

# engrcs PICmicro User Guide

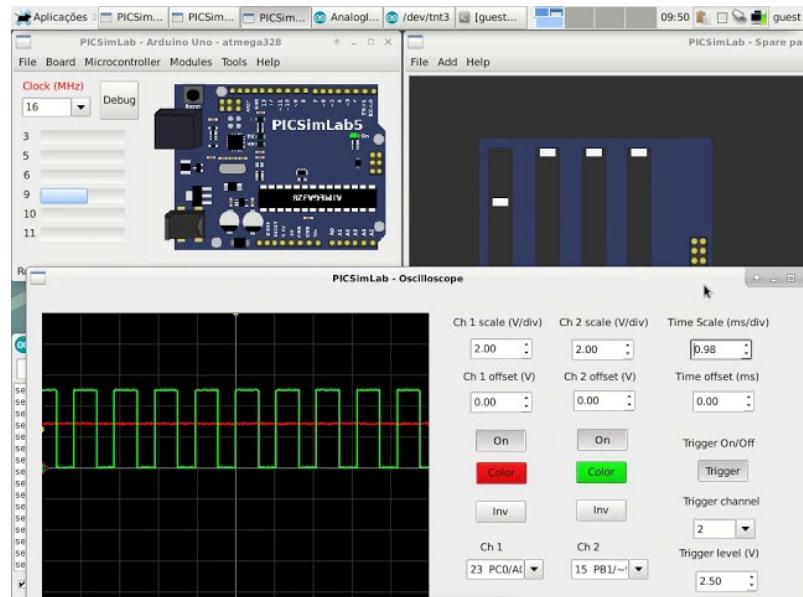
[Home](#) » [engrcs](#) » engrcs PICmicro User Guide 

## Contents

- [1 engrcs PICmicro](#)
- [2 Pin Layout](#)
- [3 Program Memory Layout](#)
- [4 Data Memory Layout “General Purpose Registers \(GPR\)” Direct Addressing](#)
- [5 PICmicro Instruction Set Summary 1/3](#)
- [6 PICmicro Instruction Set Summary 2/3](#)
- [7 PICmicro Instruction Set Summary 3/3](#)
- [8 Special Function Register \(SFR\) Map 1/3](#)
- [9 Special Function Register Map 2/3](#)
- [10 Special Function Register Map 3/3](#)
- [11 Documents / Resources](#)
- [12 Related Posts](#)

