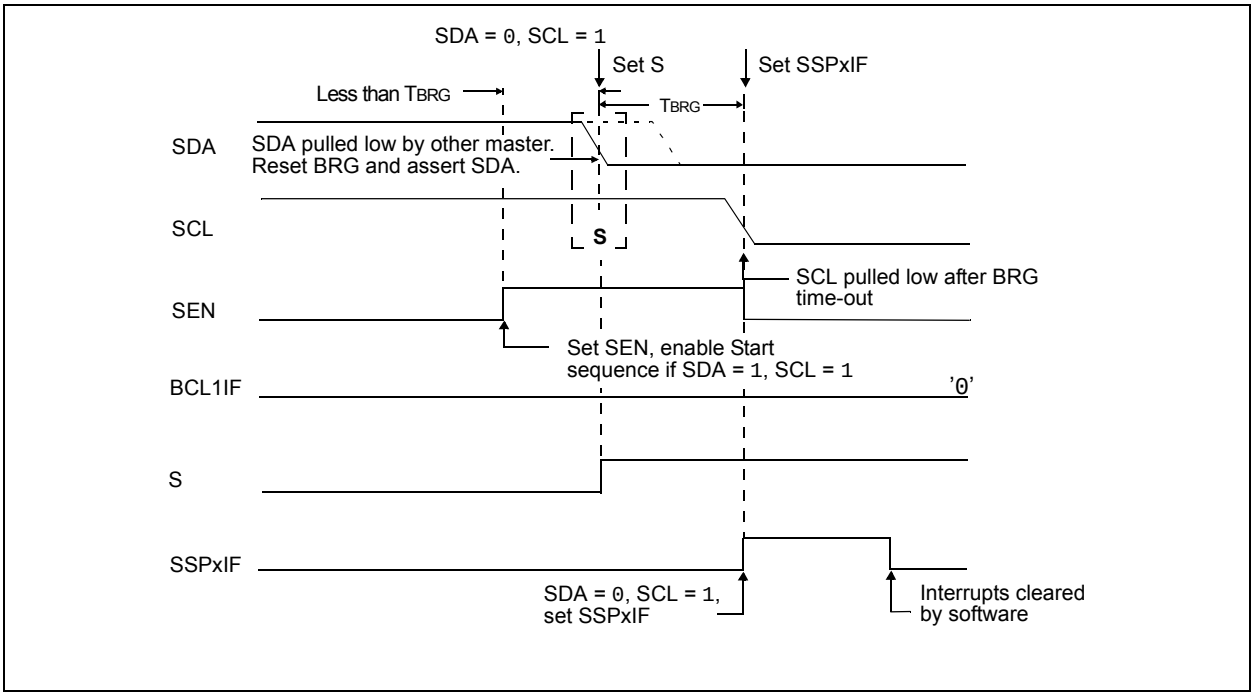


FIGURE 33-35: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION

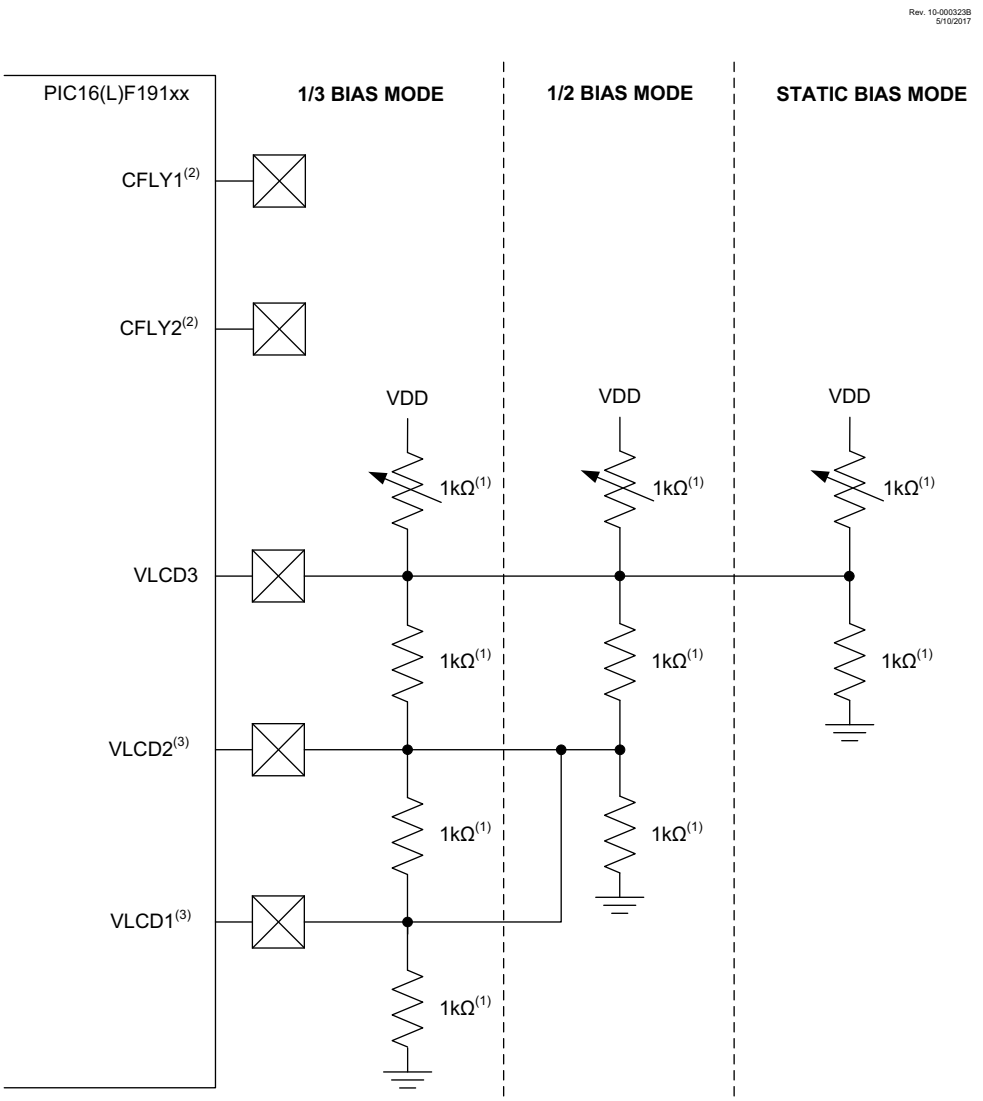


35.7.3 LCD VOLTAGE SUPPLIED FROM EXTERNAL RESISTOR LADDER

In this mode, the LCD charge pump is completely disabled. The LCD bias levels are tied to VDD and are generated using an external divider. The difference is that the internal voltage reference is also disabled and the bottom of the ladder is tied to ground (VSS); see Figure 35-6. The value of the resistors, and the difference between VSS and VDD, determine the contrast range; no software adjustment is possible. This configuration is also used where the LCD's current requirements exceed the capacity of the charge pump and the high power (HP) internal resistor ladder and the software contrast control is not needed.

Depending on the bias type required, resistors are connected between some or all of the pins. A potentiometer can also be connected between VLCD3 and VDD to allow for hardware controlled contrast adjustment.

FIGURE 35-6: CONNECTIONS FOR LCD VOLTAGE SUPPLIED FROM EXTERNAL LADDER, STATIC, 1/2 AND 1/3 BIAS MODES (LCDVSRC<3:0> = 1000)



Note 1: These values are provided for design guidance only; they should be optimized for the application by the designer.

