



# Programming Microcontroller

Using Embedded C

- \* Embedded C is nothing but a subset of C language which is compatible with certain microcontrollers.
- \* Some features are added using header files like `<avr/io.h>`, `<util/delay.h>`.
- \* `scanf()` and `printf()` are removed as the inputs are scanned from the sensors and outputs are given to the ports.
- \* Control structures remain the same like if-statement, for loop, do-while etc.



# *What is Embedded C?*

```
* //Headers
* #include<avr/io.h> //Header file for AVR i/o
* #include<util/delay.h> //Header file for delay
* // main program
* Int main()
* {
* while(1)
* { code... }
* Return (0);
* }
```

**\* Structure of a C program  
for an embedded system**

\* Syntax:

```
if( condition)
{
    statement.....
}
else
{
    statement.....
}
```

\* **If-statement**

```
Int a=4;  
Int b=5;  
If(a>b)  
    printf(" a is largest");  
else  
    printf(" b is largest");
```



**Program for if-  
statement**

\*Syntax:

Initial counter

Do

{

statement.....

update statement

}

While(condition);



**Do- while statement**

```
Int a=4;  
do  
{  
    a++;  
}  
while(a>5);
```



**Program for do-while**

\* Syntax:

```
For( initial counter; test condition; update stmt)
{
    statement.....
    statement.....
}
```

\* Program:

```
for(int i=0;i<5;i++)
    printf("Hello Robofifa'14");
```



**For- statement**



## \*We have four Ports

\*PORT A

\*PORT B

\*PORT C

\*PORT D

\*All ports have Read-Modify-Write functionality.

\* (all pins are capable of performing dual functions!)

# \*In ATmega16

- \* Each port pin have three register bits
- \* 1. DDRx - Data Direction Register
- \* 2. PORTx - Port output
- \* 3. PINx - Port input

## \*Three Registers

- \* If DDRx is configured as logic high, the pin is configured as output pin.
- \* If DDRx is configured as logic low, the pin is configured as input pin.
- \* `DDRx = 0xff; // configuring as o/p`
- \* `DDRx = 0x00; // configuring as i/p`

\* **DDRx**

- \* If PORTx is written logic one when the pin is configured as an output pin, the port pin is driven high (one).
- \* If PORTxn is written logic zero when the pin is configured as an output pin, the port pin is driven low (zero).

DDRx	PORTx	Output
1	0	0
1	1	1

- \* Ex: DDRB = 0xff; //configured as o/p
- \* PORTB = 0xff; //output high
- \* give delay
- \* PORTB = 0x00; //output low

\* PORTx

- \* PINx is used to read the data
- \* Ex: Say a sensor is connected to lsb of Port D.
- \* To read the status of the sensor, we use PIND
- \* i.e., `x=PIND;` // x acquires the status of portD
- \* `if(x&0b00000001)`
- \* `{ sensor is ON }`
- \* `else`
- \* `{ sensor is OFF }`

\* PINx

Program:

```
main()
{
    DDRB=0XFF;
    while(1)
    {
        PORTB=0XFF;
        _delay_ms(255);
        PORTB=0X00;
        _delay_ms(255);
    }
}
```



**ex: Blinking LED**

Port Pin	Alternate Function
PA7	ADC7 (ADC input channel 7)
PA6	ADC6 (ADC input channel 6)
PA5	ADC5 (ADC input channel 5)
PA4	ADC4 (ADC input channel 4)
PA3	ADC3 (ADC input channel 3)
PA2	ADC2 (ADC input channel 2)
PA1	ADC1 (ADC input channel 1)
PA0	ADC0 (ADC input channel 0)

## \* Alternate Functions of Port A

Port Pin	Alternate Functions
PB7	SCK (SPI Bus Serial Clock)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB5	MOSI (SPI Bus Master Output/Slave Input)
PB4	$\overline{SS}$ (SPI Slave Select Input)
PB3	AIN1 (Analog Comparator Negative Input) OC0 (Timer/Counter0 Output Compare Match Output)
PB2	AIN0 (Analog Comparator Positive Input) INT2 (External Interrupt 2 Input)
PB1	T1 (Timer/Counter1 External Counter Input)
PB0	T0 (Timer/Counter0 External Counter Input) XCK (USART External Clock Input/Output)

 **Alternate functions  
of Port B.**



**Table 28.** Port C Pins Alternate Functions

Port Pin	Alternate Function
PC7	TOSC2 (Timer Oscillator Pin 2)
PC6	TOSC1 (Timer Oscillator Pin 1)
PC5	TDI (JTAG Test Data In)
PC4	TDO (JTAG Test Data Out)
PC3	TMS (JTAG Test Mode Select)
PC2	TCK (JTAG Test Clock)
PC1	SDA (Two-wire Serial Bus Data Input/Output Line)
PC0	SCL (Two-wire Serial Bus Clock Line)

 **Alternate Functions of  
PORT C.**

Port Pin	Alternate Function
PD7	OC2 (Timer/Counter2 Output Compare Match Output)
PD6	ICP1 (Timer/Counter1 Input Capture Pin)
PD5	OC1A (Timer/Counter1 Output Compare A Match Output)
PD4	OC1B (Timer/Counter1 Output Compare B Match Output)
PD3	INT1 (External Interrupt 1 Input)
PD2	INT0 (External Interrupt 0 Input)
PD1	TXD (USART Output Pin)
PD0	RXD (USART Input Pin)

 Alternate Functions of Port D.