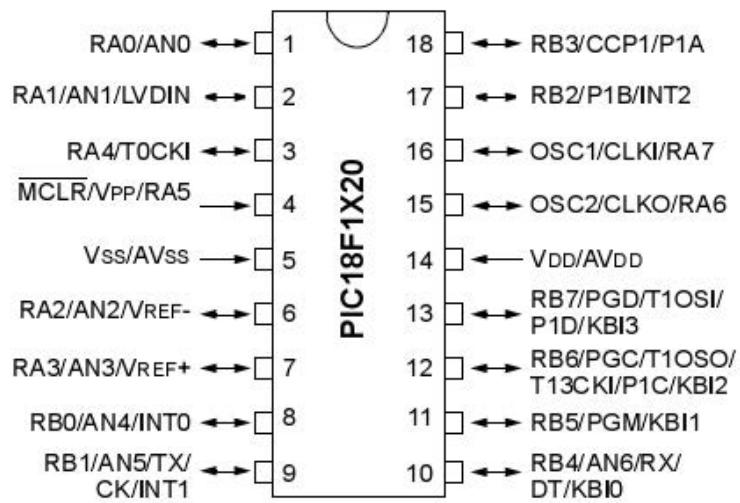
**engrcs**

## engrcs PICmicro



## Pin Layout

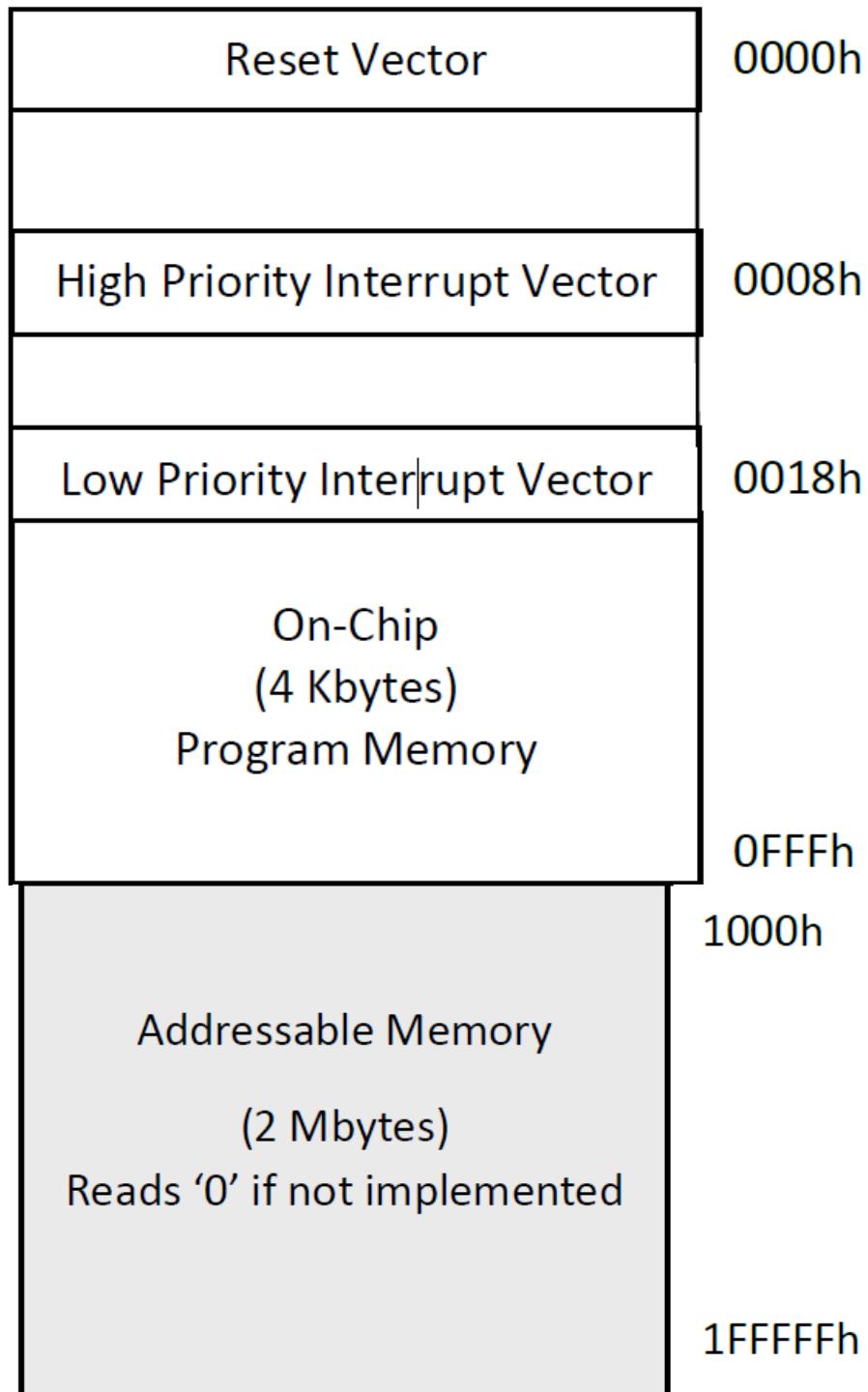
18-Pin PDIP, SOIC



