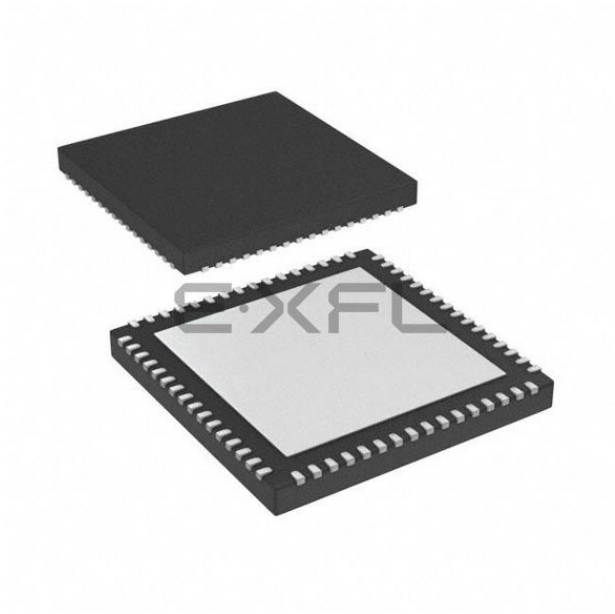


Welcome to E-XFL.COM

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	54
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 17x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1947t-i-mr

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Website at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Website; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our website at **www.microchip.com** to receive the most current information on all of our products.

TABLE 3-8: PIC16(L)F1946/47 MEMORY MAP, BANK 15

Bank 15	
791h	LCDCON
792h	LCDPS
793h	LCDREF
794h	LCDCST
795h	LCDRL
796h	—
797h	—
798h	LCDSE0
799h	LCDSE1
79Ah	LCDSE2
79Bh	LCDSE3
79Ch	LCDSE4
79Dh	LCDSE5
79Eh	—
79Fh	—
7A0h	LCDDATA0
7A1h	LCDDATA1
7A2h	LCDDATA2
7A3h	LCDDATA3
7A4h	LCDDATA4
7A5h	LCDDATA5
7A6h	LCDDATA6
7A7h	LCDDATA7
7A8h	LCDDATA8
7A9h	LCDDATA9
7AAh	LCDDATA10
7ABh	LCDDATA11
7ACh	LCDDATA12
7ADh	LCDDATA13
7AEh	LCDDATA14
7AFh	LCDDATA15
7B0h	LCDDATA16
7B1h	LCDDATA17
7B2h	LCDDATA18
7B3h	LCDDATA19
7B4h	LCDDATA20
7B5h	LCDDATA21
7B6h	LCDDATA22
7B7h	LCDDATA23
7B8h – 7EFh	Unimplemented Read as ‘0’

Legend: = Unimplemented data memory locations, read as ‘0’.

TABLE 3-9: PIC16(L)F1946/47 MEMORY MAP, BANK 31

Bank 31	
F8Ch	—
F8Dh	—
F8Eh	—
F8Fh	—
F90h	—
F91h	—
F92h	—
F93h	—
F94h	—
F95h	—
F96h	—
F97h	—
F98h	—
F99h	—
F9Ah	—
F9Bh	—
F9Ch	—
F9Dh	—
F9Eh	—
F9Fh	—
FA0h	—
FA1h	—
FA2h	—
FA3h	—
FA4h	—
FA5h	—
FA6h	—
FA7h	—
FA8h	—
FA9h	—
FAAh	—
FABh	—
FDFh	—
FC0h	—
FDFh	—
FE0h	—
FE1h	—
FE2h	—
FE3h	—
FE4h	STATUS_SHAD
FE5h	WREG_SHAD
FE6h	BSR_SHAD
FE7h	PCLATH_SHAD
FE8h	FSR0L_SHAD
FE9h	FSR0H_SHAD
FEAh	FSR1L_SHAD
FEBh	FSR1H_SHAD
FECh	—
FEDh	STKPTR
FEEh	TOSL
FEFh	TOSH

Legend: = Unimplemented data memory locations, read as ‘0’.

TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (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 all other Resets
Bank 7											
380h(2)	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
381h(2)	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
382h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
383h(2)	STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu
384h(2)	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
385h(2)	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
386h(2)	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
387h(2)	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
388h(2)	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
389h(2)	WREG	Working Register								0000 0000	uuuu uuuu
38Ah(1, 2)	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
38Bh(2)	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
38Ch	LATF	PORTF Data Latch								xxxx xxxx	uuuu uuuu
38Dh	LATG	—	—	LATG5	LATG4	LATG3	LATG2	LATG1	LATG0	--xx xxxx	--uu uuuu
38Eh	—	Unimplemented								—	—
38Fh	—	Unimplemented								—	—
390h	—	Unimplemented								—	—
391h	—	Unimplemented								—	—
392h	—	Unimplemented								—	—
393h	—	Unimplemented								—	—
394h	IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0	0000 0000	0000 0000
395h	IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0	0000 0000	0000 0000
396h	IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0	0000 0000	0000 0000
397h	—	Unimplemented								—	—
398h	—	Unimplemented								—	—
399h	—	Unimplemented								—	—
39Ah	—	Unimplemented								—	—
39Bh	—	Unimplemented								—	—
39Ch	—	Unimplemented								—	—
39Dh	—	Unimplemented								—	—
39Eh	—	Unimplemented								—	—
39Fh	—	Unimplemented								—	—

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<14:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: Unimplemented, read as '1'.

TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (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 all other Resets
Bank 8											
400h(2)	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
401h(2)	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
402h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
403h(2)	STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu
404h(2)	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
405h(2)	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
406h(2)	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
407h(2)	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
408h(2)	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
409h(2)	WREG	Working Register								0000 0000	uuuu uuuu
40Ah(1, 2)	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
40Bh(2)	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
40Ch	ANSELF	ANSELF7	ANSELF6	ANSELF5	ANSELF4	ANSELF3	ANSELF2	ANSELF1	ANSELF0	1111 1111	1111 1111
40Dh	ANSELG	—	—	—	ANSELG4	ANSELG3	ANSELG2	ANSELG1	—	---1 111-	---1 111-
40Eh	—	Unimplemented								—	—
40Fh	—	Unimplemented								—	—
410h	—	Unimplemented								—	—
411h	—	Unimplemented								—	—
412h	—	Unimplemented								—	—
413h	—	Unimplemented								—	—
414h	—	Unimplemented								—	—
415h	TMR4	Timer 4 Module Register								0000 0000	0000 0000
416h	PR4	Timer 4 Period Register								1111 1111	1111 1111
417h	T4CON	—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		-000 0000	-000 0000
418h	—	Unimplemented								—	—
419h	—	Unimplemented								—	—
41Ah	—	Unimplemented								—	—
41Bh	—	Unimplemented								—	—
41Ch	TMR6	Timer 6 Module Register								0000 0000	0000 0000
41Dh	PR6	Timer 6 Period Register								1111 1111	1111 1111
41Eh	T6CON	—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		-000 0000	-000 0000
41Fh	—	Unimplemented								—	—

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<14:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: Unimplemented, read as '1'.

5.2.2 INTERNAL CLOCK SOURCES

The device may be configured to use the internal oscillator block as the system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in Configuration Words to select the INTOSC clock source, which will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to the internal oscillator during run-time. See **Section 5.3 "Clock Switching"**for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT is available for general purpose I/O or CLKOUT.

The function of the OSC2/CLKOUT pin is determined by the $\overline{\text{CLKOUTEN}}$ bit in Configuration Words.

The internal oscillator block has two independent oscillators and a dedicated Phase-Lock Loop, HFPLL that can produce one of three internal system clock sources.

1. The **HFINTOSC** (High-Frequency Internal Oscillator) is factory calibrated and operates at 16 MHz. The HFINTOSC source is generated from the 500 kHz MFINTOSC source and the dedicated Phase-Lock Loop, HFPLL. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
2. The **MFINTOSC** (Medium-Frequency Internal Oscillator) is factory calibrated and operates at 500 kHz. The frequency of the MFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
3. The **LFINTOSC** (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

5.2.2.1 HFINTOSC

The High-Frequency Internal Oscillator (HFINTOSC) is a factory-calibrated 16 MHz internal clock source. The frequency of the HFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

The output of the HFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of multiple frequencies derived from the HFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See **Section 5.2.2.7 "Internal Oscillator Clock Switch Timing"** for more information.

The HFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = `100`, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '`1x`'.

A fast start-up oscillator allows internal circuits to power-up and stabilize before switching to HFINTOSC.

The High-Frequency Internal Oscillator Ready bit (HFIOFR) of the OSCSTAT register indicates when the HFINTOSC is running.

The High-Frequency Internal Oscillator Status Locked bit (HFIOFL) of the OSCSTAT register indicates when the HFINTOSC is running within 2% of its final value.

The High-Frequency Internal Oscillator Stable bit (HFIOFS) of the OSCSTAT register indicates when the HFINTOSC is running within 0.5% of its final value.

5.2.2.2 MFINTOSC

The Medium-Frequency Internal Oscillator (MFINTOSC) is a factory calibrated 500 kHz internal clock source. The frequency of the MFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

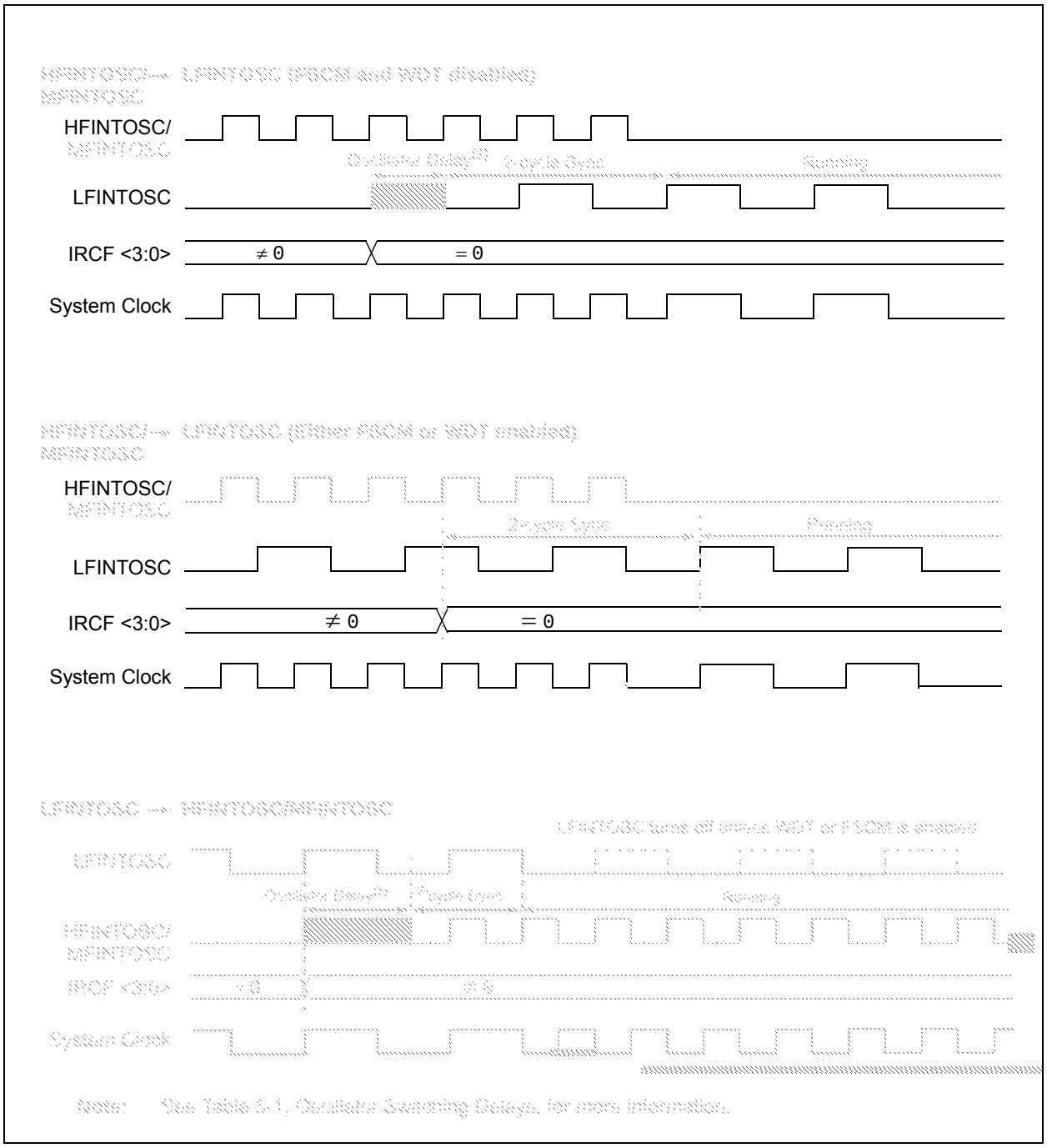
The output of the MFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of nine frequencies derived from the MFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See **Section 5.2.2.7 "Internal Oscillator Clock Switch Timing"** for more information.

The MFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = `100`, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '`1x`'

The Medium-Frequency Internal Oscillator Ready bit (MFIOFR) of the OSCSTAT register indicates when the MFINTOSC is running.

FIGURE 5-7: INTERNAL OSCILLATOR SWITCH TIMING



7.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIE1, PIE2, PIE3 and PIE4 registers)

The INTCON, PIR1, PIR2, PIR3 and PIR4 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while the GIE bit is set:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See **"Section 7.5 "Automatic Context Saving"."**)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The `RETFIE` instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupt's operation, refer to its peripheral chapter.

Note 1:	Individual interrupt flag bits are set, regardless of the state of any other enable bits.
2:	All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again.

7.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The latency for synchronous interrupts is 3 or 4 instruction cycles. For asynchronous interrupts, the latency is 3 to 5 instruction cycles, depending on when the interrupt occurs. See Figure 7-2 and Figure 7-3 for more details.

8.0 LOW DROPOUT (LDO) VOLTAGE REGULATOR

The PIC16F1946/47 has an internal Low Dropout Regulator (LDO) which provides operation above 3.6V. The LDO regulates a voltage for the internal device logic while permitting the VDD and I/O pins to operate at a higher voltage. There is no user enable/disable control available for the LDO, it is always active. The PIC16LF1946/47 operates at a maximum VDD of 3.6V and does not incorporate an LDO.

A device I/O pin may be configured as the LDO voltage output, identified as the VCAP pin. Although not required, an external low-ESR capacitor may be connected to the VCAP pin for additional regulator stability.

The VCAPEN bit of Configuration Words enables or disables the VCAP pin. Refer to Table 8-1.

TABLE 8-1: VCAPEN SELECT BIT

VCAPEN	Pin
0	RF0
1	No VCAP

On power-up, the external capacitor will load the LDO voltage regulator. To prevent erroneous operation, the device is held in Reset while a constant current source charges the external capacitor. After the cap is fully charged, the device is released from Reset. For more information on the constant current rate, refer to the LDO Regulator Characteristics Table in **Section 30.0 "Electrical Specifications"**.

TABLE 8-2: SUMMARY OF CONFIGURATION WORD WITH LDO

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG2	13:8	—	—	LVP	$\overline{\text{DEBUG}}$	—	BORV	STVREN	PLLEN	56
	7:0	—	—	—	VCAPEN	—	—	WRT1	WRT0	

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by LDO.

10.1 Independent Clock Source

The WDT derives its time base from the 31 kHz LFINTOSC internal oscillator. Time intervals in this chapter are based on a nominal interval of 1 ms. See **Section 30.0 "Electrical Specifications"** for the LFINTOSC tolerances.

10.2 WDT Operating Modes

The Watchdog Timer module has four operating modes controlled by the WDTE<1:0> bits in Configuration Words. See Table 10-1.

10.2.1 WDT IS ALWAYS ON

When the WDTE bits of Configuration Words are set to '11', the WDT is always on.

WDT protection is active during Sleep.

10.2.2 WDT IS OFF IN SLEEP

When the WDTE bits of Configuration Words are set to '10', the WDT is on, except in Sleep.

WDT protection is not active during Sleep.

10.2.3 WDT CONTROLLED BY SOFTWARE

When the WDTE bits of Configuration Words are set to '01', the WDT is controlled by the SWDTEN bit of the WDTCON register.

WDT protection is unchanged by Sleep. See Table 10-1 for more details.

TABLE 10-1: WDT OPERATING MODES

WDTE<1:0>	SWDTEN	Device Mode	WDT Mode
11	X	X	Active
10	X	Awake	Active
		Sleep	Disabled
01	1	X	Active
	0		Disabled
00	X	X	Disabled

10.3 Time-Out Period

The WDTPS bits of the WDTCON register set the time-out period from 1 ms to 256 seconds (nominal). After a Reset, the default time-out period is two seconds.

