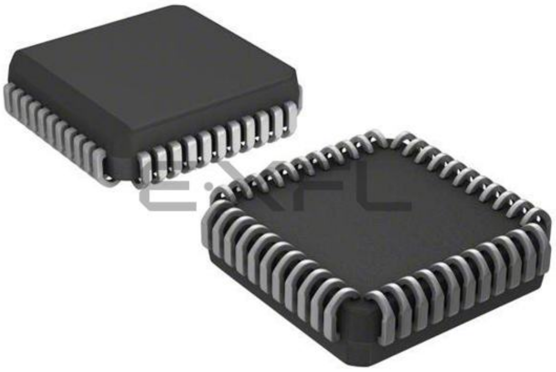


Welcome to E-XFL.COM

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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	33
Program Memory Size	16KB (8K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lc442-i-l

4.7.1 TWO-WORD INSTRUCTIONS

The PIC18CXX2 devices have four two-word instructions: MOVFF, CALL, GOTO and LFSR. The second word of these instructions has the 4 MSBs set to 1's and is a special kind of NOP instruction. The lower 12-bits of the second word contain data to be used by the instruction. If the first word of the instruction is executed, the data in the second word is accessed. If the second word of the instruction is executed by itself (first word was skipped), it will execute as a NOP. This action is necessary when the two-word instruction is preceded by a conditional instruction that changes the PC. A program example that demonstrates this concept is shown in Example 4-3. Refer to Section 19.0 for further details of the instruction set.

EXAMPLE 4-3: TWO-WORD INSTRUCTIONS

CASE 1:			
Object Code	Source Code		
0110 0110 0000 0000	TSTFSZ	REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF	REG1, REG2	; No, execute 2-word instruction
1111 0100 0101 0110			; 2nd operand holds address of REG2
0010 0100 0000 0000	ADDWF	REG3	; continue code
CASE 2:			
Object Code	Source Code		
0110 0110 0000 0000	TSTFSZ	REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF	REG1, REG2	; Yes
1111 0100 0101 0110			; 2nd operand becomes NOP
0010 0100 0000 0000	ADDWF	REG3	; continue code

4.8 Lookup Tables

Lookup tables are implemented two ways. These are:

- Computed GOTO
- Table Reads

4.8.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL).

A lookup table can be formed with an ADDWF PCL instruction and a group of RETLW 0xnn instructions. WREG is loaded with an offset into the table, before executing a call to that table. The first instruction of the called routine is the ADDWF PCL instruction. The next instruction executed will be one of the RETLW 0xnn instructions that returns the value 0xnn to the calling function.

The offset value (value in WREG) specifies the number of bytes that the program counter should advance.

In this method, only one data byte may be stored in each instruction location and room on the return address stack is required.

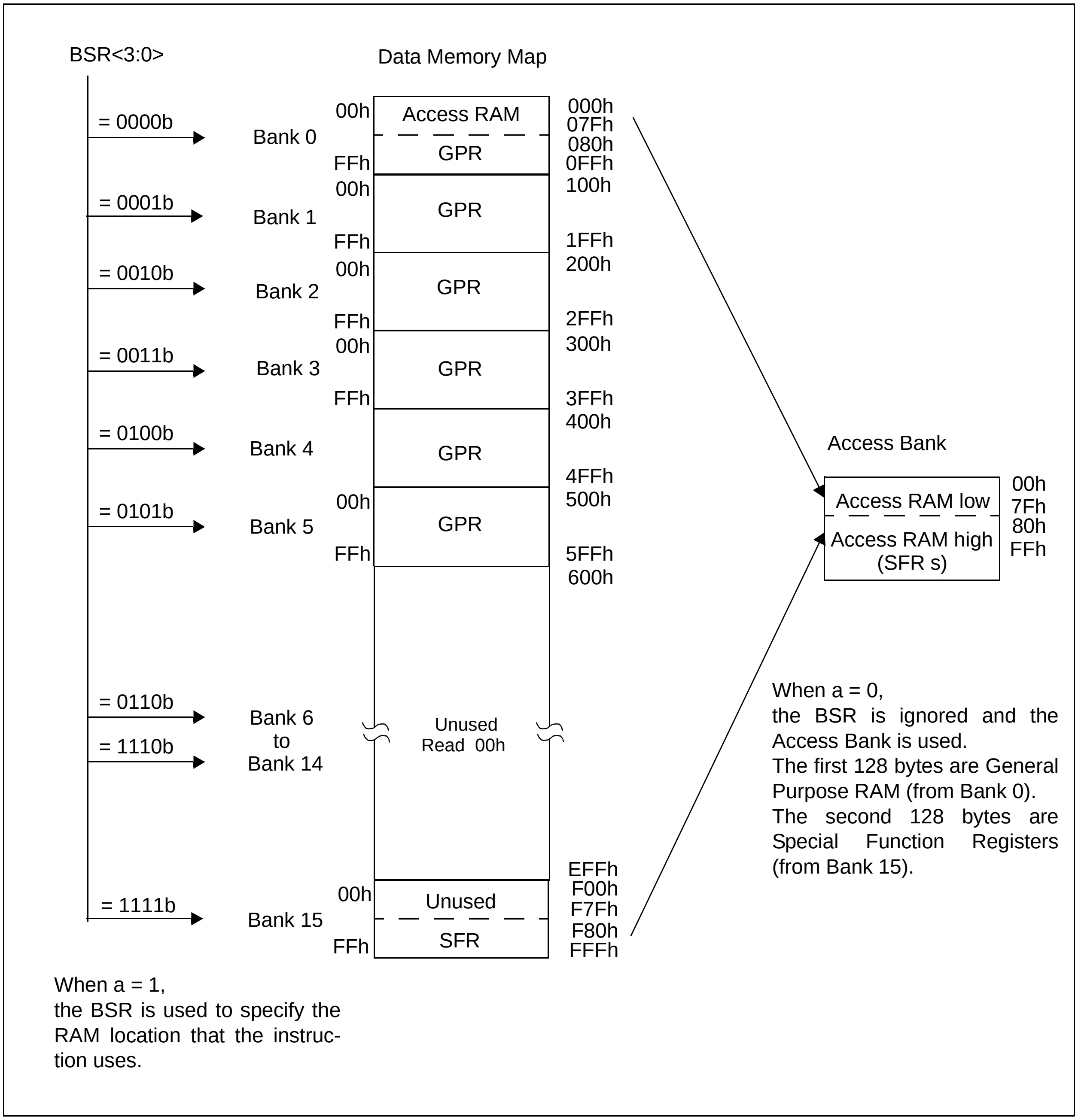
4.8.2 TABLE READS/TABLE WRITES

A better method of storing data in program memory allows 2 bytes of data to be stored in each instruction location.

Lookup table data may be stored 2 bytes per program word by using table reads and writes. The table pointer (TBLPTR) specifies the byte address and the table latch (TABLAT) contains the data that is read from, or written to program memory. Data is transferred to/from program memory one byte at a time.

A description of the Table Read/Table Write operation is shown in Section 5.0.

FIGURE 4-7: DATA MEMORY MAP FOR PIC18C252/452



4.13.1 RCON REGISTER

The Reset Control (RCON) register contains flag bits that allow differentiation between the sources of a device RESET. These flags include the $\overline{\text{TO}}$, $\overline{\text{PD}}$, $\overline{\text{POR}}$, $\overline{\text{BOR}}$ and $\overline{\text{RI}}$ bits. This register is readable and writable.