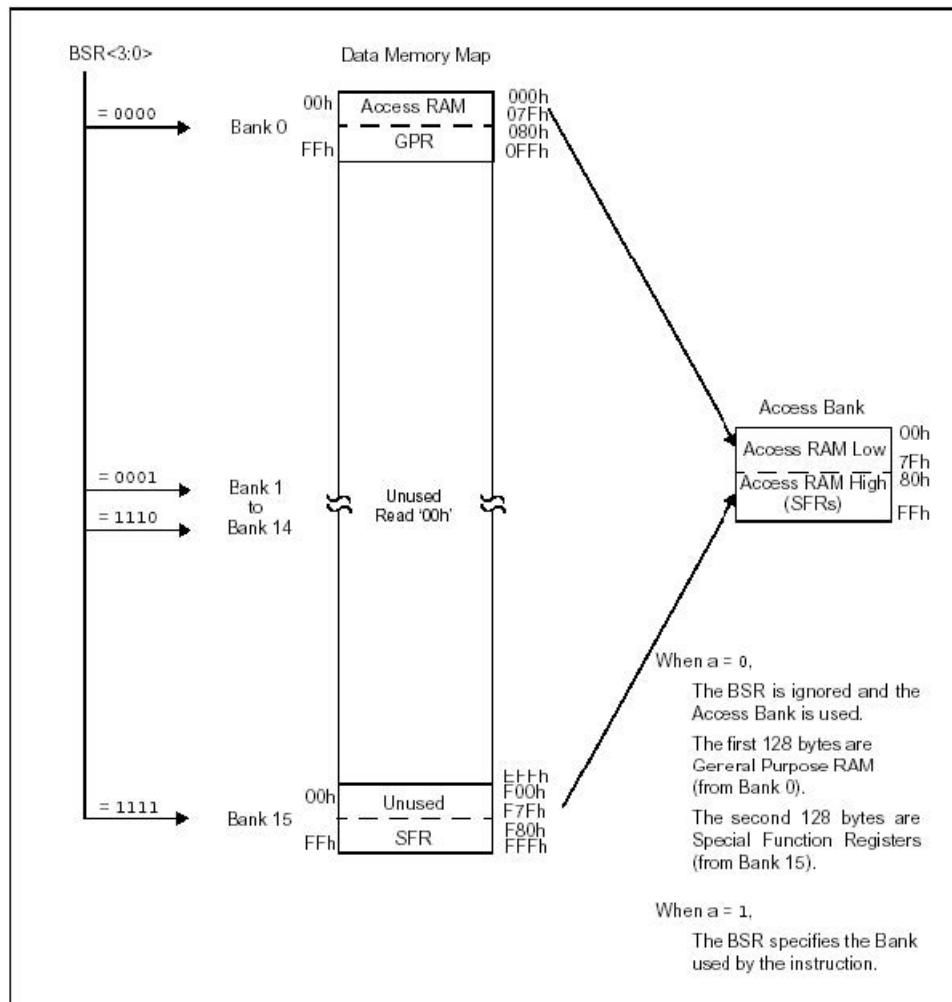
The two pins whose definition is constant are pins 5 and 14, which are ground and power.

- Pin 5 Ground (0 V)
- Pin 14 Power (2 to 5.5 V)