2: It can be used as GPIO.

3: In 1/2 Bias mode, this pin can be used a GPIO.

4: In Static mode, this pin can be used a GPIO.

REGISTER 35-1: LCDCON: LCD CONTROL REGISTER

R/W-0	R/W-0	HS/C-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
LCDEN	SLPEN	WERR	CS	LMUX3	LMUX2	LMUX1	LMUX0
bit 7							bit 0

Legend:	C = Clearable bit	HS = Bit is set by hardware	
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 **LCDEN**: LCD Enable bit
1 = LCD module is enabled
0 = LCD module is disabled

bit 6 **SLPEN**: LCD Display Sleep-Enabled bit
1 = Module will stop driving in Sleep
0 = Module will continue driving in Sleep

bit 5 **WERR**: LCD Write Failed Error bit(1)
1 = Write failure to LCDDATA register occurred (must be reset in software)
0 = No LCD write error

bit 4 **CS**: Clock Source Select bit
1 = SOSC Selected
0 = LFINTOSC Selected

bit 3-0 **LMUX<3:0>**: Common Selection bits. Specifies the number of commons(2)

LMUX<3:0>	Multiplex	Bias
0000	All COMs off	—
0001	Static (COM0)	Static
0010	1/2 MUX (COM<1:0>)	1/2
0011	1/3 MUX (COM<2:0>)	1/3
0100	1/4 MUX (COM<3:0>)	1/3
0101	1/5 MUX (COM<4:0>)	1/3
0110	1/6 MUX (COM<5:0>)	1/3
0111	1/7 MUX (COM<6:0>)	1/3
1000	1/8 MUX (COM<7:0>)	1/3

Note 1: Bit can only be set by hardware and only cleared in software by writing to zero.
2: Cannot be changed when LCDEN = 1.