> Note 1: If the BOREN configuration bit is set (Brown-out Reset enabled), the $\overline{\text{BOR}}$ bit is '1' on a Power-on Reset. After a Brown-out Reset has occurred, the $\overline{\text{BOR}}$ bit will be clear and must be set by firmware to indicate the occurrence of the next Brown-out Reset.
> If the BOREN configuration bit is clear (Brown-out Reset disabled), $\overline{\text{BOR}}$ is unknown after Power-on Reset and

REGISTER 4-3: RCON REGISTER

7.1 INTCON Registers

The INTCON Registers are readable and writable registers, which contains various enable, priority, and flag bits.

REGISTER 7-1: INTCON REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
bit 7							bit 0

bit 7 GIE/GIEH: Global Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked interrupts
0 = Disables all interrupts
When IPEN = 1:
1 = Enables all high priority interrupts
0 = Disables all high priority interrupts

bit 6 PEIE/GIEL: Peripheral Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
When IPEN = 1:
1 = Enables all low priority peripheral interrupts
0 = Disables all low priority peripheral interrupts

bit 5 TMR0IE: TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 overflow interrupt
0 = Disables the TMR0 overflow interrupt

bit 4 INT0IE: INT0 External Interrupt Enable bit
1 = Enables the INT0 external interrupt
0 = Disables the INT0 external interrupt

bit 3 RBIE: RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 TMR0IF: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

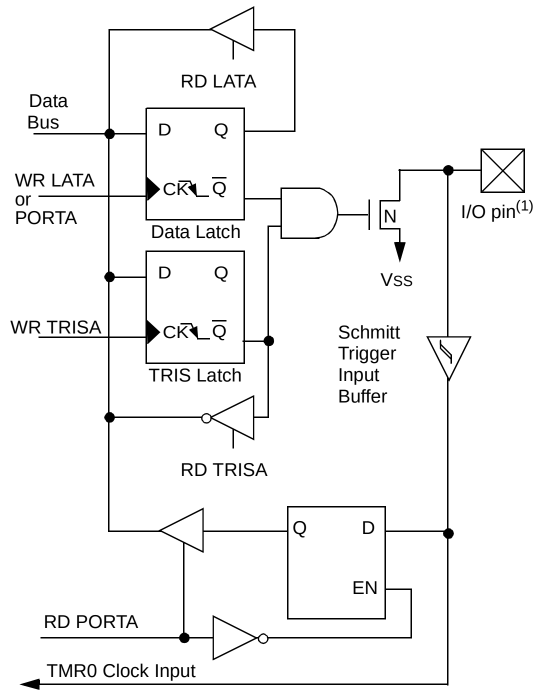
bit 1 INT0IF: INT0 External Interrupt Flag bit
1 = The INT0 external interrupt occurred (must be cleared in software)
0 = The INT0 external interrupt did not occur

bit 0 RBIF: RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

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

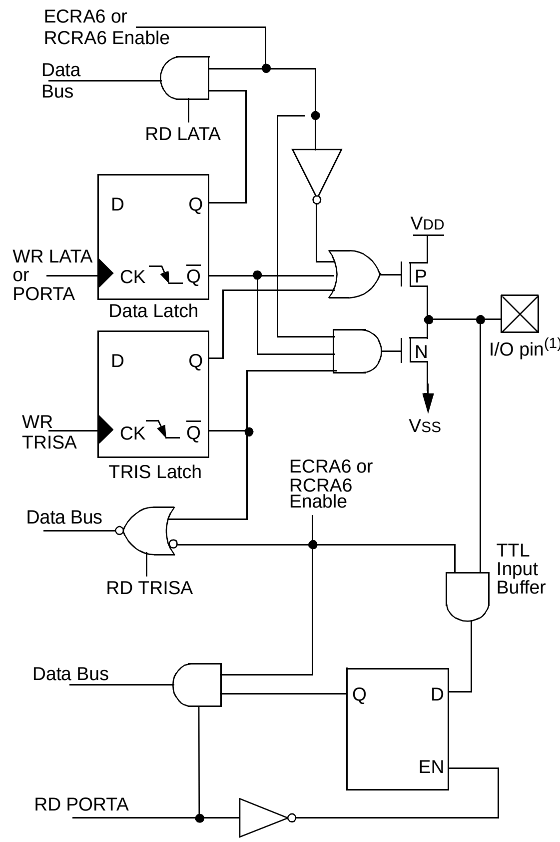
> **Note:** Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

FIGURE 8-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



Note 1: I/O pins have protection diodes to VDD and VSS.

FIGURE 8-3: BLOCK DIAGRAM OF RA6



Note 1: I/O pins have protection diodes to VDD and VSS.

TABLE 8-5: PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input.
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin, Timer1 oscillator input, or Capture2 input/ Compare2 output/PWM output when CCP2MX configuration bit is disabled.
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/ PWM1 output.
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I^2C modes.
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or Data I/O (I^2C mode).
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port Data output.
RC6/TX/CK	bit6	ST	Input/output port pin, Addressable USART Asynchronous Transmit, or Addressable USART Synchronous Clock.
RC7/RX/DT	bit7	ST	Input/output port pin, Addressable USART Asynchronous Receive, or Addressable USART Synchronous Data.

Legend: ST = Schmitt Trigger input

TABLE 8-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

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
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
LATC	LATC Data Output Register								xxxx xxxx	uuuu uuuu
TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged

8.6 Parallel Slave Port

The Parallel Slave Port is implemented on the 40-pin devices only (PIC18C4X2).

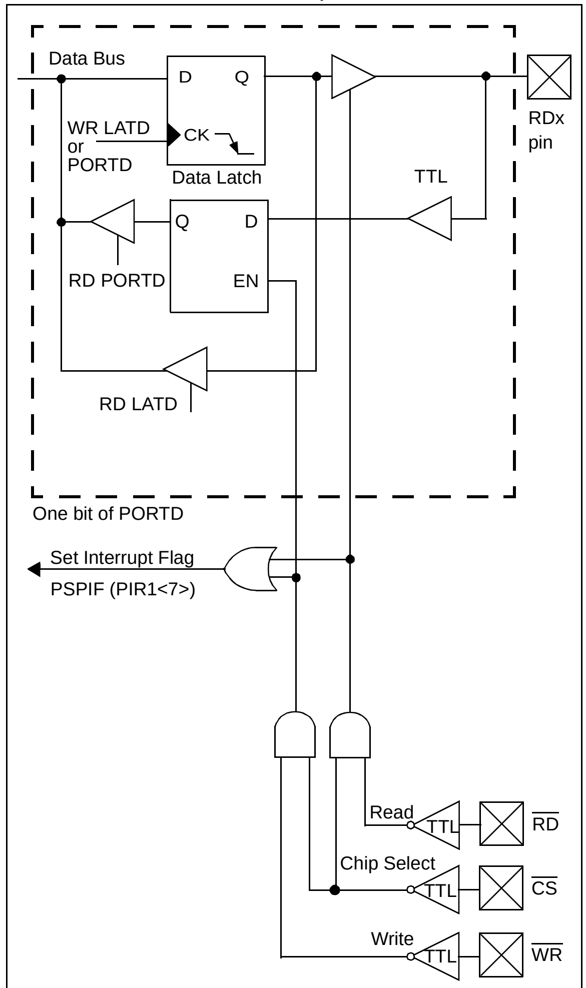
PORTD operates as an 8-bit wide, parallel slave port, or microprocessor port, when control bit PSPMODE (TRISE<4>) is set. It is asynchronously readable and writable by the external world through $\overline{RD}$ control input pin RE0/$\overline{RD}$ and $\overline{WR}$ control input pin RE1/$\overline{WR}$.