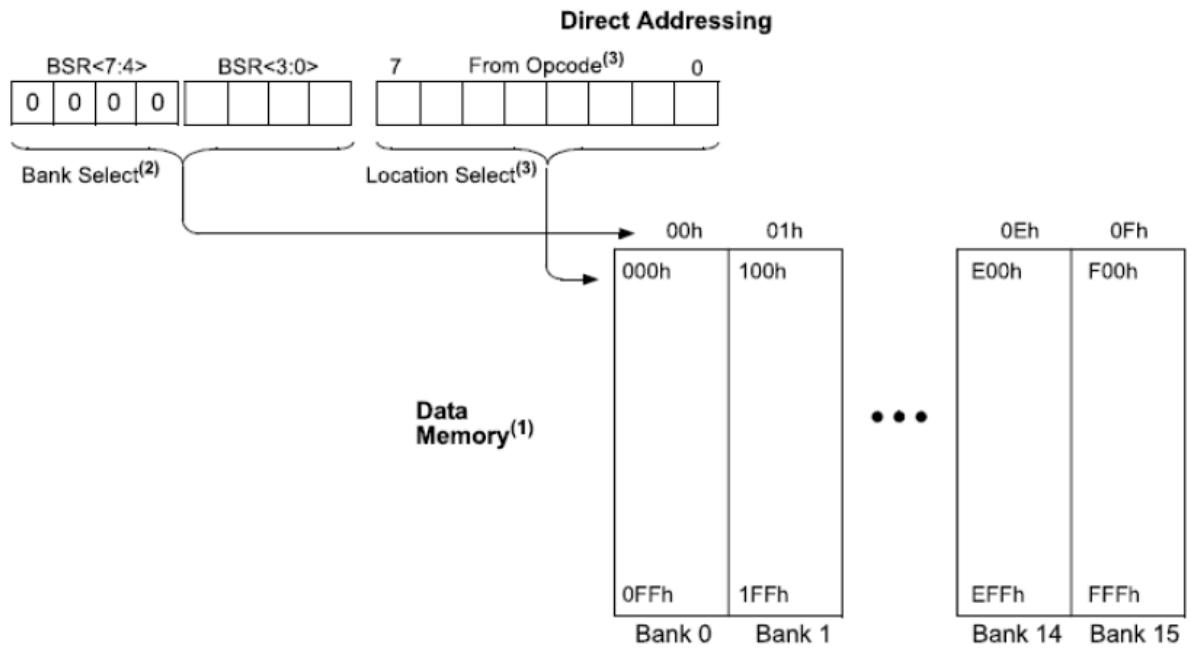
## Program Memory Layout



## Data Memory Layout “General Purpose Registers (GPR)”



## Direct Addressing



## PICmicro Instruction Set Summary 1/3

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb	Lsb				
<b>BYTE-ORIENTED FILE REGISTER OPERATIONS</b>								
ADDWF f, d, a	Add WREG and f	1	0010 01da	ffff	ffff	C, DC, Z, OV, N	1, 2	
ADDWFC f, d, a	Add WREG and Carry bit to f	1	0010 00da	ffff	ffff	C, DC, Z, OV, N	1, 2	
ANDWF f, d, a	AND WREG with f	1	0001 01da	ffff	ffff	Z, N	1, 2	
CLRF f, a	Clear f	1	0110 101a	ffff	ffff	Z	2	
COMF f, d, a	Complement f	1	0001 11da	ffff	ffff	Z, N	1, 2	
CPFSEQ f, a	Compare f with WREG, skip =	1 (2 or 3)	0110 001a	ffff	ffff	None	4	
CPFGT f, a	Compare f with WREG, skip >	1 (2 or 3)	0110 010a	ffff	ffff	None	4	
CPFSLT f, a	Compare f with WREG, skip <	1 (2 or 3)	0110 000a	ffff	ffff	None	1, 2	
DECf f, d, a	Decrement f	1	0000 01da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3, 4	
DECFSZ f, d, a	Decrement f, Skip if 0	1 (2 or 3)	0010 11da	ffff	ffff	None	1, 2, 3, 4	
DCFSNZ f, d, a	Decrement f, Skip if Not 0	1 (2 or 3)	0100 11da	ffff	ffff	None	1, 2	
INCF f, d, a	Increment f	1	0010 10da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3, 4	
INCFSZ f, d, a	Increment f, Skip if 0	1 (2 or 3)	0011 11da	ffff	ffff	None	4	
INFSNZ f, d, a	Increment f, Skip if Not 0	1 (2 or 3)	0100 10da	ffff	ffff	None	1, 2	
IORWF f, d, a	Inclusive OR WREG with f	1	0001 00da	ffff	ffff	Z, N	1, 2	
MOVf f, d, a	Move f	1	0101 00da	ffff	ffff	Z, N	1	
MOVFF f <sub>s</sub> , f <sub>d</sub>	Move f <sub>s</sub> (source) to 1st word f <sub>d</sub> (destination) 2nd word	2	1100 ffff	ffff	ffff	None		
			1111 ffff	ffff	ffff			
MOVWF f, a	Move WREG to f	1	0110 111a	ffff	ffff	None		
MULWF f, a	Multiply WREG with f	1	0000 001a	ffff	ffff	None		
NEGF f, a	Negate f	1	0110 110a	ffff	ffff	C, DC, Z, OV, N	1, 2	
RLCF f, d, a	Rotate Left f through Carry	1	0011 01da	ffff	ffff	C, Z, N		
RLNCF f, d, a	Rotate Left f (No Carry)	1	0100 01da	ffff	ffff	Z, N	1, 2	
RRCF f, d, a	Rotate Right f through Carry	1	0011 00da	ffff	ffff	C, Z, N		
RRNCF f, d, a	Rotate Right f (No Carry)	1	0100 00da	ffff	ffff	Z, N		
SETF f, a	Set f	1	0110 100a	ffff	ffff	None		
SUBFWB f, d, a	Subtract f from WREG with borrow	1	0101 01da	ffff	ffff	C, DC, Z, OV, N	1, 2	
SUBWF f, d, a	Subtract WREG from f	1	0101 11da	ffff	ffff	C, DC, Z, OV, N		
SUBWFB f, d, a	Subtract WREG from f with borrow	1	0101 10da	ffff	ffff	C, DC, Z, OV, N	1, 2	
SWAPF f, d, a	Swap nibbles in f	1	0011 10da	ffff	ffff	None	4	
TSTFSZ f, a	Test f, skip if 0	1 (2 or 3)	0110 011a	ffff	ffff	None	1, 2	
XORWF f, d, a	Exclusive OR WREG with f	1	0001 10da	ffff	ffff	Z, N		
<b>BIT-ORIENTED FILE REGISTER OPERATIONS</b>								
BCF f, b, a	Bit Clear f	1	1001 bbba	ffff	ffff	None	1, 2	
BSF f, b, a	Bit Set f	1	1000 bbba	ffff	ffff	None	1, 2	
BTFSZ f, b, a	Bit Test f, Skip if Clear	1 (2 or 3)	1011 bbba	ffff	ffff	None	3, 4	
BTFSZ f, b, a	Bit Test f, Skip if Set	1 (2 or 3)	1010 bbba	ffff	ffff	None	3, 4	
BTG f, d, a	Bit Toggle f	1	0111 bbba	ffff	ffff	None	1, 2	