10.4 Clearing the WDT

The WDT is cleared when any of the following conditions occur:

- Any Reset
- CLRWDT instruction is executed
- Device enters Sleep
- Device wakes up from Sleep
- Oscillator fail
- WDT is disabled
- Oscillator Start-up Timer (OST) is running

See Table 10-2 for more information.

10.5 Operation During Sleep

When the device enters Sleep, the WDT is cleared. If the WDT is enabled during Sleep, the WDT resumes counting.

When the device exits Sleep, the WDT is cleared again. The WDT remains clear until the OST, if enabled, completes. See **Section 5.0 "Oscillator Module (With Fail-Safe Clock Monitor)"** for more information on the OST.

When a WDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register are changed to indicate the event. See **Section 3.0 "Memory Organization"** and STATUS register (**Register 3-1)** for more information.

TABLE 10-2: WDT CLEARING CONDITIONS

Conditions	WDT
WDTE<1:0> = 00	Cleared
WDTE<1:0> = 01 and SWDTEN = 0	
WDTE<1:0> = 10 and enter Sleep	
CLRWDT Command	
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK	
Exit Sleep + System Clock = XT, HS, LP	Cleared until the end of OST
Change INTOSC divider (IRCF bits)	Unaffected

REGISTER 11-5: EECON1: EEPROM CONTROL 1 REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W-x/q	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
S = Bit can only be set	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HC = Bit is cleared by hardware

bit 7 **EEPGD:** Flash Program/Data EEPROM Memory Select bit
- 1 = Accesses program space Flash memory
- 0 = Accesses data EEPROM memory

bit 6 **CFGS:** Flash Program/Data EEPROM or Configuration Select bit
- 1 = Accesses Configuration, User ID and Device ID Registers
- 0 = Accesses Flash Program or data EEPROM Memory

bit 5 **LWLO:** Load Write Latches Only bit

If CFGS = 1 (Configuration space) **OR** CFGS = 0 and EEPGD = 1 (program Flash):
- 1 = The next WR command does not initiate a write; only the program memory latches are updated.
- 0 = The next WR command writes a value from EEDATH:EEDATL into program memory latches and initiates a write of all the data stored in the program memory latches.

If CFGS = 0 and EEPGD = 0: (Accessing data EEPROM)
LWLO is ignored. The next WR command initiates a write to the data EEPROM.

bit 4 **FREE:** Program Flash Erase Enable bit

If CFGS = 1 (Configuration space) **OR** CFGS = 0 and EEPGD = 1 (program Flash):
- 1 = Performs an erase operation on the next WR command (cleared by hardware after completion of erase).
- 0 = Performs a write operation on the next WR command.

If EEPGD = 0 and CFGS = 0: (Accessing data EEPROM)
FREE is ignored. The next WR command will initiate both a erase cycle and a write cycle.

bit 3 **WRERR:** EEPROM Error Flag bit
- 1 = Condition indicates an improper program or erase sequence attempt or termination (bit is set automatically on any set attempt (write '1') of the WR bit).
- 0 = The program or erase operation completed normally.

bit 2 **WREN:** Program/Erase Enable bit
- 1 = Allows program/erase cycles
- 0 = Inhibits programming/erasing of program Flash and data EEPROM

bit 1 **WR:** Write Control bit
- 1 = Initiates a program Flash or data EEPROM program/erase operation.
  The operation is self-timed and the bit is cleared by hardware once operation is complete.
  The WR bit can only be set (not cleared) in software.
- 0 = Program/erase operation to the Flash or data EEPROM is complete and inactive.

bit 0 **RD:** Read Control bit
- 1 = Initiates an program Flash or data EEPROM read. Read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
- 0 = Does not initiate a program Flash or data EEPROM data read.

REGISTER 12-5: ANSELA: PORTA ANALOG SELECT REGISTER

U-0	U-0	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	ANSA5	—	ANSA3	ANSA2	ANSA1	ANSA0
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 7-6 **Unimplemented:** Read as '0'

bit 5 **ANSA5**: Analog Select between Analog or Digital Function on pins RA<5>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.

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

bit 3-0 **ANSA<3:0>**: Analog Select between Analog or Digital Function on pins RA<3:0>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

TABLE 12-3: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON0	—	CHS<4:0>					GO/$\overline{\text{DONE}}$	ADON	161
ADCON1	ADFM	ADCS<2:0>			—	—	ADPREF<1:0>		162
ANSELA	—	—	ANSA5	—	ANSA3	ANSA2	ANSA1	ANSA0	126
CPSCON0	CPSON	CPSRM	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	322
CPSCON1	—	—	—	CPSCH<4:0>					323
DACCON0	DACEN	DACLPS	DACOE	---	DACPSS<1:0>		---	DACNSS	171
LATA	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	125
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	330
LCDSE2	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	330
LCDSE4	SE39	SE38	SE37	SE36	SE35	SE34	SE33	SE32	330
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			188
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	125
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125

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

TABLE 12-4: SUMMARY OF CONFIGURATION WORD WITH PORTA

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG1	13:8	—	—	FCMEN	IESO	$\overline{\text{CLKOUTEN}}$	BOREN<1:0>		$\overline{\text{CPD}}$	54
	7:0	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE<1:0>		FOSC<2:0>			

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by PORTA.