It can directly interface to an 8-bit microprocessor data bus. The external microprocessor can read or write the PORTD latch as an 8-bit latch. Setting bit PSPMODE enables port pin RE0/$\overline{RD}$ to be the $\overline{RD}$ input, RE1/$\overline{WR}$ to be the $\overline{WR}$ input and RE2/$\overline{CS}$ to be the $\overline{CS}$ (chip select) input. For this functionality, the corresponding data direction bits of the TRISE register (TRISE<2:0>) must be configured as inputs (set). The A/D port configuration bits PCFG2:PCFG0 (ADCON1<2:0>) must be set, which will configure pins RE2:RE0 as digital I/O.

A write to the PSP occurs when both the $\overline{CS}$ and $\overline{WR}$ lines are first detected low. A read from the PSP occurs when both the $\overline{CS}$ and $\overline{RD}$ lines are first detected low.

The PORTE I/O pins become control inputs for the microprocessor port when bit PSPMODE (TRISE<4>) is set. In this mode, the user must make sure that the TRISE<2:0> bits are set (pins are configured as digital inputs), and the ADCON1 is configured for digital I/O. In this mode, the input buffers are TTL.

FIGURE 8-10: PORTD AND PORTE BLOCK DIAGRAM (PARALLEL SLAVE PORT)



Note: I/O pin has protection diodes to VDD and VSS.

FIGURE 8-11: PARALLEL SLAVE PORT WRITE WAVEFORMS

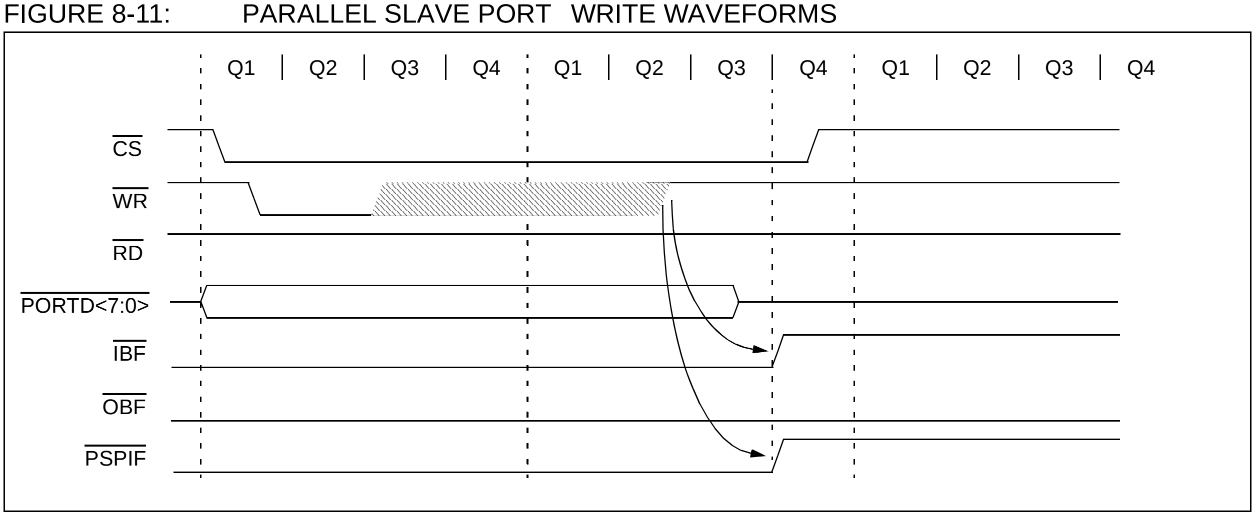


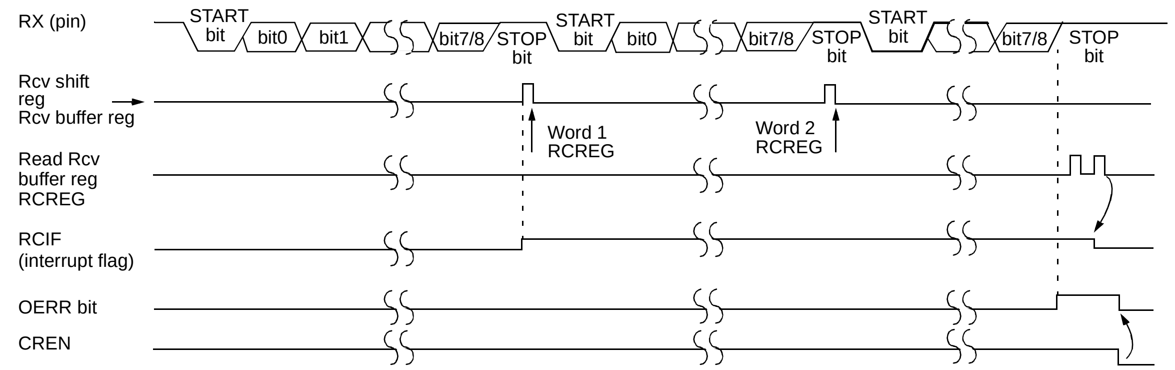
TABLE 15-3: BAUD RATES FOR SYNCHRONOUS MODE

BAUD RATE (K)	FOSC = 40 MHz			FOSC = 20 MHz			FOSC = 16 MHz			FOSC = 10 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	NA			NA			NA			NA		
1.2	NA			NA			NA			NA		
2.4	NA			NA			NA			NA		
9.6	NA			NA			NA			9.766	+1.73	255
19.2	NA			19.53	+1.73	255	19.23	+0.16	207	19.23	+0.16	129
76.8	76.92	0	129	76.92	+0.16	64	76.92	+0.16	51	75.76	-1.36	32
96	96.15	0	103	96.15	+0.16	51	95.24	-0.79	41	96.15	+0.16	25
300	303.03	-0.01	32	294.1	-1.96	16	307.69	+2.56	12	312.5	+4.17	7
500	500.00	0	19	500	0	9	500	0	7	500	0	4
HIGH	39.06		255	5000		0	4000		0	2500		0
LOW	10000.00		0	19.53		255	15.625		255	9.766		255

BAUD RATE (K)	FOSC = 7.15909 MHz			FOSC = 5.0688 MHz			FOSC = 4 MHz			FOSC = 3.579545 MHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	NA			NA			NA			NA		
1.2	NA			NA			NA			NA		
2.4	NA			NA			NA			NA		
9.6	9.622	+0.23	185	9.6	0	131	9.615	+0.16	103	9.622	+0.23	92
19.2	19.24	+0.23	92	19.2	0	65	19.231	+0.16	51	19.04	-0.83	46
76.8	77.82	+1.32	22	79.2	+3.13	15	76.923	+0.16	12	74.57	-2.90	11
96	94.20	-1.88	18	97.48	+1.54	12	1000	+4.17	9	99.43	+3.57	8
300	298.3	-0.57	5	316.8	+5.60	3	NA			298.3	-0.57	2
500	NA			NA			NA			NA		
HIGH	1789.8		0	1267		0	100		0	894.9		0
LOW	6.991		255	4.950		255	3.906		255	3.496		255