### Note

- When a Port register is modified as a function of itself (e.g., MOVF PORTB, 1, 0), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- If this instruction is executed on the TMRO register (and where applicable, d = 1), the prescaler will be cleared if assigned.
- If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.
- Some instructions are 2-word instructions. The second word of these instructions will be executed as a Nop, unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

5. If the table write starts the write cycle to internal memory, the write will continue until terminated.

### PICmicro Instruction Set Summary 2/3

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes	
			MSb	LSb					
<b>LITERAL OPERATIONS</b>									
ADDLW k	Add literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N		
ANDLW k	AND literal with WREG	1	0000	1011	kkkk	kkkk	Z, N		
IORLW k	Inclusive OR literal with WREG	1	0000	1001	kkkk	kkkk	Z, N		
LFSR f, k	Move literal (12-bit) 2nd word to FSRx 1st word	2	1110	1110	00ff	kkkk	None		
MOVLB k	Move literal to BSR<3:0>	1	0000	0001	0000	kkkk	None		
MOVlw k	Move literal to WREG	1	0000	1110	kkkk	kkkk	None		
MULLW k	Multiply literal with WREG	1	0000	1101	kkkk	kkkk	None		
RETLW k	Return with literal in WREG	2	0000	1100	kkkk	kkkk	None		
SUBLW k	Subtract WREG from literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N		
XORLW k	Exclusive OR literal with WREG	1	0000	1010	kkkk	kkkk	Z, N		
<b>DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS</b>									
TBLRD*	Table read	2	0000	0000	0000	1000	None		
TBLRD*+	Table read with post-increment		0000	0000	0000	1001	None		
TBLRD*-	Table read with post-decrement		0000	0000	0000	1010	None		
TBLRD**	Table read with pre-increment		0000	0000	0000	1011	None		
TBLWT*	Table write	2 (5)	0000	0000	0000	1100	None		
TBLWT*+	Table write with post-increment		0000	0000	0000	1101	None		
TBLWT*-	Table write with post-decrement		0000	0000	0000	1110	None		
TBLWT**	Table write with pre-increment		0000	0000	0000	1111	None		

#### Note

- When a Port register is modified as a function of itself (e.g., MOVF PORTB, 1, 0), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
- If this instruction is executed on the TMRO register (and where applicable, d =1), the prescaler will be cleared if assigned.
- If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP

Some instructions are 2-word instructions. The second word of these instructions will be executed as a NOP, unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

- If the table write starts the write cycle to internal memory, the write will continue until terminated.

## PICmicro Instruction Set Summary 3/3

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb	LSb				
<b>CONTROL OPERATIONS</b>								
BC n	Branch if Carry	1 (2)	1110 0010	nnnn nnnn	nnnn	nnnn	None	
BN n	Branch if Negative	1 (2)	1110 0110	nnnn nnnn	nnnn	nnnn	None	
BNC n	Branch if Not Carry	1 (2)	1110 0011	nnnn nnnn	nnnn	nnnn	None	
BNN n	Branch if Not Negative	1 (2)	1110 0111	nnnn nnnn	nnnn	nnnn	None	
BNOV n	Branch if Not Overflow	1 (2)	1110 0101	nnnn nnnn	nnnn	nnnn	None	
BNZ n	Branch if Not Zero	1 (2)	1110 0001	nnnn nnnn	nnnn	nnnn	None	
BOV n	Branch if Overflow	1 (2)	1110 0100	nnnn nnnn	nnnn	nnnn	None	
BRA n	Branch Unconditionally	2	1101 0nnn	nnnn nnnn	nnnn	nnnn	None	
BZ n	Branch if Zero	1 (2)	1110 0000	nnnn nnnn	nnnn	nnnn	None	
CALL n, s	Call subroutine 1st word	2	1110 110s	kkkk kkkk	kkkk	kkkk	None	
	2nd word		1111 kkkk	kkkk kkkk	kkkk	kkkk		
CLRWDT —	Clear Watchdog Timer	1	0000 0000	0000 0100	0000	0100	TO, PD	
DAW —	Decimal Adjust WREG	1	0000 0000	0000 0111	0000	0111	C	
GOTO n	Go to address 1st word	2	1110 1111	kkkk kkkk	kkkk	kkkk	None	
	2nd word		1111 kkkk	kkkk kkkk	kkkk	kkkk		
NOP —	No Operation	1	0000 0000	0000 0000	0000	0000	None	
NOP —	No Operation	1	1111 xxxx	xxxx xxxx	xxxx	xxxx	None	4
POP —	Pop top of return stack (TOS)	1	0000 0000	0000 0110	0000	0110	None	
PUSH —	Push top of return stack (TOS)	1	0000 0000	0000 0101	0000	0101	None	
RCALL n	Relative Call	2	1101 1nnn	nnnn nnnn	nnnn	nnnn	None	
RESET	Software device Reset	1	0000 0000	1111 1111	1111	1111	All	
RETFIE s	Return from interrupt enable	2	0000 0000	0001 000s	0001	000s	GIE/GIEH, PEIE/GIEL	
RETLW k	Return with literal in WREG	2	0000 1100	kkkk kkkk	kkkk	kkkk	None	
RETURN s	Return from Subroutine	2	0000 0000	0001 001s	0001	001s	None	
SLEEP —	Go into Standby mode	1	0000 0000	0000 0011	0000	0011	TO, PD	

### Note

1. When a Port register is modified as a function of itself (e.g., MOVF PORTB, 1, 0), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.
2. If this instruction is executed on the TMRO register (and where applicable, d = 1), the prescaler will be cleared if assigned.
3. If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.
4. Some instructions are 2-word instructions. The second word of these instructions will be executed as a NOP unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have valid instructions.
5. If the table write starts the write cycle to internal memory, the write will continue until terminated.

