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	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	13
Program Memory Size	896B (512 x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	96 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c620a-20e-ss

NOTES:

NOTES:

RETFIE	Return from Interrupt
Syntax:	[label] RETFIE
Operands:	None
Operation:	TOS PC, 1 GIE
Status Affected:	None
Encoding:	00 | 0000 | 0000 | 1001
Description:	Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example	RETFIE After Interrupt PC = TOS GIE = 1

RETLW	Return with Literal in W
Syntax:	[label] RETLW k
Operands:	0 k 255
Operation:	k (W); TOS PC
Status Affected:	None
Encoding:	11 | 01xx | kkkk | kkkk
Description:	The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example	`CALL TABLE;W contains table` ;offset value `TABLE` ;W now has table value ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; RETLW kn ; End of table Before Instruction W = 0x07 After Instruction W = value of k8

RETURN	Return from Subroutine
Syntax:	[label] RETURN
Operands:	None
Operation:	TOS PC
Status Affected:	None
Encoding:	00 | 0000 | 0000 | 1000
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.
Words:	1
Cycles:	2
Example	RETURN After Interrupt PC = TOS

SUBLW	Subtract W from Literal
Syntax:	[label] SUBLW k
Operands:	0 k 255
Operation:	k - (W) W)
Status Affected:	C, DC, Z
Encoding:	11 | 110x | kkkk | kkkk
Description:	The W register is subtracted (2 s complement method) from the eight bit literal 'k'. The result is placed in the W register.
Words:	1
Cycles:	1
Example 1:	SUBLW 0x02 Before Instruction W = 1 C = ? After Instruction W = 1 C = 1; result is positive
Example 2:	Before Instruction W = 2 C = ? After Instruction W = 0 C = 1; result is zero
Example 3:	Before Instruction W = 3 C = ? After Instruction W = 0xFF C = 0; result is negative

SUBWF	Subtract W from f
Syntax:	[label] SUBWF f,d
Operands:	0 f 127 d [0,1]
Operation:	(f) - (W) dest)
Status Affected:	C, DC, Z
Encoding:	00 | 0010 | dfff | ffff
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'.
Words:	1
Cycles:	1
Example 1:	SUBWF REG1,1 Before Instruction REG1= 3 W = 2 C = ? After Instruction REG1= 1 W = 2 C = 1; result is positive
Example 2:	Before Instruction REG1= 2 W = 2 C = ? After Instruction REG1= 0 W = 2 C = 1; result is zero
Example 3:	Before Instruction REG1= 1 W = 2 C = ? After Instruction REG1= 0xFF W = 2 C = 0; result is negative