BAUD RATE (K)	FOSC = 1 MHz			FOSC = 32.768 kHz		
	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	NA			0.303	+1.14	26
1.2	1.202	+0.16	207	1.170	-2.48	6
2.4	2.404	+0.16	103	NA		
9.6	9.615	+0.16	25	NA		
19.2	19.24	+0.16	12	NA		
76.8	83.34	+8.51	2	NA		
96	NA			NA		
300	NA			NA		
500	NA			NA		
HIGH	250		0	8.192		0
LOW	0.9766		255	0.032		255

FIGURE 15-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

TABLE 15-7: REGISTERS ASSOCIATED WI TH ASYNCHRONOUS RECEPTION

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
INTCON	GIE/GIEH	PEIE/ GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP(1)	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 -00x	0000 -00x
RCREG	USART Receive Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC		BRGH	TRMT	TX9D	0000 -010	0000 -010
SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'.
Shaded cells are not used for Asynchronous Reception.

Note 1: The PSPIF, PSPIE and PSPIP bits are reserved on the PIC18C2X2 devices. Always maintain these bits clear.

19.1 Instruction Set

ADDLW ADD literal to WREG

Syntax:	[label] ADDLW k
Operands:	0 dk d255
Operation:	(WREG) + k o WREG
Status Affected:	N,OV, C, DC, Z

Encoding:

0000	1111	kkkk	kkkk

Description: The contents of WREG are added to the 8-bit literal 'k' and the result is placed in WREG.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to WREG

Example: ADDLW 0x15

Before Instruction

WREG = 0x10

After Instruction

WREG = 0x25

ADDWF ADD WREG to f

Syntax:	[label] ADDWF f [,d [,a] f [,d [,a]
Operands:	0 df d255 d [0,1] a [0,1]
Operation:	(WREG) + (f) o dest
Status Affected:	N,OV, C, DC, Z

Encoding:

0010	01da	ffff	ffff

Description: Add WREG to register 'f'. If 'd' is 0, the result is stored in WREG. If 'd' is 1, the result is stored back in register 'f' (default). If a is 0, the Access Bank will be selected. If a is 1, the BSR is used.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: ADDWF REG, 0, 0

Before Instruction

WREG = 0x17
REG = 0xC2

After Instruction

WREG = 0xD9
REG = 0xC2

RLNCF — Rotate Left f (no carry)

Syntax: [label] RLNCF f [,d [,a]

Operands: 0 df d255
d [0,1]
a [0,1]

Operation: (f<n>) o dest<n+1>,
(f<7>) o dest<0>

Status Affected: N,Z

Encoding:

0100	01da	ffff	ffff

Description: The contents of register 'f' are rotated one bit to the left. If 'd' is 0, the result is placed in WREG. If 'd' is 1, the result is stored back in register 'f' (default). If a is 0, the Access Bank will be selected, overriding the BSR value. If a is 1, then the bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: RLNCF REG, 1, 0

Before Instruction
REG = 1010 1011

After Instruction
REG = 0101 0111

RRCF — Rotate Right f through Carry

Syntax: [label] RRCF f [,d [,a]

Operands: 0 df d255
d [0,1]
a [0,1]

Operation: (f<n>) o dest<n-1>,
(f<0>) o C,
(C) o dest<7>

Status Affected: C,N,Z

Encoding:

0011	00da	ffff	ffff

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 WREG. If 'd' is 1, the result is placed back in register 'f' (default). If a is 0, the Access Bank will be selected, overriding the BSR value. If a is 1, then the bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: RRCF REG, 0, 0

Before Instruction
REG = 1110 0110
C = 0

After Instruction
REG = 1110 0110
WREG = 0111 0011
C = 0

TBLRD Table Read

Syntax: [label] TBLRD (*; *+; *-; +*)

Operands: None

Operation:
if TBLRD *,
(Prog Mem (TBLPTR)) o TABLAT;
TBLPTR - No Change;
if TBLRD *+,
(Prog Mem (TBLPTR)) o TABLAT;
(TBLPTR) +1 o TBLPTR;
if TBLRD *-,
(Prog Mem (TBLPTR)) o TABLAT;
(TBLPTR) -1 o TBLPTR;
if TBLRD +*,
(TBLPTR) +1 o TBLPTR;
(Prog Mem (TBLPTR)) o TABLAT;

Status Affected: None

Encoding:

0000	0000	0000	10nn nn=0 * =1 *+ =2 *- =3 +*

Description: This instruction is used to read the contents of Program Memory (P.M.). To address the program memory, a pointer called Table Pointer (TBLPTR) is used.
The TBLPTR (a 21-bit pointer) points to each byte in the program memory. TBLPTR has a 2 Mbyte address range.

TBLPTR[0] = 0: Least Significant Byte of Program Memory Word

TBLPTR[0] = 1: Most Significant Byte of Program Memory Word

The TBLRDinstruction can modify the value of TBLPTR as follows:

- no change
- post-increment
- post-decrement
- pre-increment

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
No operation	No operation (Read Program Memory)	No operation	No operation (Write TABLAT)

TBLRD Table Read (cont'd)

Example 1: TBLRD *+ ;

Before Instruction

TABLAT	=	0x55
TBLPTR	=	0x00A356
MEMORY(0x00A356)	=	0x34

After Instruction

TABLAT	=	0x34
TBLPTR	=	0x00A357

Example 2: TBLRD +* ;

Before Instruction

TABLAT	=	0xAA
TBLPTR	=	0x01A357
MEMORY(0x01A357)	=	0x12
MEMORY(0x01A358)	=	0x34

After Instruction

TABLAT	=	0x34
TBLPTR	=	0x01A358

20.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

The PICDEM 3 demonstration board is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with an LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 3 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer with an adapter socket, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 3 demonstration board to test firmware. A prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM 3 demonstration board is a LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM 3 demonstration board provides an additional RS-232 interface and Windows software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

20.14 PICDEM 17 Demonstration Board

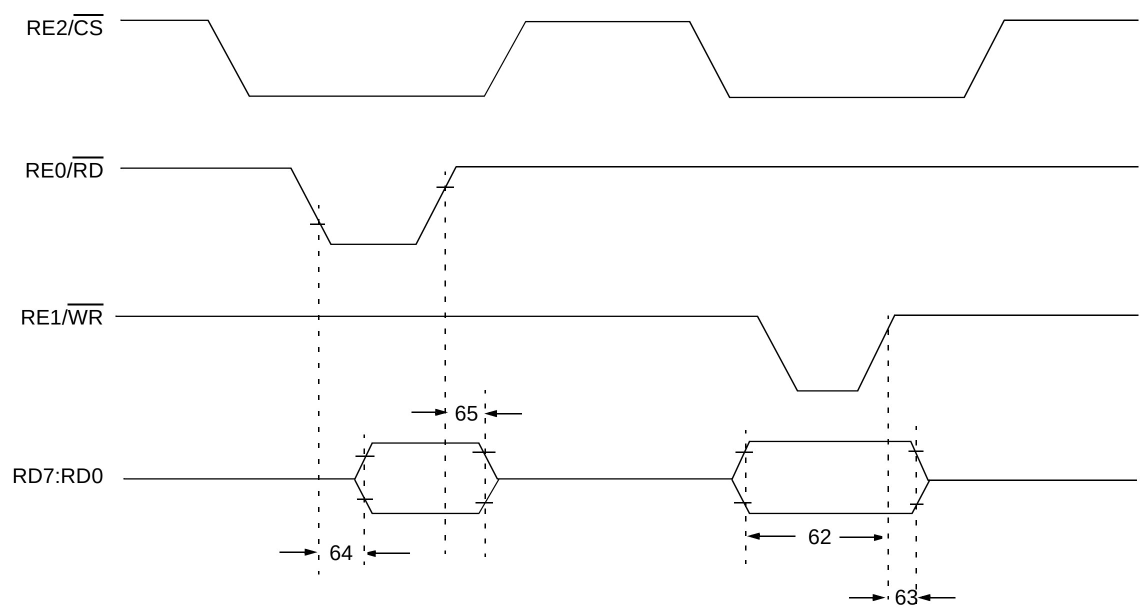
The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included and the user may erase it and program it with the other sample programs using the PRO MATE II device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the PICMASTER emulator and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

20.15 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip s HCS Secure Data Products. The HCS evaluation kit includes a LCD display to show changing codes, a decoder to decode transmissions and a programming interface to program test transmitters.

NOTES:

FIGURE 21-11: PARALLEL SLAVE PORT TIMING (PIC18C4X2)



Note: Refer to Figure 21-4 for load conditions.

TABLE 21-10: PARALLEL SLAVE PORT REQUIREMENTS (PIC18C4X2)

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
62	TdtV2wrH	Data in valid before $\overline{WR}$ nor $\overline{CS}$ n (setup time)		20 25		ns ns	 Extended Temp. Range
63	TwrH2dtI	$\overline{WR}$ nor $\overline{CS}$ nto data in invalid (hold time)	PIC18CXXX	20		ns	
			PIC18LCXXX	35		ns	
64	TrdL2dtV	$\overline{RD}$ pand $\overline{CS}$ pto data out valid			80 90	ns ns	 Extended Temp. Range
65	TrdH2dtI	$\overline{RD}$ nor $\overline{CS}$ nto data out invalid		10	30	ns	
66	TibfINH	Inhibit of the IBF flag bit being cleared from $\overline{WR}$ nor $\overline{CS}$ n			$3T_{CY}$		

FIGURE 22-7: TYPICAL IDD vs. FOSC OVER VDD (LP MODE)

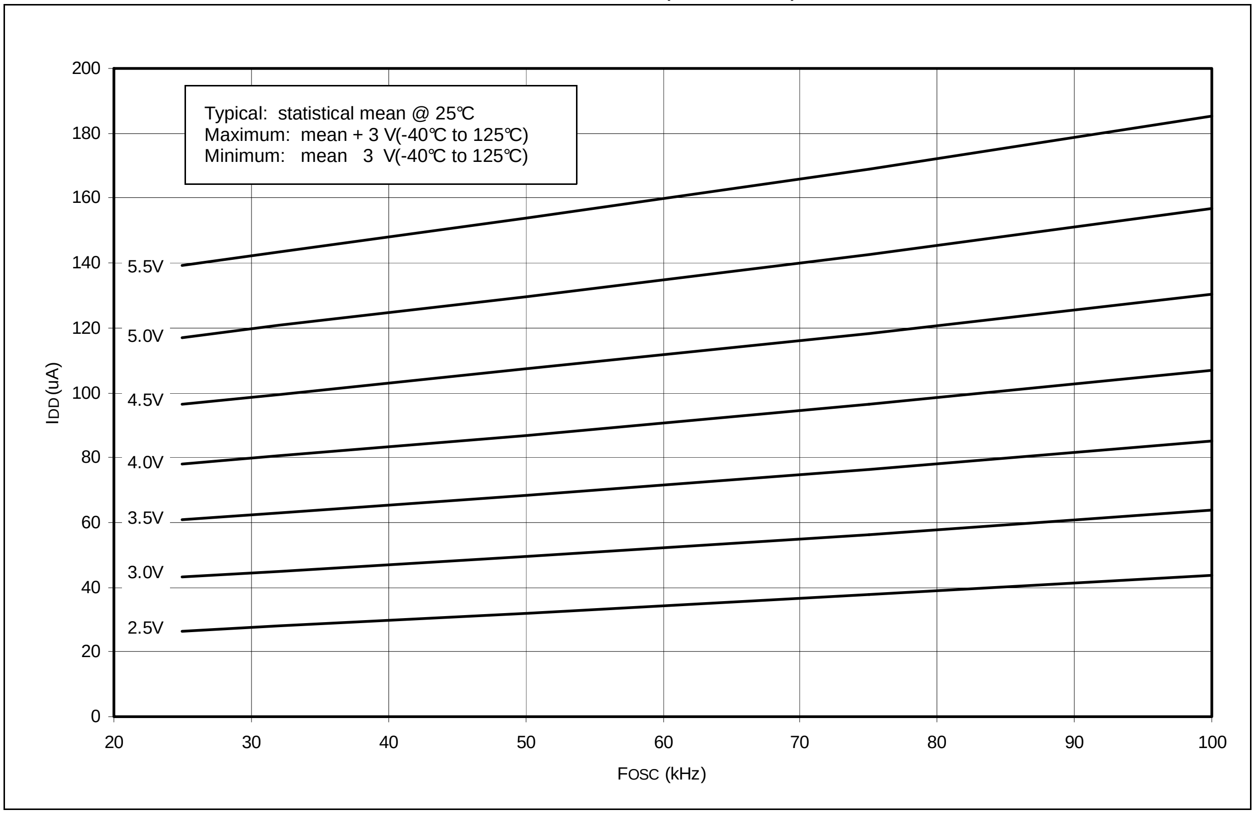


FIGURE 22-8: MAXIMUM IDD vs. FOSC OVER VDD (LP MODE)

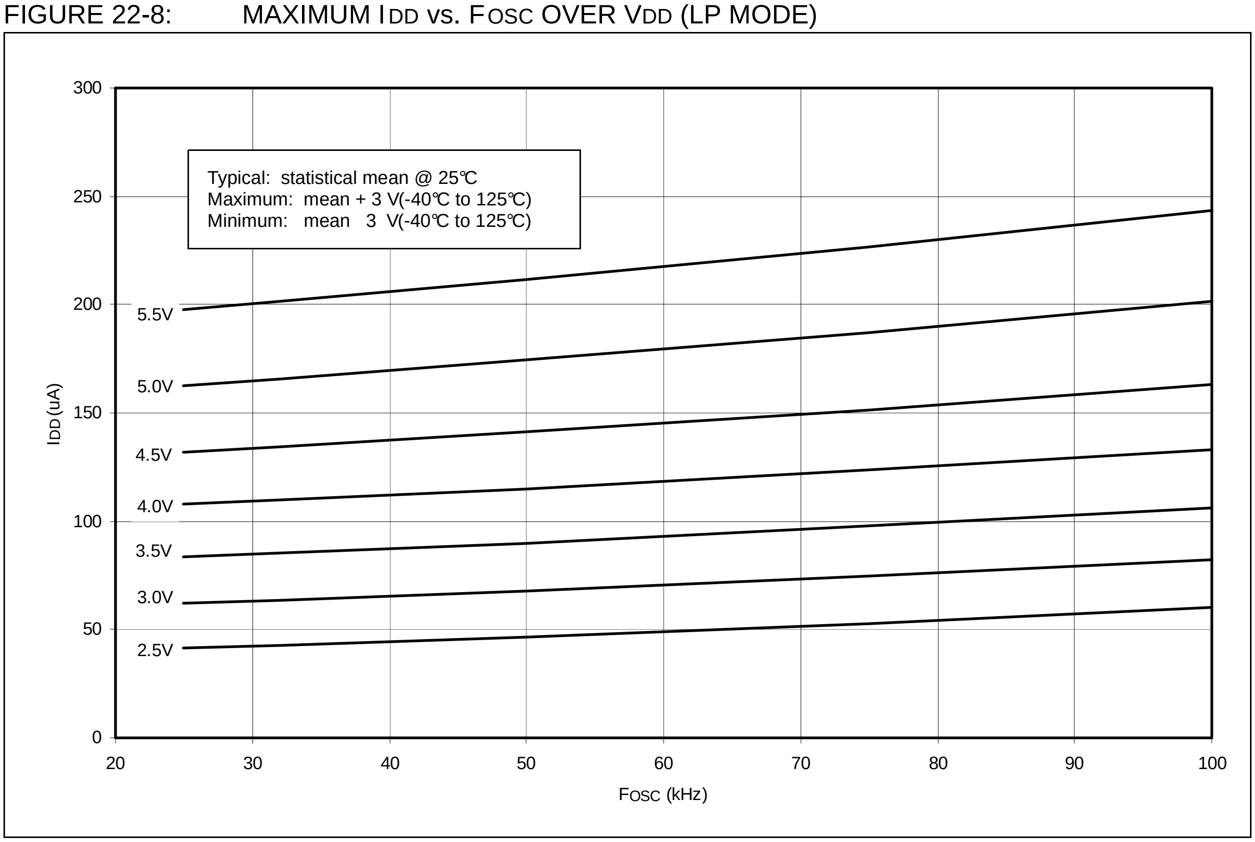


FIGURE 22-9: TYPICAL AND MAXIMUM I_{DD} vs. V_{DD} (TIMER1 AS MAIN OSCILLATOR, 32.768 kHz, C = 47 pF)

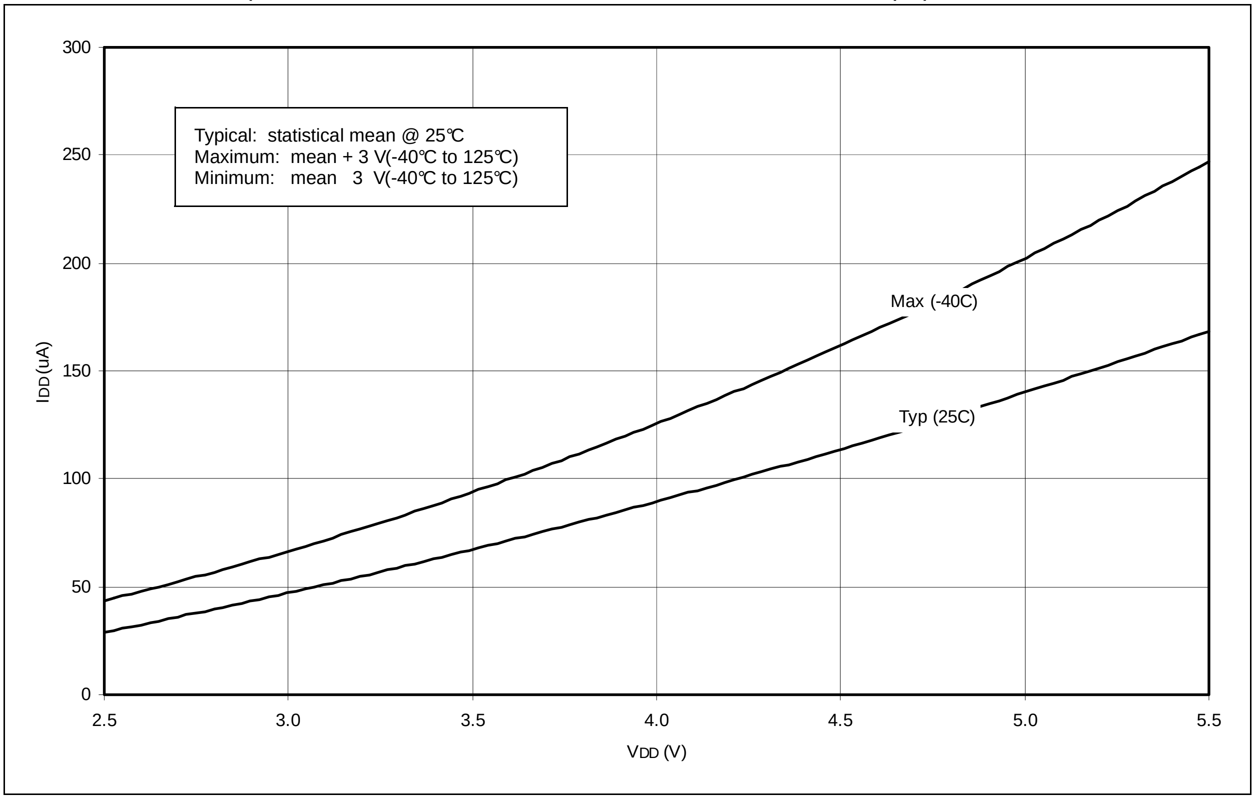


FIGURE 22-10: AVERAGE F_{OSC} vs. V_{DD} FOR VARIOUS VALUES OF R (RC MODE, C = 20 pF, 25°C)

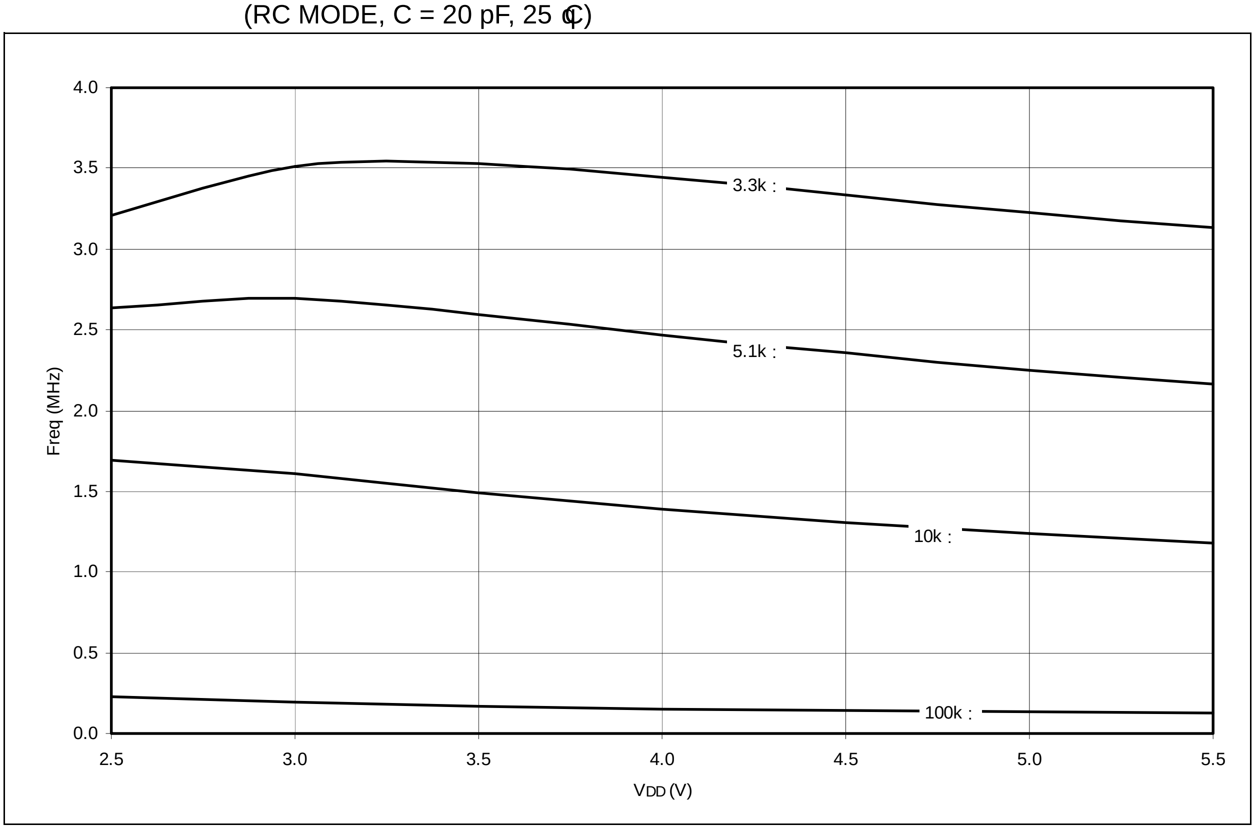


FIGURE 22-13: IPD vs. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED)

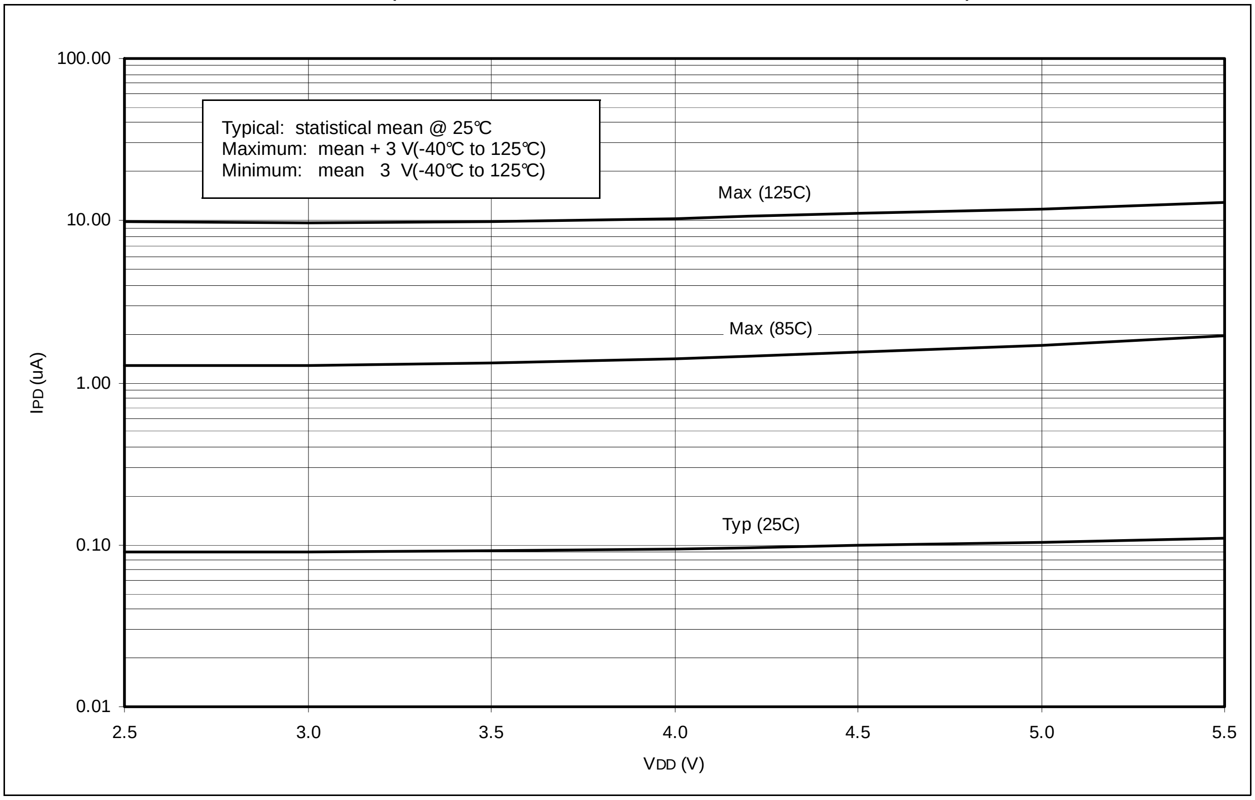
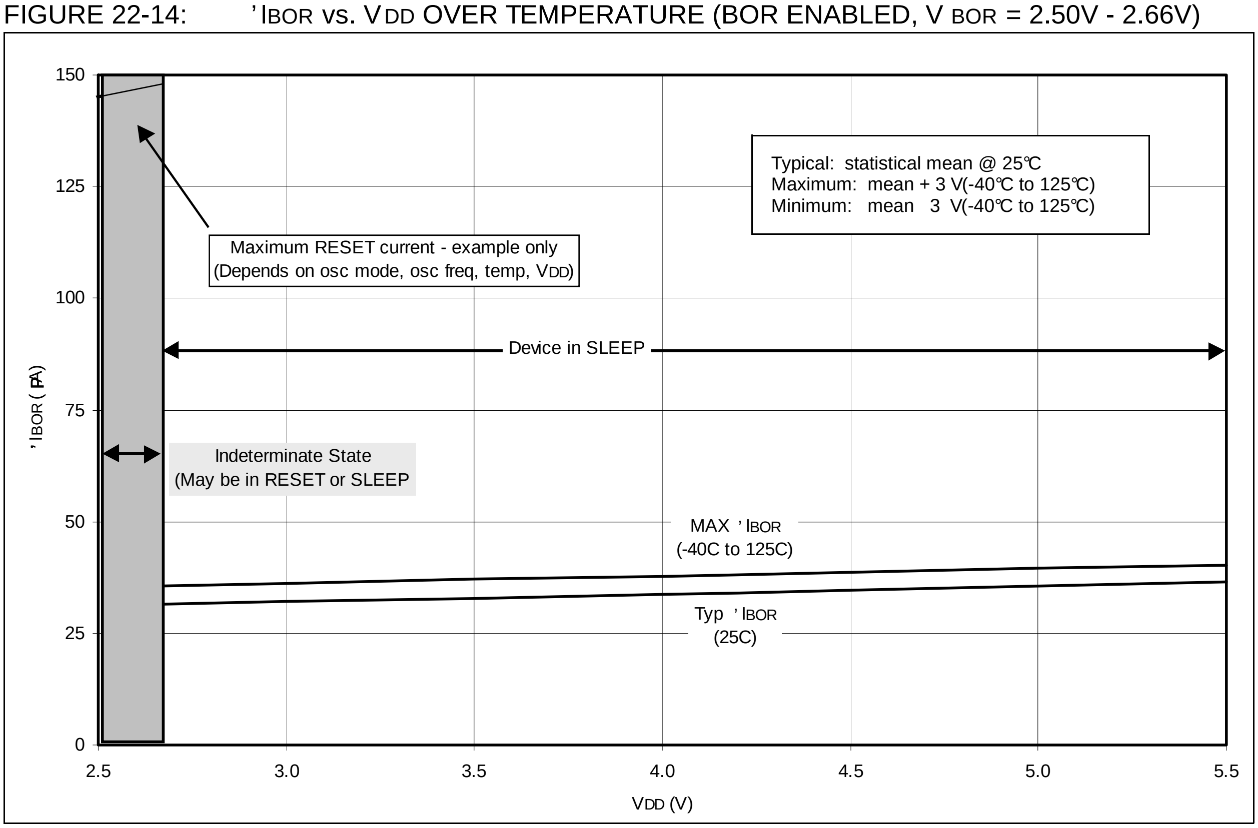
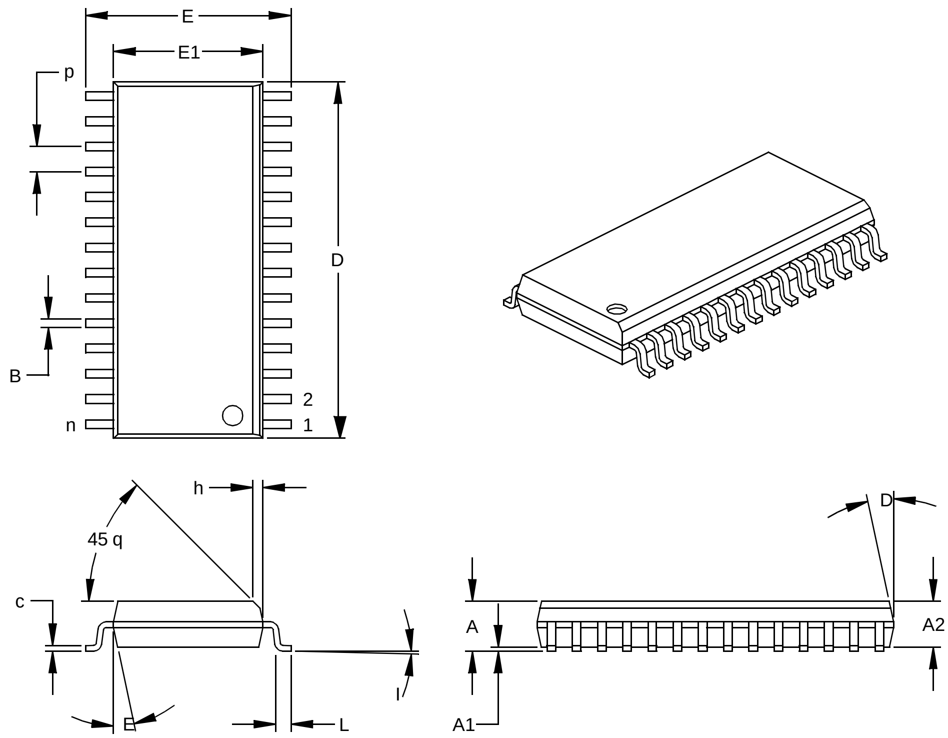