TABLE 38-1: REGISTER FILE SUMMARY FOR PIC16(L)F19195/6/7 DEVICES (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
1D20h	LCDDATA8	S23C1	S22C1	S21C1	S20C1	S19C1	S18C1	S17C1	S16C1	615
1D21h	LCDDATA9	S31C1	S30C1	S29C1	S28C1	S27C1	S26C1	S25C1	S24C1	615
1D22h	LCDDATA10	S39C1	S38C1	S37C1	S36C1	S35C1	S34C1	S33C1	S32C1	615
1D23h	LCDDATA11	—	—	S45C1	S44C1	S43C1	S42C1	S41C1	S40C1	615
1D24h	LCDDATA12	S07C2	S06C2	S05C2	S04C2	S03C2	S02C2	S01C2	S00C2	615
1D25h	LCDDATA13	S15C2	S14C2	S13C2	S12C2	S11C2	S10C2	S09C2	S08C2	615
1D26h	LCDDATA14	S23C2	S22C2	S21C2	S20C2	S19C2	S18C2	S17C2	S16C2	615
1D27h	LCDDATA15	S31C2	S30C2	S29C2	S28C2	S27C2	S26C2	S25C2	S24C2	615
1D28h	LCDDATA16	S39C2	S38C2	S37C2	S36C2	S35C2	S34C2	S33C2	S32C2	615
1D29h	LCDDATA17	—	—	S45C2	S44C2	S43C2	S42C2	S41C2	S40C2	615
1D2Ah	LCDDATA18	S07C3	S06C3	S05C3	S04C3	S03C3	S02C3	S01C3	S00C3	615
1D2Bh	LCDDATA19	S15C3	S14C3	S13C3	S12C3	S11C3	S10C3	S09C3	S08C3	615
1D2Ch	LCDDATA20	S23C3	S22C3	S21C3	S20C3	S19C3	S18C3	S17C3	S16C3	615
1D2Dh	LCDDATA21	S31C3	S30C3	S29C3	S28C3	S27C3	S26C3	S25C3	S24C3	615
1D2Eh	LCDDATA22	S39C3	S38C3	S37C3	S36C3	S35C3	S34C3	S33C3	S32C3	615
1D2Fh	LCDDATA23	—	—	S45C3	S44C3	S43C3	S42C3	S41C3	S40C3	615
1D30h	LCDDATA24	S07C4	S06C4	S05C4	S04C4	S03C4	S02C4	S01C4	S00C4	615
1D31h	LCDDATA25	S15C4	S14C4	S13C4	S12C4	S11C4	S10C4	S09C4	S08C4	615
1D32h	LCDDATA26	S23C4	S22C4	S21C4	S20C4	S19C4	S18C4	S17C4	S16C4	615
1D33h	LCDDATA27	S31C4	S30C4	S29C4	S28C4	S27C4	S26C4	S25C4	S24C4	615
1D34h	LCDDATA28	S39C4	S38C4	S37C4	S36C4	S35C4	S34C4	S33C4	S32C4	615
1D35h	LCDDATA29	—	—	S45C4	S44C4	S43C4	S42C4	S41C4	S40C4	615
1D36h	LCDDATA30	S07C5	S06C5	S05C5	S04C5	S03C5	S02C5	S01C5	S00C5	615
1D37h	LCDDATA31	S15C5	S14C5	S13C5	S12C5	S11C5	S10C5	S09C5	S08C5	615
1D38h	LCDDATA32	S23C5	S22C5	S21C5	S20C5	S19C5	S18C5	S17C5	S16C5	615
1D39h	LCDDATA33	S31C5	S30C5	S29C5	S28C5	S27C5	S26C5	S25C5	S24C5	615
1D3Ah	LCDDATA34	S39C5	S38C5	S37C5	S36C5	S35C5	S34C5	S33C5	S32C5	615
1D3Bh	LCDDATA35	—	—	S45C5	S44C5	S43C5	S42C5	S41C5	S40C5	615
1D3Ch	LCDDATA36	S07C6	S06C6	S05C6	S04C6	S03C6	S02C6	S01C6	S00C6	615
1D3Dh	LCDDATA37	S15C6	S14C6	S13C6	S12C6	S11C6	S10C6	S09C6	S08C6	615
1D3Eh	LCDDATA38	S23C6	S22C6	S21C6	S20C6	S19C6	S18C6	S17C6	S16C6	615
1D3Fh	LCDDATA39	S31C6	S30C6	S29C6	S28C6	S27C6	S26C6	S25C6	S24C6	615
1D40h	LCDDATA40	S39C6	S38C6	S37C6	S36C6	S35C6	S34C6	S33C6	S32C6	615
1D41h	LCDDATA41	—	—	S45C6	S44C6	S43C6	S42C6	S41C6	S40C6	615
1D42h	LCDDATA42	S07C7	S06C7	S05C7	S04C7	S03C7	S02C7	S01C7	S00C7	615
1D43h	LCDDATA43	S15C7	S14C7	S13C7	S12C7	S11C7	S10C7	S09C7	S08C7	615
1D44h	LCDDATA44	S23C7	S22C7	S21C7	S20C7	S19C7	S18C7	S17C7	S16C7	615
1D45h	LCDDATA45	S31C7	S30C7	S29C7	S28C7	S27C7	S26C7	S25C7	S24C7	615
1D46h	LCDDATA46	S39C7	S38C7	S37C7	S36C7	S35C7	S34C7	S33C7	S32C7	615
1D47h	LCDDATA47	—	—	S45C7	S44C7	S43C7	S42C7	S41C7	S40C7	615
1D48h — 1D6Fh	—	Unimplemented								