## Special Function Register (SFR) Map 1/3

Address	Name	Address	Name	Address	Name	Address	Name
FFFh	TOSU	FDFh	INDF2 <sup>(2)</sup>	FBFh	CCPR1H	F9Fh	IPR1
FFEh	TOSH	FDEh	POSTINC2 <sup>(2)</sup>	FBEh	CCPR1L	F9Eh	PIR1
FFDh	TOSL	FDDh	POSTDEC2 <sup>(2)</sup>	FBDh	CCP1CON	F9Dh	PIE1
FFCh	STKPTR	FDCh	PREINC2 <sup>(2)</sup>	FBCh	—	F9Ch	—
FFBh	PCLATU	FDBh	PLUSW2 <sup>(2)</sup>	FBBh	—	F9Bh	OSCTUNE
FFAh	PCLATH	FDAh	FSR2H	FBAh	—	F9Ah	—
FF9h	PCL	FD9h	FSR2L	FB9h	—	F99h	—
FF8h	TBLPTRU	FD8h	STATUS	FB8h	—	F98h	—
FF7h	TBLPTRH	FD7h	TMR0H	FB7h	PWM1CON	F97h	—
FF6h	TBLPTRL	FD6h	TMR0L	FB6h	ECCPAS	F96h	—
FF5h	TABLAT	FD5h	T0CON	FB5h	—	F95h	—
FF4h	PRODH	FD4h	—	FB4h	—	F94h	—
FF3h	PRODL	FD3h	OSCCON	FB3h	TMR3H	F93h	TRISB
FF2h	INTCON	FD2h	LVDCON	FB2h	TMR3L	F92h	TRISA
FF1h	INTCON2	FD1h	WDTCON	FB1h	T3CON	F91h	—
FF0h	INTCON3	FD0h	RCON	FB0h	SPBRGH	F90h	—
FEFh	INDF0 <sup>(2)</sup>	FCFh	TMR1H	FAFh	SPBRG	F8Fh	—
FEEh	POSTINC0 <sup>(2)</sup>	FCEh	TMR1L	FAEh	RCREG	F8Eh	—
FEDh	POSTDEC0 <sup>(2)</sup>	FCDh	T1CON	FADh	TXREG	F8Dh	—
FECh	PREINC0 <sup>(2)</sup>	FCCh	TMR2	FACH	TXSTA	F8Ch	—
FEBh	PLUSW0 <sup>(2)</sup>	FCBh	PR2	FABh	RCSTA	F8Bh	—
FEAh	FSR0H	FCAh	T2CON	FAAh	BAUDCTL	F8Ah	LATB
FE9h	FSR0L	FC9h	—	FA9h	EEADR	F89h	LATA
FE8h	WREG	FC8h	—	FA8h	EEDATA	F88h	—
FE7h	INDF1 <sup>(2)</sup>	FC7h	—	FA7h	EECON2	F87h	—
FE6h	POSTINC1 <sup>(2)</sup>	FC6h	—	FA6h	EECON1	F86h	—
FE5h	POSTDEC1 <sup>(2)</sup>	FC5h	—	FA5h	—	F85h	—
FE4h	PREINC1 <sup>(2)</sup>	FC4h	ADRESH	FA4h	—	F84h	—
FE3h	PLUSW1 <sup>(2)</sup>	FC3h	ADRESL	FA3h	—	F83h	—
FE2h	FSR1H	FC2h	ADCON0	FA2h	IPR2	F82h	—
FE1h	FSR1L	FC1h	ADCON1	FA1h	PIR2	F81h	PORTB
FE0h	BSR	FC0h	ADCON2	FA0h	PIE2	F80h	PORTA

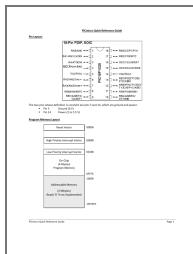
## Special Function Register Map 2/3

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
TOSU	—	—	—	Top-of-Stack Upper Byte (TOS<20:16>)					---0 0000
TOSH	Top-of-Stack High Byte (TOS<15:8>)								0000 0000
TOSL	Top-of-Stack Low Byte (TOS<7:0>)								0000 0000
STKPTR	STKFUL	STKUNF	—	Return Stack Pointer					00-0 0000
PCLATU	—	—	bit 21 <sup>(3)</sup>	Holding Register for PC<20:16>					---0 0000
PCLATH	Holding Register for PC<15:8>								0000 0000
PCL	PC Low Byte (PC<7:0>)								0000 0000
TBLPTRU	—	—	bit 21	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)					--00 0000
TBLPTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)								0000 0000
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								0000 0000
TABLAT	Program Memory Table Latch								0000 0000
PRODH	Product Register High Byte								xxxxx xxxx
PRODL	Product Register Low Byte								xxxxx xxxx
INTCON	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMROIF	INT0IF	RBIF	0000 000x
INTCON2	RBPU	INTEDG0	INTEDG1	INTEDG2	—	TMROIP	—	RBIP	1111 -1-1
INTCON3	INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF	11-0 0-00
INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed (not a physical register)								N/A
POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented (not a physical register)								N/A
POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented (not a physical register)								N/A
PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register)								N/A
PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 offset by W (not a physical register)								N/A
FSR0H	—	—	—	—	Indirect Data Memory Address Pointer 0 High				---- 0000
FSR0L	Indirect Data Memory Address Pointer 0 Low Byte								xxxxx xxxx
WREG	Working Register								xxxxx xxxx
INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed (not a physical register)								N/A
POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented (not a physical register)								N/A
POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented (not a physical register)								N/A
PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register)								N/A
PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 offset by W (not a physical register)								N/A
FSR1H	—	—	—	—	Indirect Data Memory Address Pointer 1 High				---- 0000
FSR1L	Indirect Data Memory Address Pointer 1 Low Byte								xxxxx xxxx
BSR	—	—	—	—	Bank Select Register				---- 0000
INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed (not a physical register)								N/A
POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented (not a physical register)								N/A
POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented (not a physical register)								N/A
PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register)								N/A
PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 offset by W (not a physical register)								N/A
FSR2H	—	—	—	—	Indirect Data Memory Address Pointer 2 High				---- 0000
FSR2L	Indirect Data Memory Address Pointer 2 Low Byte								xxxxx xxxx
STATUS	—	—	—	N	OV	Z	DC	C	---x xxxx
TMR0H	Timer0 Register High Byte								0000 0000
TMR0L	Timer0 Register Low Byte								xxxxx xxxx
T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0	1111 1111
OSCCON	IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0	0000 q000
LVDCON	—	—	IVRST	LVDEN	LVDL3	LVDL2	LVDL1	LVDL0	--00 0101
WDTCON	—	—	—	—	—	—	—	SWDTEN	--- ---0
RCON	IPEN	—	—	RI	TO	PD	POR	BOR	0--1 11q0