FIGURE 22-14: ’IBOR vs. VDD OVER TEMPERATURE (BOR ENABLED, VBOR = 2.50V - 2.66V)



28-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)

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



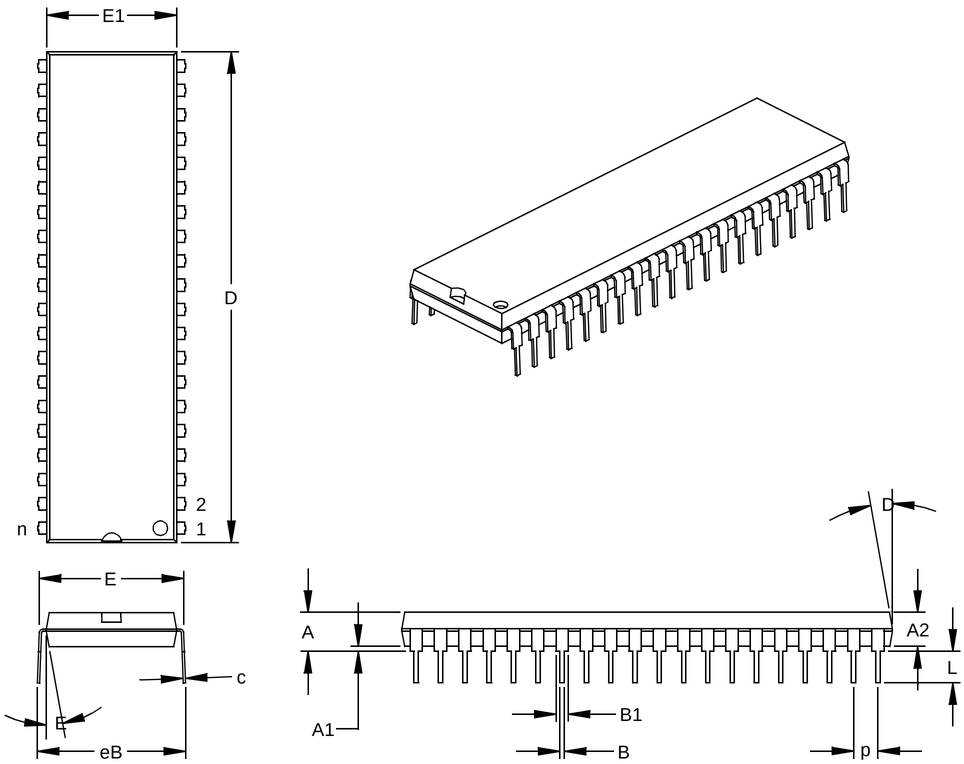
	Units	INCHES*			MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		28			28	
Pitch	p		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff §	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.288	.295	.299	7.32	7.49	7.59
Overall Length	D	.695	.704	.712	17.65	17.87	18.08
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle Top	I	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.013	0.23	0.28	0.33
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	D	0	12	15	0	12	15
Mold Draft Angle Bottom	E	0	12	15	0	12	15

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010 (0.254mm) per side.
JEDEC Equivalent: MS-013
Drawing No. C04-052

40-Lead Plastic Dual In-line (P) – 600 mil (PDIP)

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



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		40			40	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.160	.175	.190	4.06	4.45	4.83
Molded Package Thickness	A2	.140	.150	.160	3.56	3.81	4.06
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.595	.600	.625	15.11	15.24	15.88
Molded Package Width	E1	.530	.545	.560	13.46	13.84	14.22
Overall Length	D	2.045	2.058	2.065	51.94	52.26	52.45
Tip to Seating Plane	L	.120	.130	.135	3.05	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.030	.050	.070	0.76	1.27	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing §	eB	.620	.650	.680	15.75	16.51	17.27
Mold Draft Angle Top	D	5	10	15	5	10	15
Mold Draft Angle Bottom	E	5	10	15	5	10	15

* Controlling Parameter
§ Significant Characteristic
Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010 (0.254mm) per side.
JEDEC Equivalent: MO-011
Drawing No. C04-016