REGISTER 12-18: LATE: PORTE DATA LATCH 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
LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0
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 **LATE<7:0>**: PORTE Output Latch Value bits(1)

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 12-19: ANSELE: PORTE ANALOG SELECT REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	—	—	ANSE2	ANSE1	ANSE0
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 **ANSE<7:0>**: Analog Select between Analog or Digital Function on Pins RE<7:0>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

TABLE 12-12: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON	P3CSEL	P3BSEL	P2DSEL	P2CSEL	P2BSEL	CCP2SEL	P1CSEL	P1BSEL	123
ANSELE	—	—	—	—	—	ANSE2	ANSE1	ANSE0	138
CCPxCON	PxM<1:0>(1)		DCxB<1:0>		CCPxM<3:0>				227
LATE	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	138
LCDCON	LCDEN	SLPEN	WERR	—	CS<1:0>		LMUX<1:0>		326
LCDREF	LCDIRE	LCDIRS	LCDIRI	—	VLCD3PE	VLCD2PE	VLCD1PE	—	328
LCDSE2	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	330
PORTE	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	137
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	137

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTE.
Note 1: Applies to ECCP modules only.

REGISTER 23-6: PSTRxCON: PWM STEERING CONTROL REGISTER(1)

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
—	—	—	STRxSYNC	STRxD	STRxC	STRxB	STRxA
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 7-5 **Unimplemented:** Read as '0'

bit 4 **STRxSYNC:** Steering Sync bit
1 = Output steering update occurs on next PWM period
0 = Output steering update occurs at the beginning of the instruction cycle boundary

bit 3 **STRxD:** Steering Enable bit D
1 = PxD pin has the PWM waveform with polarity control from CCPxM<1:0>
0 = PxD pin is assigned to port pin

bit 2 **STRxC:** Steering Enable bit C
1 = PxC pin has the PWM waveform with polarity control from CCPxM<1:0>
0 = PxC pin is assigned to port pin

bit 1 **STRxB:** Steering Enable bit B
1 = PxB pin has the PWM waveform with polarity control from CCPxM<1:0>
0 = PxB pin is assigned to port pin

bit 0 **STRxA:** Steering Enable bit A
1 = PxA pin has the PWM waveform with polarity control from CCPxM<1:0>
0 = PxA pin is assigned to port pin

Note 1: The PWM Steering mode is available only when the CCPxCON register bits CCPxM<3:2> = 11 and PxM<1:0> = 00.

REGISTER 24-2: SSPxCON1: SSPx CONTROL REGISTER 1

R/C/HS-0/0	R/C/HS-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
WCOL	SSPOV	SSPEN	CKP	SSPM<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	HS = Bit is set by hardware	C = User cleared

bit 7 **WCOL:** Write Collision Detect bit

Master mode:

1 = A write to the SSPxBUF register was attempted while the I^2C conditions were not valid for a transmission to be started

0 = No collision

Slave mode:

1 = The SSPxBUF register is written while it is still transmitting the previous word (must be cleared in software)

0 = No collision

bit 6 **SSPOV:** Receive Overflow Indicator bit(1)

In SPI mode:

1 = A new byte is received while the SSPxBUF register is still holding the previous data. In case of overflow, the data in SSPxSR is lost. Overflow can only occur in Slave mode. In Slave mode, the user must read the SSPxBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPxBUF register (must be cleared in software).

0 = No overflow

In I^2C mode:

1 = A byte is received while the SSPxBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode (must be cleared in software).

0 = No overflow

bit 5 **SSPEN:** Synchronous Serial Port Enable bit

In both modes, when enabled, these pins must be properly configured as input or output

In SPI mode:

1 = Enables serial port and configures SCKx, SDOx, SDIx and $\overline{SSx}$ as the source of the serial port pins(2)

0 = Disables serial port and configures these pins as I/O port pins

In I^2C mode:

1 = Enables the serial port and configures the SDAx and SCLx pins as the source of the serial port pins(3)

0 = Disables serial port and configures these pins as I/O port pins

bit 4 **CKP:** Clock Polarity Select bit

In SPI mode:

1 = Idle state for clock is a high level

0 = Idle state for clock is a low level

In I^2C Slave mode:

SCLx release control

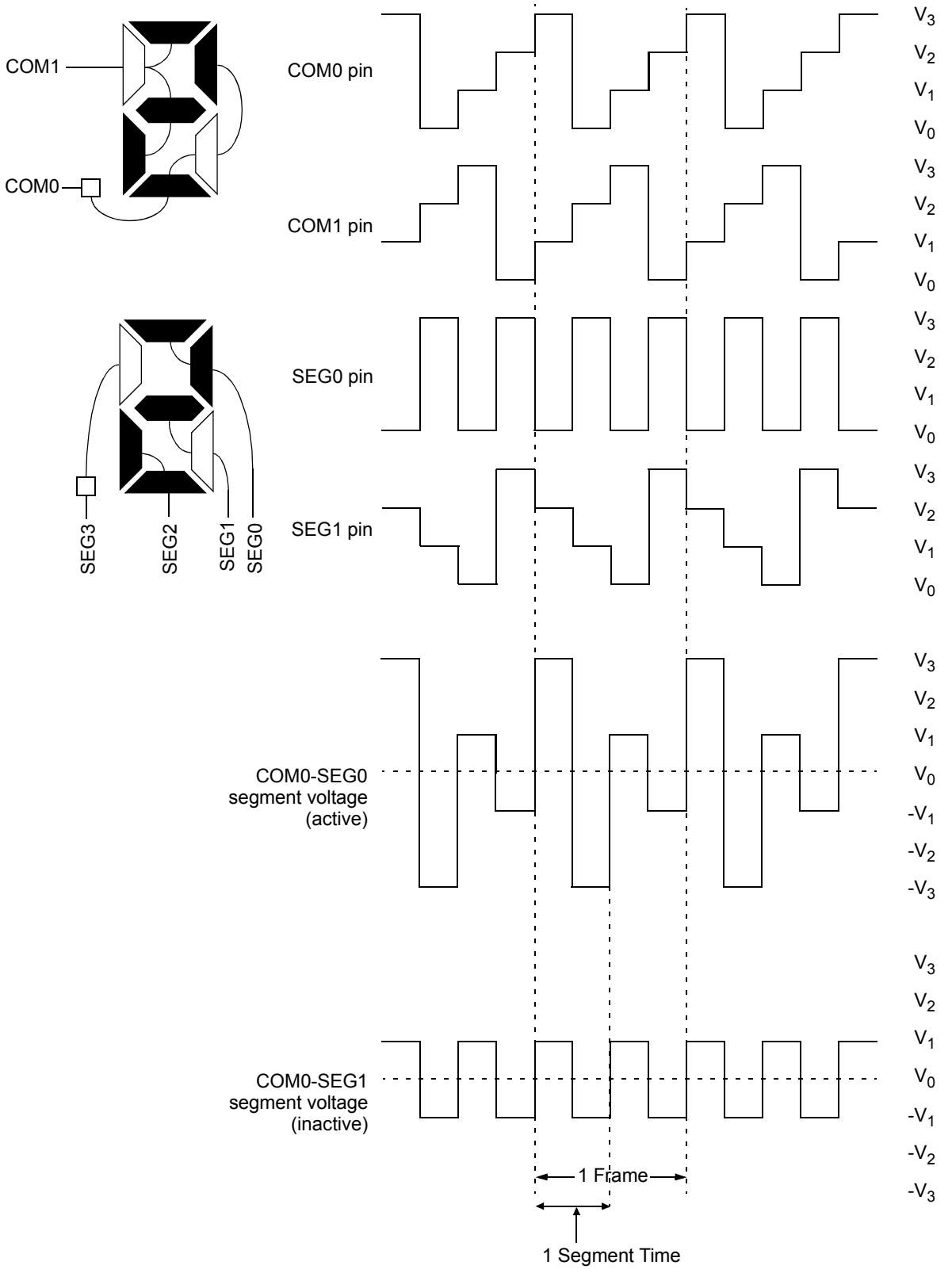
1 = Enable clock

0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

In I^2C Master mode:

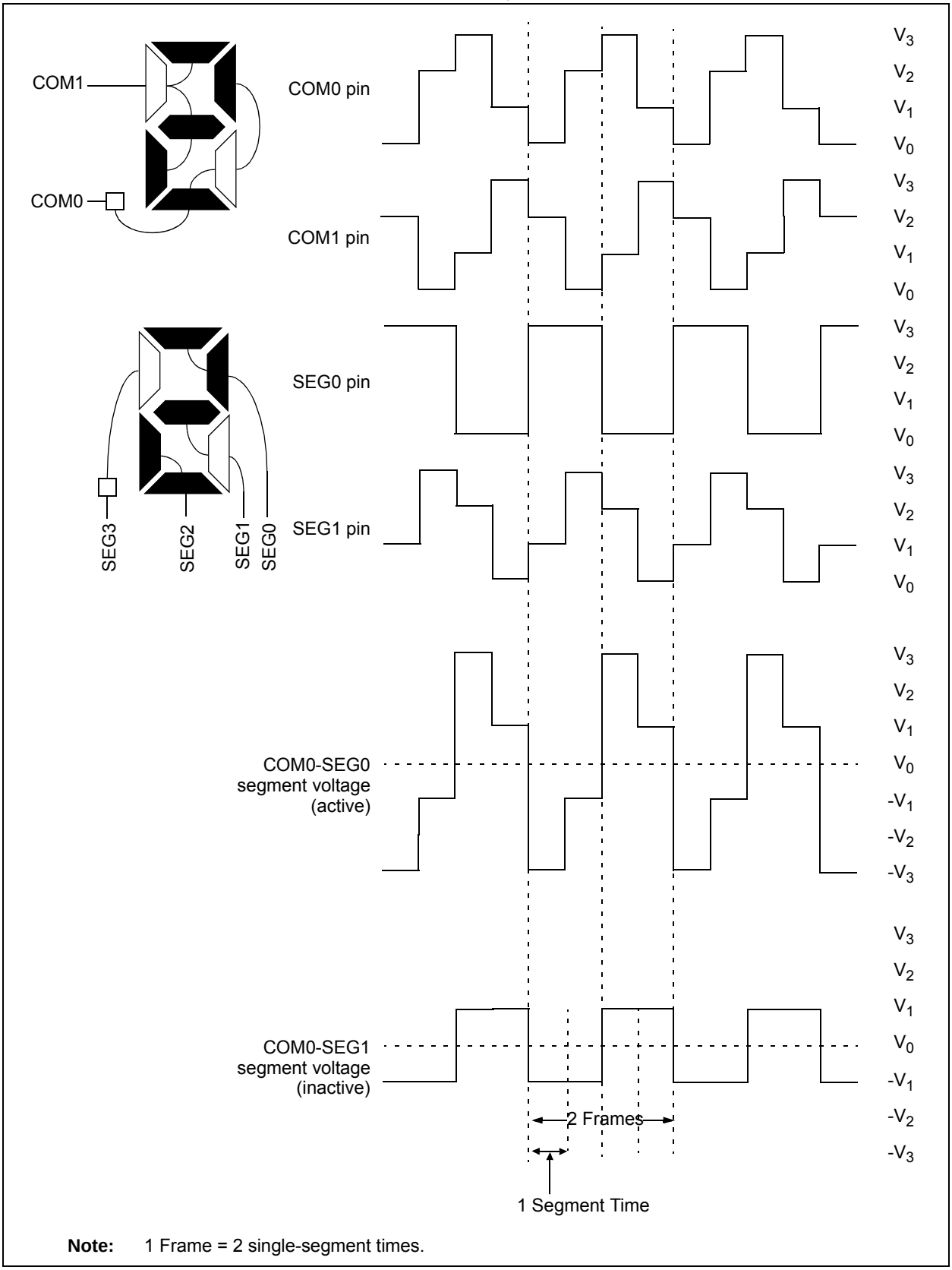
Unused in this mode

FIGURE 27-11: TYPE-A WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE



Note: 1 Frame = 2 single-segment times.

FIGURE 27-12: TYPE-B WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE



Note: 1 Frame = 2 single-segment times.

CALL **Call Subroutine**

Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → PC<14:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a 2-cycle instruction.

CALLW **Subroutine Call With W**

Syntax:	[*label*] CALLW
Operands:	None
Operation:	(PC) +1 → TOS, (W) → PC<7:0>, (PCLATH<6:0>) → PC<14:8>
Status Affected:	None
Description:	Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into PC<14:8>. CALLW is a 2-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.

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.

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.

COMF **Complement f**

Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	($\bar{f}$) → (destination)
Status Affected:	Z
Description:	The contents of register ‘f’ are complemented. If ‘d’ is ‘0’, the result is stored in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’.

DECF **Decrement f**

Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register ‘f’. If ‘d’ is ‘0’, the result is stored in the W register. If ‘d’ is ‘1’, the result is stored back in register ‘f’.

RRF **Rotate Right f through Carry**

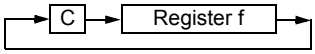
Syntax: [*label*] RRF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.



SLEEP **Enter Sleep mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down Status bit, $\overline{PD}$ is cleared. Time-out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into Sleep mode with the oscillator stopped.

SUBLW **Subtract W from literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.

C = 0	W > k
C = 1	W ≤ k
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f.

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SUBWFB **Subtract W from f with Borrow**

Syntax: SUBWFB f {,d}

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) – (W) – ($\overline{B}$) → dest

Status Affected: C, DC, Z

Description: Subtract W and the BORROW flag (CARRY) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