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Unimplemented data memory locations, read as '0'.

FIGURE 39-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

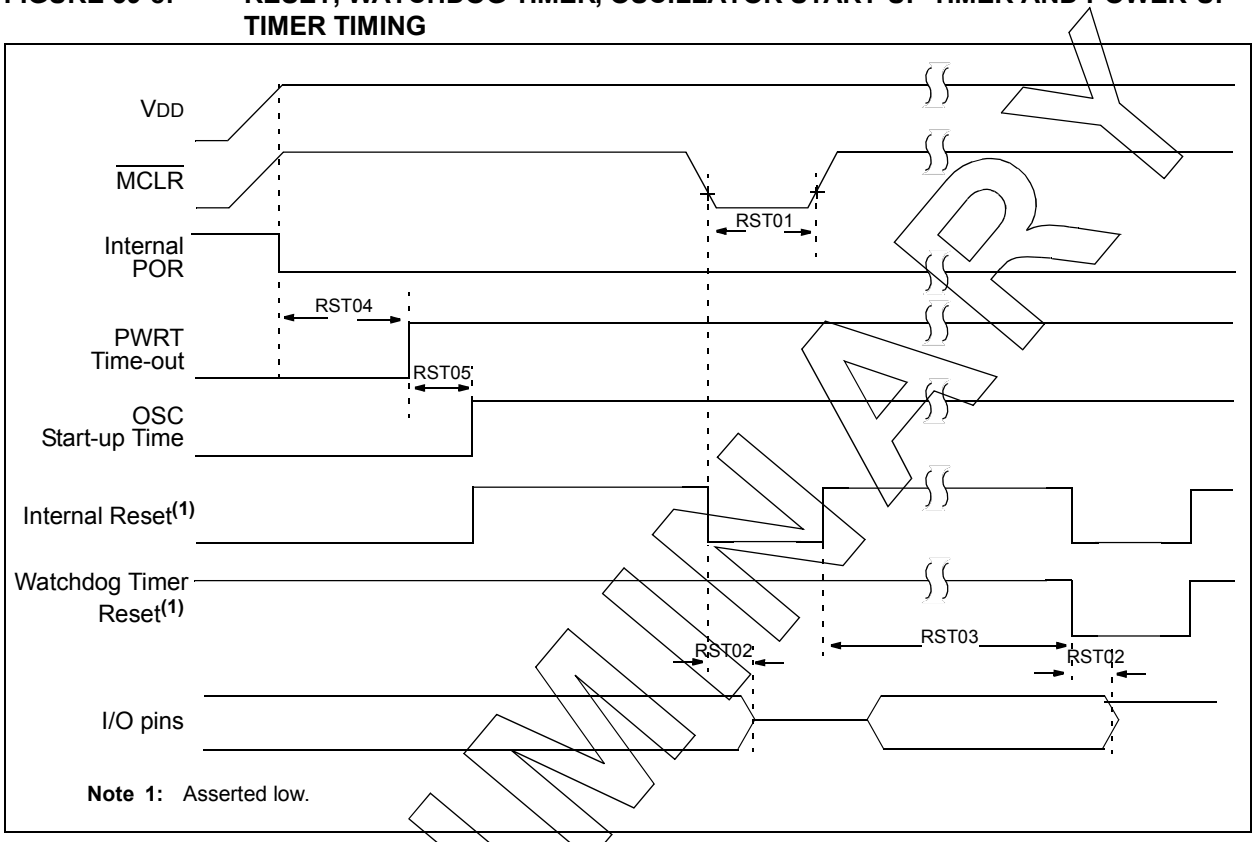


FIGURE 39-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS