AMIT

Beyond Education


# Digital Input Output

# Digital Input Output

A Digital Input Output is a peripheral that deals with digital signals, either by generating a digital signal (Output Mode) or by receiving it (Input Mode).

Any peripheral consists of Memory + Circuit

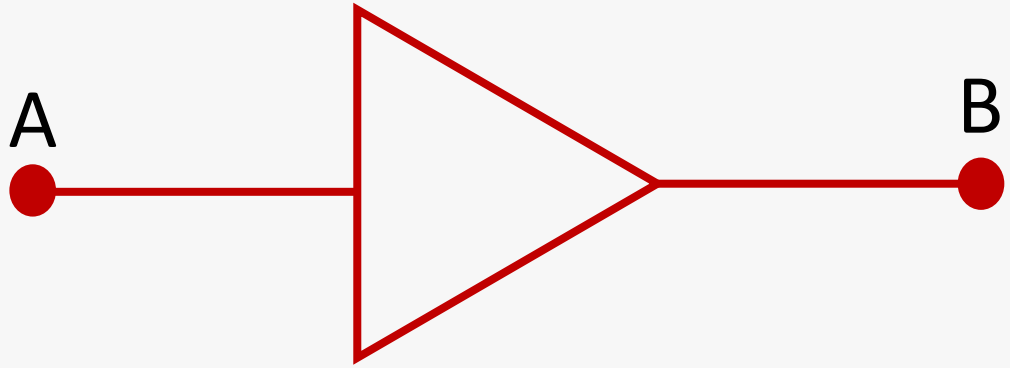
- Memory : to interface with the processor because the processor can only deal with the memory.
- Circuit : to do its job .

# Digital Input Output

- Buffer : it is a circuit, opposite the NOT gate circuit. The voltage on point A is shown on point B.

➡ What is the benefit of ***Buffer*** ? ***(Current Limitation)***

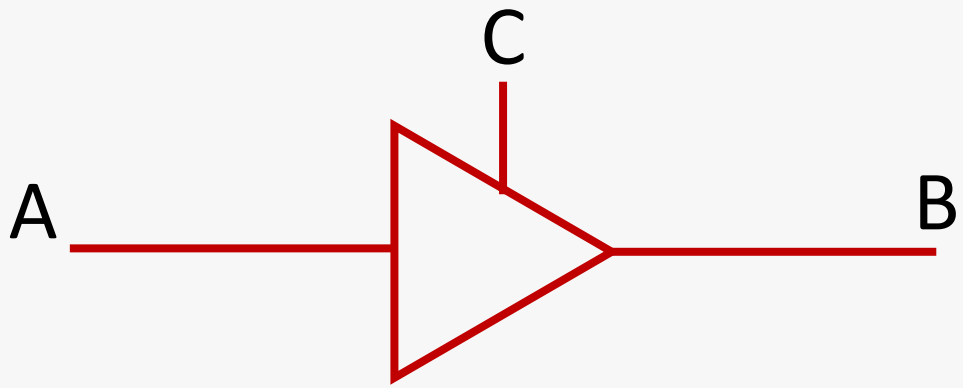
The buffer circuit has very high impedance, which means that the buffer circuit transfers volt does not transfer current .The output current is very small.




AMIT
Beyond Education

# Digital Input Output

- Tri-State Buffer : it is a normal buffer but has another signal to control the direction of the signal.



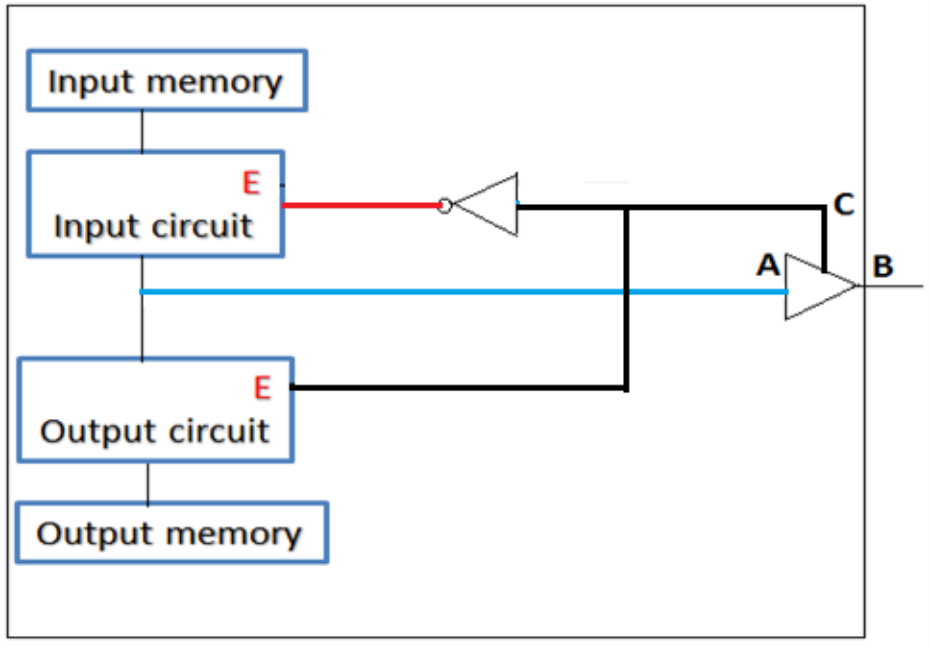

| C | Output |
|---|---|
| 1 | A to B |
| 0 | B to A |

The basic block unit for the DIO pin is the ***Tri-State Buffer***. Any DIO pin consists of a Tri-State buffer as a main component. The Tri-State buffer controls the direction of the data, A to B or B to A.

AMIT
Beyond Education

# DIO Block Diagram

Each DIO pin has this block Diagram




Writing 0 To C, Enables the Input Circuit and make the Buffer direction to (A <- B) which makes the PIN in Input Mode

Writing 1 To C, Enables the Output Circuit and make the Buffer direction to (B <- A) which makes the PIN in Output Mode

AMIT
Beyond Education

# DIO Block Diagram

If you put ***zero*** to ***c,*** the direction of the signal will be from ***B*** to ***A*** and the input circuit will be enabled and will sense any change will happen in A and write the value to the input Memory and the processor can read the input memory .

If you put ***One*** to ***c*** the direction of the signal will be from ***A*** to ***B*** and the output circuit will be enable . When the output circuit work it will read the output memory. The processor Can write on the output memory and the value will be output from ***A*** to ***B***.

AMIT
Beyond Education

# Controlling Memory

The processor is connected to ( input memory ) and ( output memory ) through Data Bus – Address Bus – Control Bus.

This Memory mostly is RAM memory. Because if this memory is Rom the CPU will need to flash driver to write on  This memory.

If we have a pin in output mode and the CPU write a value on this pin and after some times the CPU need to read the value on this pin . The CPU will read the input memory or the output  Memory ?

# Controlling Memory

- The o/p memory connected to the processor with address bus, control bus and data bus that is mean the CPU can read and write from o/p memory .

- The i/p circuit disconnected while the o/p circuit is enable so if the CPU reads from the i/p circuit it will read an old data because the last data CPU written it in the o/p memory .

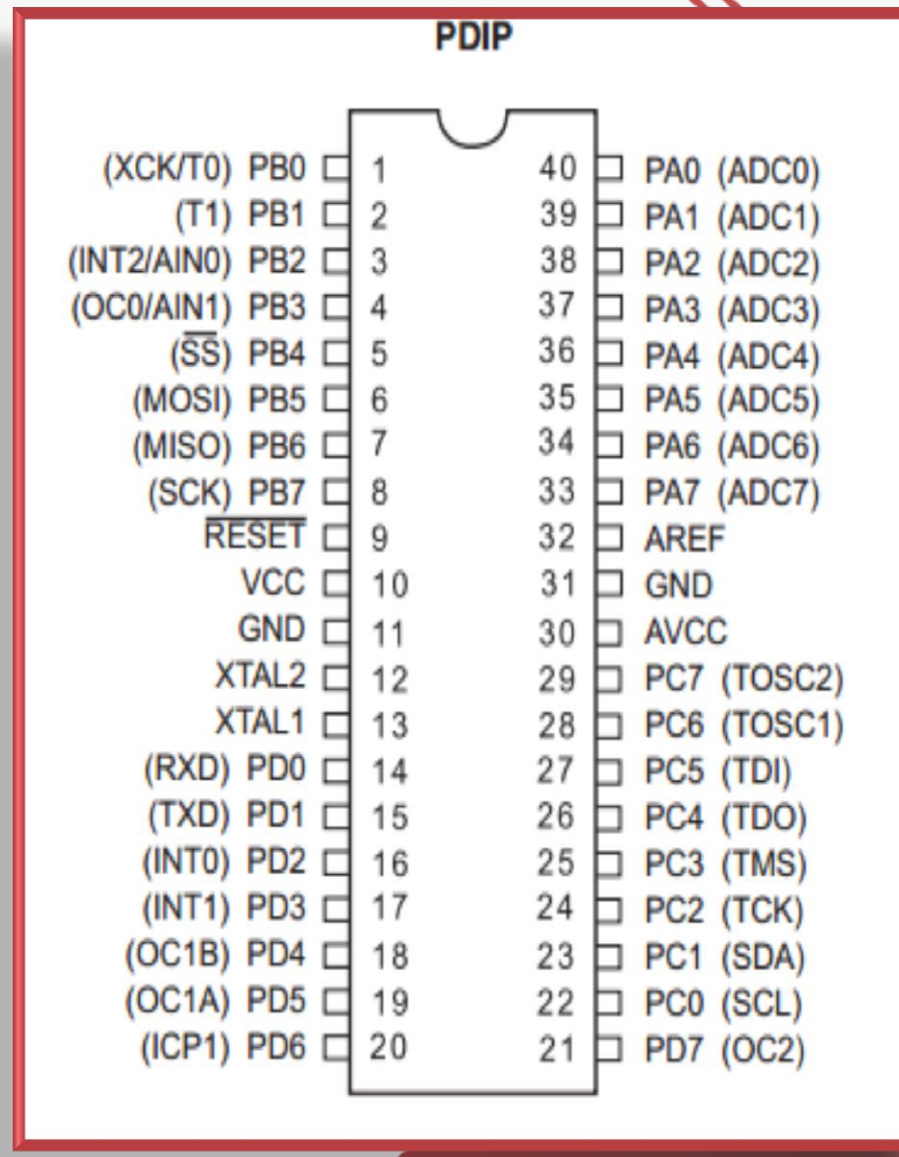
Some times the pin is synchronized. That is mean the o/p memory is synchronized with The input memory and the data written in the o/p memory you can read it from the Input memory.

# AVR Microcontroller

In our course we will use Microcontroller AVR Atmega32.
It has 32 DIO pins grouped as following:

1- **PORTA** has 8 DIO Pins from ***A0*** to ***A7***
2- **PORTB** has 8 DIO Pins from ***B0*** to ***B8***
3- **PORTC** has 8 DIO Pins from ***C0*** to ***C8***
4- **PORTD** has 8 DIO Pins from ***D0*** to ***D8***

Each pin can work either in input mode or output mode.




AMIT
Beyond Education

# AVR Microcontroller

The processor can only deal with memories so the processor will write on the control memory( ***C*** ) and the control circuit will read the value on the control memory . So the CPU can control the direction of the pin input or output.

If the pin is input the CPU can control the value to be output on the pin by writing to the o/p memory.

# AVR Microcontroller

Each memory location from what we explained called Register and has address and this address differs from microcontroller to another .

For example :-

the o/p memory has address 0x100.
the i/p memory has address 0x101.
the control memory has address 0x102.

The Data Sheet will describe all registers in the micro controller by give each register name and address.

The names of the registers only to me but if I want to access any register I will access it With his address.

So we will interested with the address not the name.

AMIT
Beyond Education

# AVR Microcontroller

In the AVR the register of the control memory is called DDR

And the register of the o/p memory is called PORT.

And the register of the i/p memory is called PIN.

We have 32 DIO pin divided into 4 groups ( A – B – C - D ) each group has 8 pins and the data bus of the AVR is 8 bit and the resolution of each register in the AVR is 8 bit so each register whether DDR , PORT or PIN is 8 bit.

# AVR Microcontroller

So each register can control one group and each bit in the register can control one pin in this group.

Group A has three register ( DDRA , PORTA & PINA ).

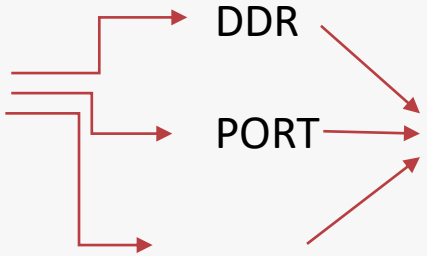
Group B has three register ( DDRB , PORTB & PINB ).

Group C has three register ( DDRC , PORTC & PINC ).

Group D has three register ( DDRD , PORTD & PIND ).

# AVR DIO

Every port has 3 control registers




The size of each register is 8 bit, every bit ***corresponding*** to 1 Pin of the port

**1- DDR** (Data Direction Register) in this register we can define the pin is output or input

- Set **0** → Input
- Set **1** → Output

**2- PORT** : This register is used in output mode to set the digital output value

- set **1** → this pin carry 5v
- set **0** → this pin carry 0v

**3- PIN**: we use this register in case the pin is defined input

- if **1** → The Pin is connected to 5v
- if **0** → The Pin in connected to 0v

AMIT
Beyond Education

# AVR DIO Example

***Example 1***

| Configuring Pin A0 and output 5V |
| --- |
| DDRA=0b00000001 ; |
| PORTA=0b00000001 ; |

***Example 2***

| Configuring All PORTD Output, lower half connected to 5V and upper half connected to ground |
| --- |
| DDRD=255; |
| PORTD=0x0f; |

AMIT
Beyond Education

# AVR DIO Example

**Remember the major three numbering systems used in C:**

Binary : PORTA=0b11111111;

Decimal : PORTA=255;

Hexadecimal : PORTA=0xff ;

All of these Statements are Equivalent

AMIT
Beyond Education

# AVR DIO Example

The Tristate Buffer can limit the current that output from the pin and the max current  can be output from the pin is 25mA and this current is called source current

In the input mode the max current that enter to the pin is 10mA and this current Is called Sync current.

# Interfacing Led

## LED Definition

Light Emitting Diode is an electrical element that emits light by supplying a voltage difference between its terminals

## LED Connection:

The LED has two pins, positive one and negative one.
In your kit there are 8 LEDs all of them are common ground..

AMIT
Beyond Education

# Interfacing Led

Our Leds can work between 80-150mW.

***P = V * I***

If you connect the positive side of the led with ***A0*** and the negative side to the ***GND***

***( Forward Connection ).***And make the direction of the ***A0*** output and write 1 to the ***bit*** number ***0*** in the output memory the pin will output ***5v*** to the led and the led will be ***light up.***

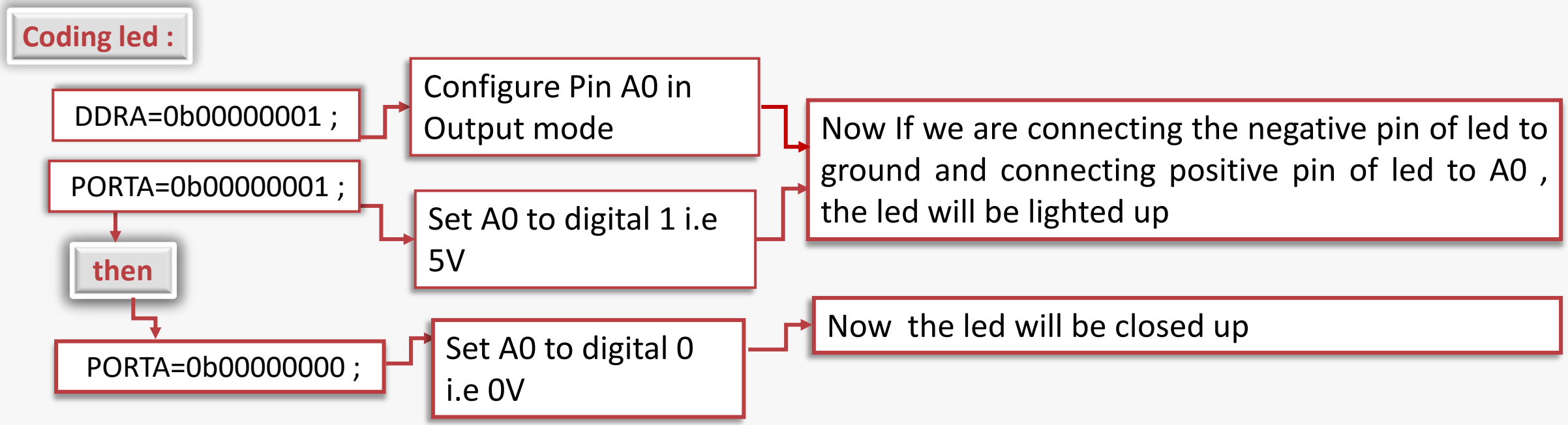
***P = 5v * 25mA = 125mWatt ( in the range between 80 to 150 ).***

And if you write ***0*** to the ***bit*** number ***0*** in the o/p memory the pin will output ***0v*** to the led

And the led ***will not light up***.

AMIT
Beyond Education

# Interfacing Led




If you connect the ***+ve*** of the led to the ***5v*** of the kit and connect the ***–ve*** to the ***GND, t***he led will be ***Burn***.

AMIT
Beyond Education

# Interfacing Led

This is because, if you take the power of the kit from ***power supply*** it will out ***350mA***

And if you take the power of the kit from your ***laptop*** it will out ***150mA*** and from the

Equation --→ ***P = V * I = 5v * 150mA = 750mW.***

So if we connect the ***+ve*** to the pins of the micro the ***leds will not be Burn.***

And if you connect it to the ***5v*** of the micro the led ***will be Burn*** because of the **high power.**

AMIT
Beyond Education

# The Super Loop

Any C project in Embedded Systems application shall have an infinite loop called the ***super loop***. This loop is a ***must*** even if you we will leave it empty !

This loop prevents the program counter ***(PC)*** from continues incrementing over the flash memory and execute a garbage code. i.e. the while(1) represents the end of the code.

```
void main(void)
{
    /* Initialization  Part */



    /* The Super Loop */
    while (1)
    {

    }
}
```

AMIT
Beyond Education

# LAB 1

Write a C code to turn on LED on Pin A0

# Using Delay Function

***Busy Loop Delay***

Software Technique the use a loop with effect just to halt the processor for certain time. We will use a library called ***"avr/delay.h"*** that provides two basic functions:

1- ***_delay_ms*** ( *_value_in_ms* ) /* Apply a delay in milli seconds */
2- ***_delay_us*** ( *_value_in_us* ) /* Apply a delay in micro seconds */

***Note***

Before using the delay library, we have to define our system frequency by writing this command:

```
#define F_CPU   12000000  /* Define a CPU frequency of 12 Mega Hertz */
```

AMIT
Beyond Education

# LAB 2

Write a C code to turn on LED on Pin A0 for 1 second and then turn it off.

# LED Blinking

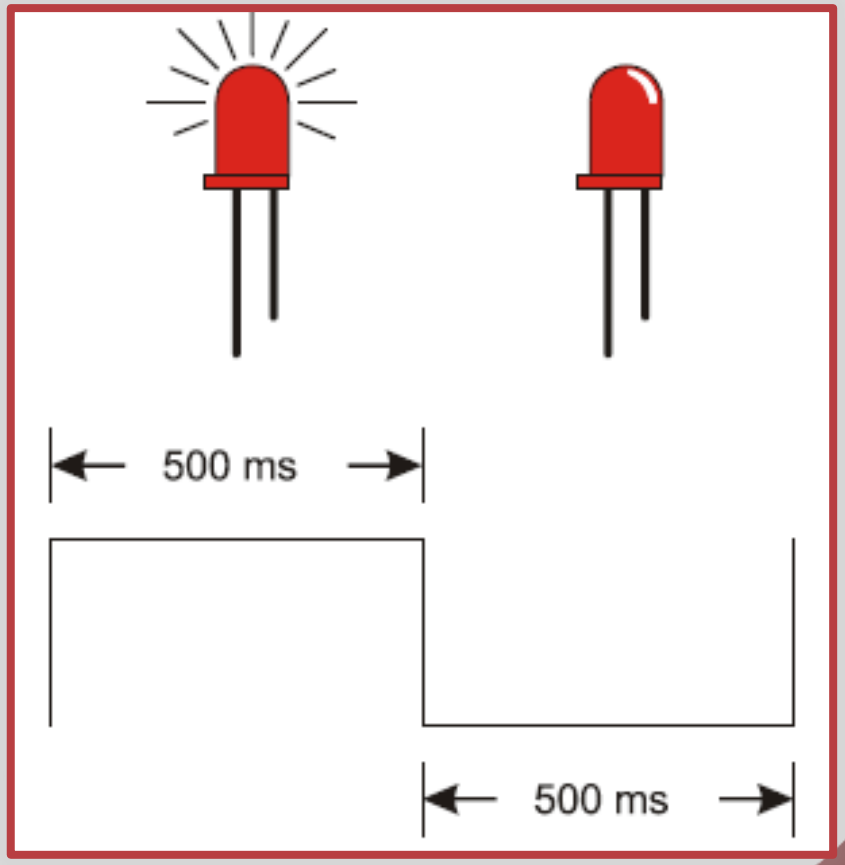
## LED Blinking Algorithm

```
/* Loop forever */
while (1)
{
   /* Turn LED on */
   PORTA = 0x01;

   /* Apply 0.5 Second Delay */
   _delay_ms(500);

  /* Turn LED off */
   PORTA = 0x00;

   /* Apply 0.5 Second Delay */
   _delay_ms(500);
}
```




AMIT
Beyond Education

# LAB 3

Write a C code to blink a LED Every 1 second

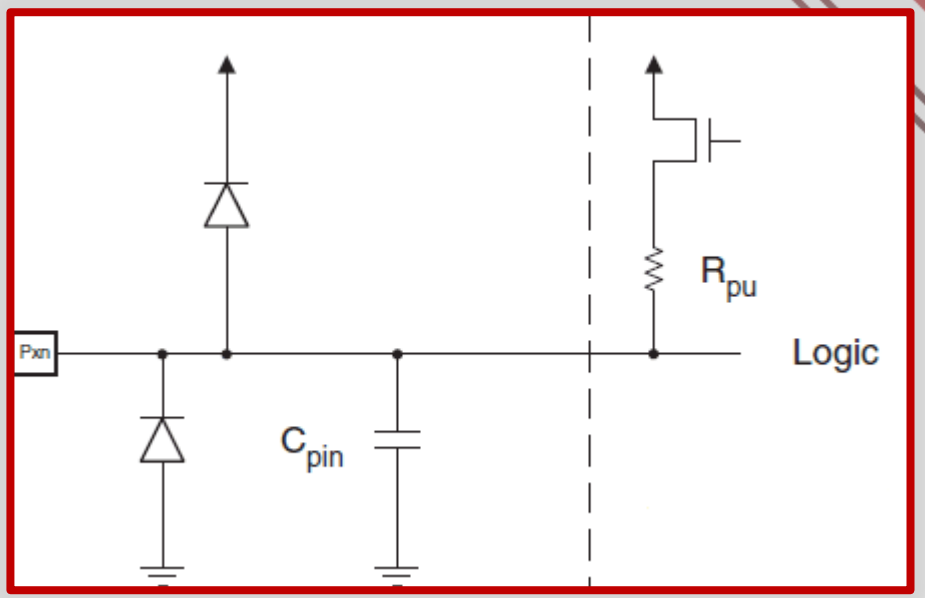
# LAB 4

Write a C Code that apply Some LED animations

AMIT
Beyond Education

# Pin equivalent circuit

All port pins have individually selectable pull-up resistors with a resistance which is independent of the supply voltage. All pins have protection diodes to both VCC and Ground.




AMIT
Beyond Education

# Pin equivalent circuit

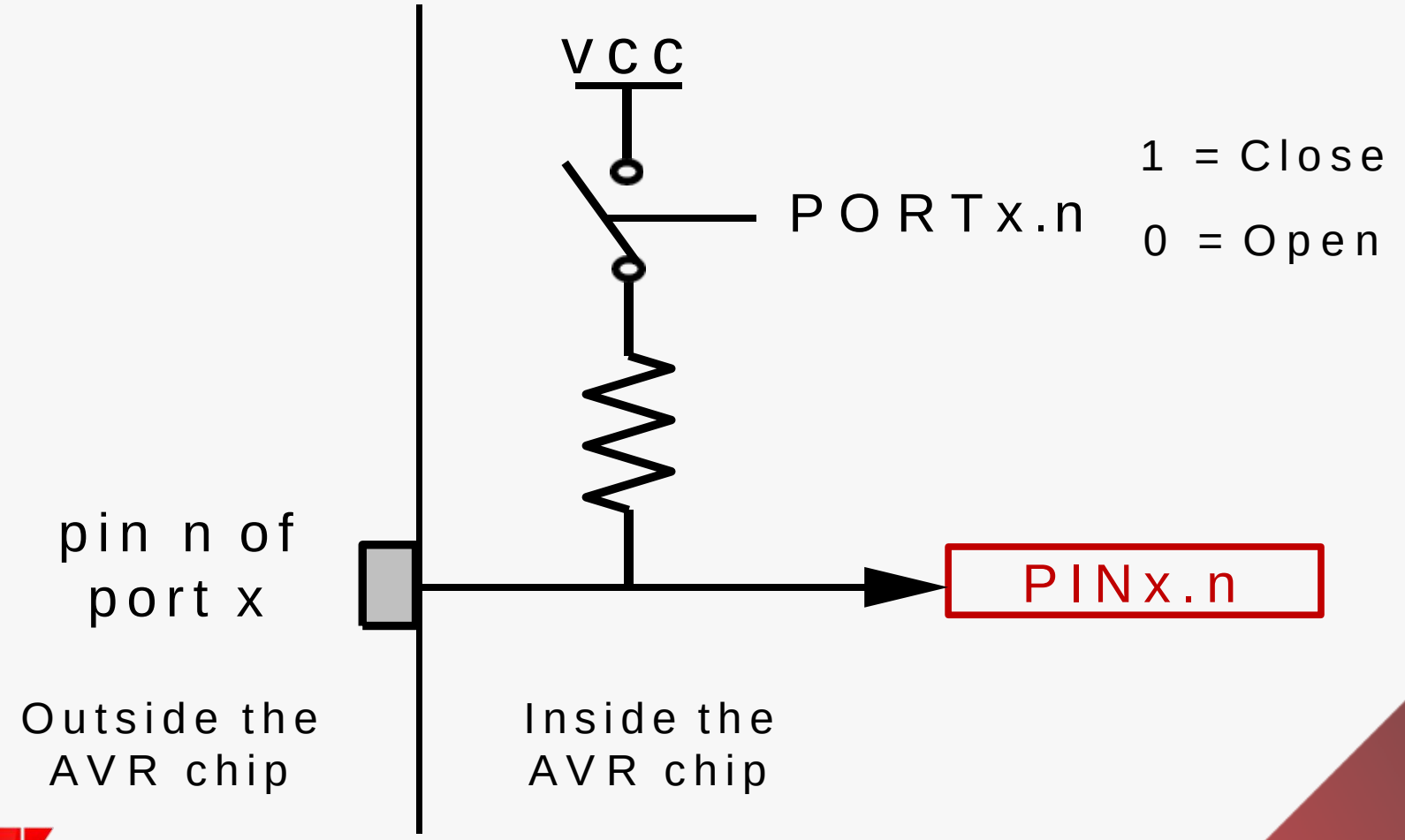
**Current sinking and sourcing**

- Each I/O pin can sink or source a maximum current of 40mA.
- Although each pin can sink or source 40mA current, it must be ensured that the current sourced or sinked from all the ports combined, should not exceed 200mA.

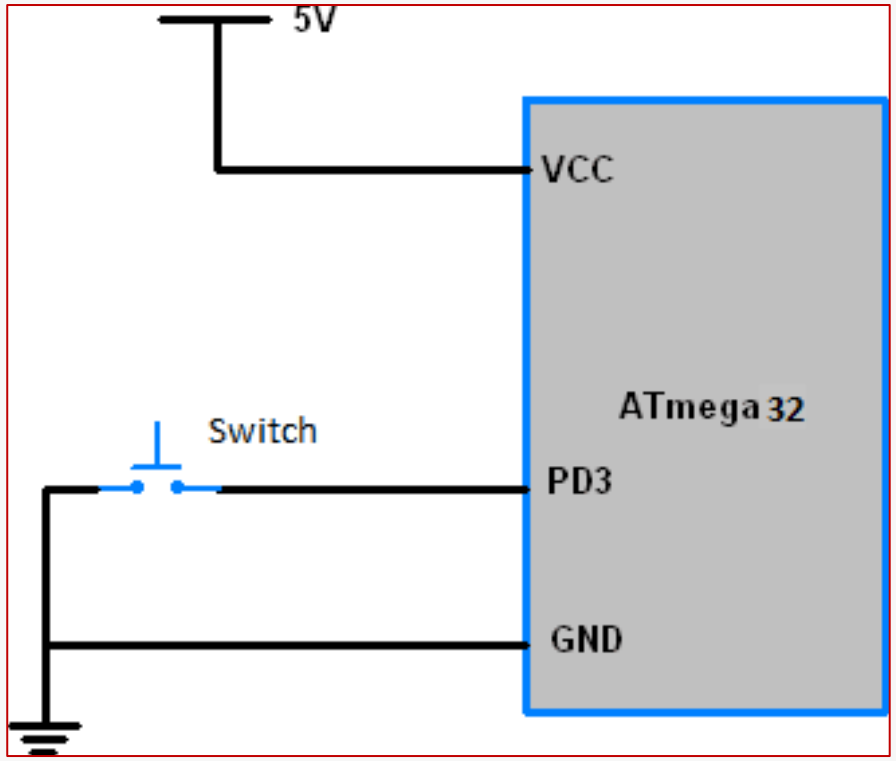
- There are further restrictions on the amount of current sourced or sinked by each port. For information on this, refer page 292 in the datasheet of ATmega16 given in the attachments section of this topic.

AMIT
Beyond Education

# Pull Up Resistor



# Button Interfacing




In This Circuit the wiring of Button is meant to be  active low

Means when Button is pressed the Input to MCU pin
PD3 is LOW signal = Logic 0 = GND

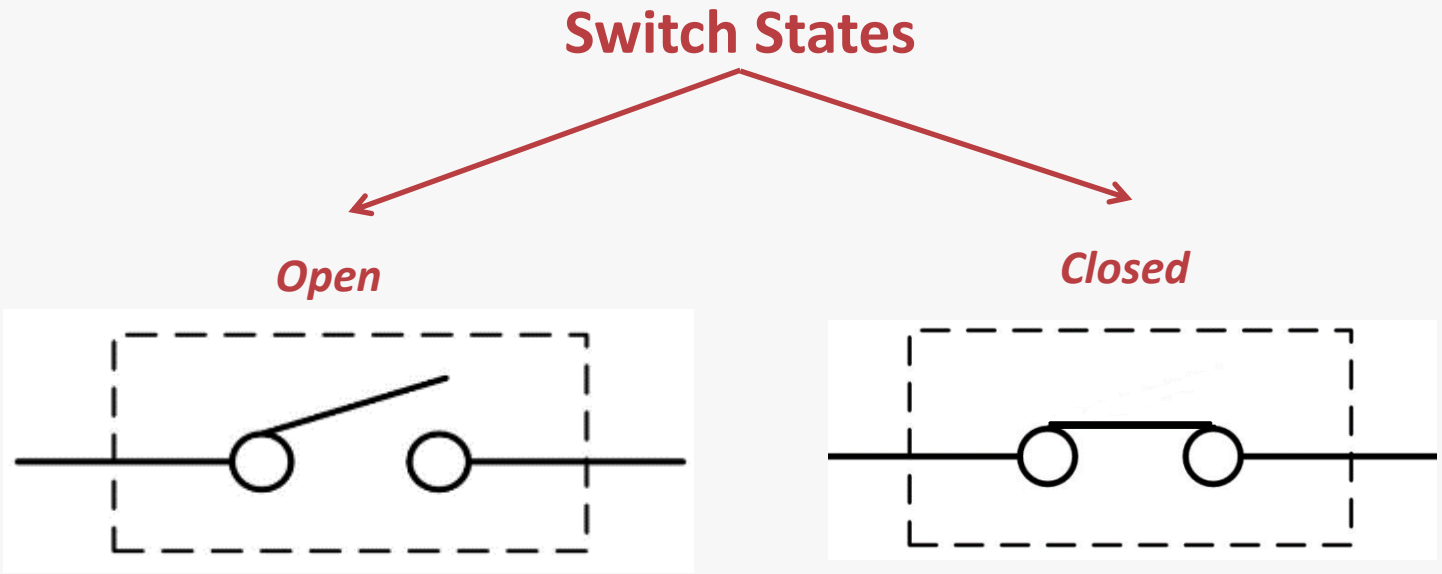
Where internal pull-up resister is activated on PD3