## Special Function Register Map 3/3

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
TMR1H	Timer1 Register High Byte								xxxxx xxxx
TMR1L	Timer1 Register Low Byte								xxxxx xxxx
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	0000 0000
TMR2	Timer2 Register								0000 0000
PR2	Timer2 Period Register								1111 1111
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000
ADRESH	A/D Result Register High Byte								xxxxx xxxx
ADRESL	A/D Result Register Low Byte								xxxxx xxxx
ADCON0	VCFG1	VCFG0	—	CHS2	CHS1	CHS0	GO/DONE	ADON	00-0 0000
ADCON1	—	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	-000 0000
ADCON2	ADFM	—	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0	0-00 0000
CCPR1H	Capture/Compare/PWM Register 1 High Byte								xxxxx xxxx
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								xxxxx xxxx
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000
PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000
ECCPAS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000
TMR3H	Timer3 Register High Byte								xxxxx xxxx
TMR3L	Timer3 Register Low Byte								xxxxx xxxx
T3CON	RD16	—	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC	TMR3CS	TMR3ON	0-00 0000
SPBRGH	EUSART Baud Rate Generator High Byte								0000 0000
SPBRG	EUSART Baud Rate Generator Low Byte								0000 0000
RCREG	EUSART Receive Register								0000 0000
TXREG	EUSART Transmit Register								0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	SEND8	BRGH	TRMT	TX9D	0000 0010
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x
DAUDCTL	—	RCIDL	—	SCKP	DRG10	—	WUC	ADDEN	-1-1 0-00
EEADR	EEPROM Address Register								0000 0000
EEDATA	EEPROM Data Register								0000 0000
EECON2	EEPROM Control Register 2 (not a physical register)								0000 0000
EECON1	EEPGD	CFG8	—	FREE	WRERR	WREN	WR	RD	xx-0 xx00
IPR2	OSCFIP	—	—	EEIP	—	LVDIP	TMR3IP	—	1--1 -11-
PIR2	OSCFIF	—	—	EEIF	—	LVDIF	TMR3IF	—	0--0 -00-
PIE2	OSCFIE	—	—	EEIE	—	LVDIE	TMR3IE	—	0--0 -00-
IPR1	—	ADIP	RCIP	TXIP	—	CCP1IP	TMR2IP	TMR1IP	-111 -111
PIR1	—	ADIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	-000 -000
PIE1	—	ADIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	-000 -000
OSCTUNE	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	--00 0000
TRISB	Data Direction Control Register for PORTB								1111 1111
TRISA	TRISA7 <sup>(2)</sup>	TRISA6 <sup>(1)</sup>	—	Data Direction Control Register for PORTA					11-1 1111
LATB	Read/Write PORTB Data Latch								xxxxx xxxx
LATA	LATA<7> <sup>(2)</sup>	LATA<6> <sup>(1)</sup>	—	Read/Write PORTA Data Latch					xx-x xxxx
PORTB	Read PORTB pins, Write PORTB Data Latch								xxxxx xxxx
PORTA	RA7 <sup>(2)</sup>	RA6 <sup>(1)</sup>	RA5 <sup>(4)</sup>	Read PORTA pins, Write PORTA Data Latch					xx0x 0000

## Documents / Resources



[\*\*engrcs PICmicro\*\*](#) [pdf] User Guide

**PICmicro**

[Manuals+](#)