FIGURE 31-15: IDD TYPICAL, EC OSCILLATOR, HIGH-POWER MODE, PIC16LF1946/47 ONLY

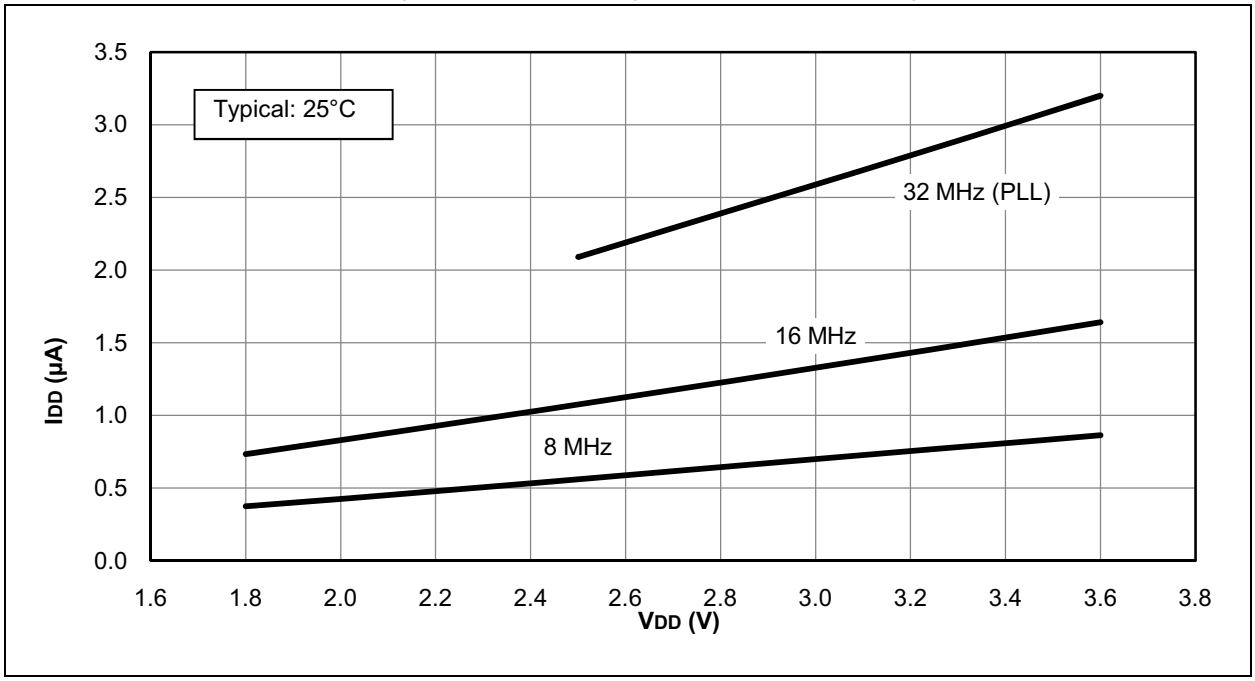


FIGURE 31-16: IDD MAXIMUM, EC OSCILLATOR, HIGH-POWER MODE, PIC16LF1946/47 ONLY

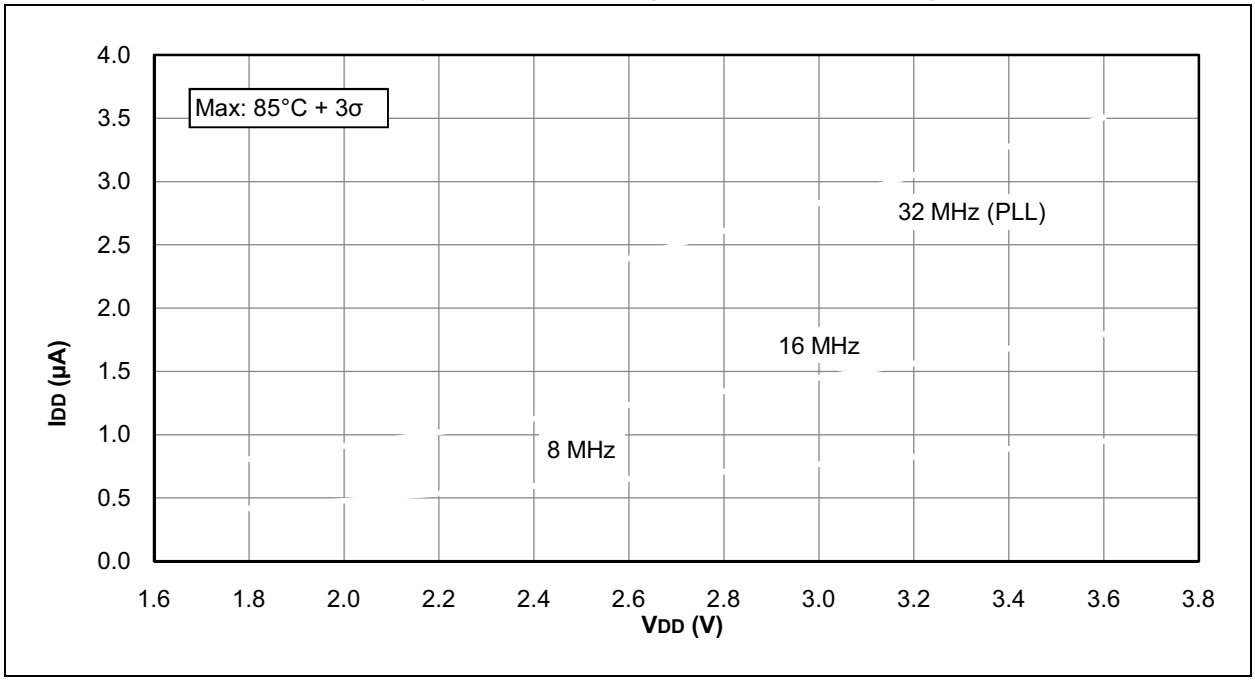


FIGURE 31-45: I_{PD}, CAPACITIVE SENSING (CPS) MODULE, HIGH-CURRENT RANGE, CPSRM = 0, PIC16LF1946/47 ONLY

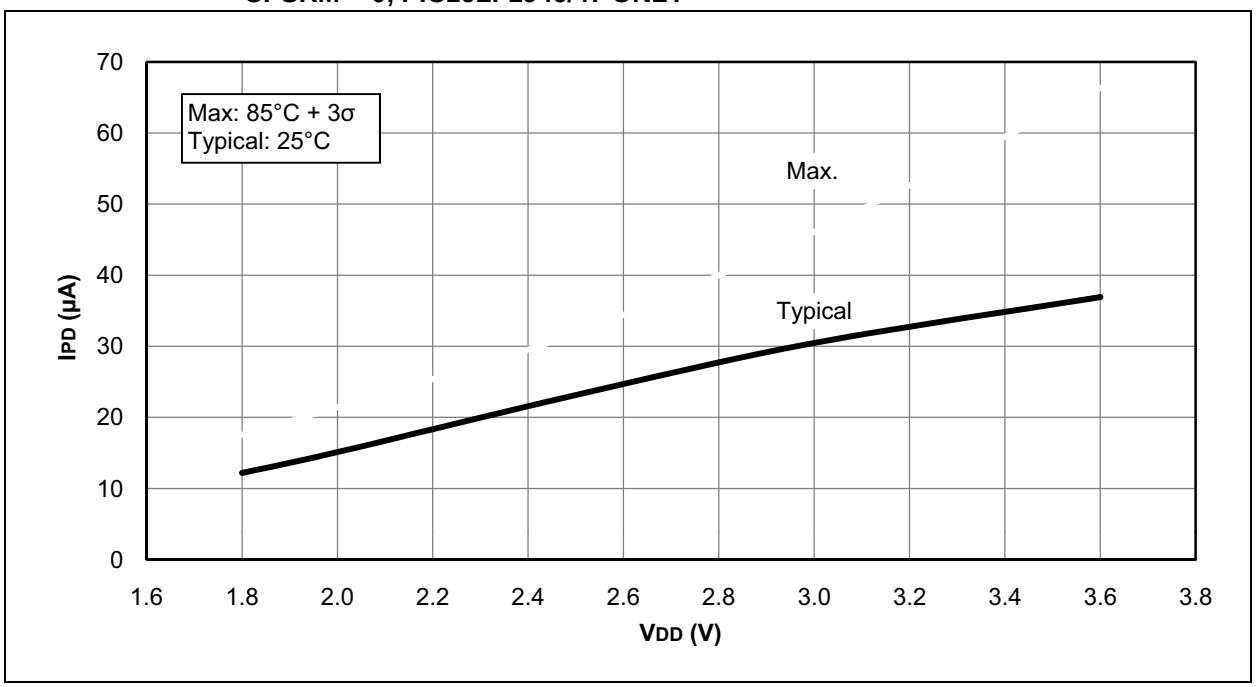


FIGURE 31-46: I_{PD}, CAPACITIVE SENSING (CPS) MODULE, HIGH-CURRENT RANGE, CPSRM = 0, PIC16F1946/47 ONLY