AMIT
Beyond Education

# Mechanical Switch

***Mechanical switch***

is an electrical component that can connect or break an electrical circuit.

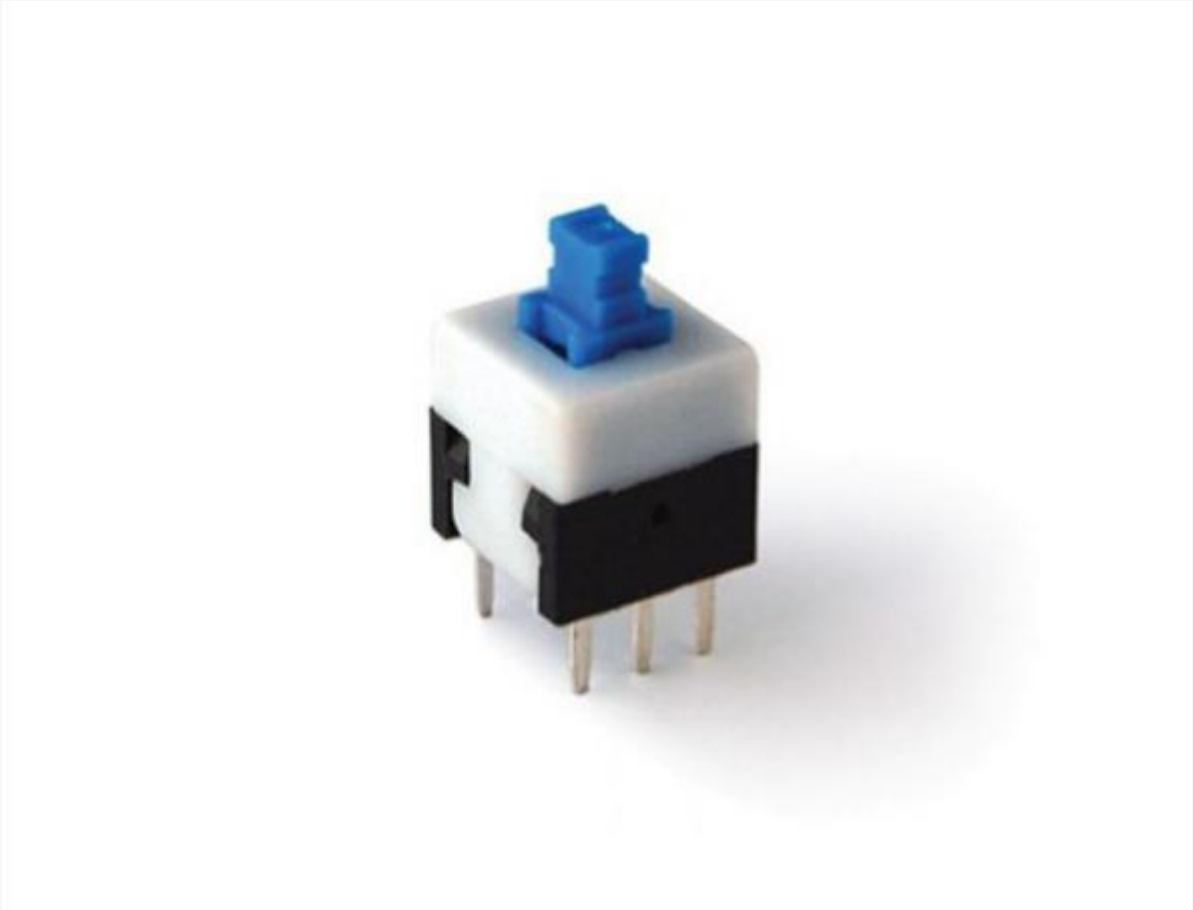



AMIT
Beyond Education

# Tactile Switch

AMIT
Beyond Education

# Push Button



AMIT
Beyond Education

# Paddle Switch

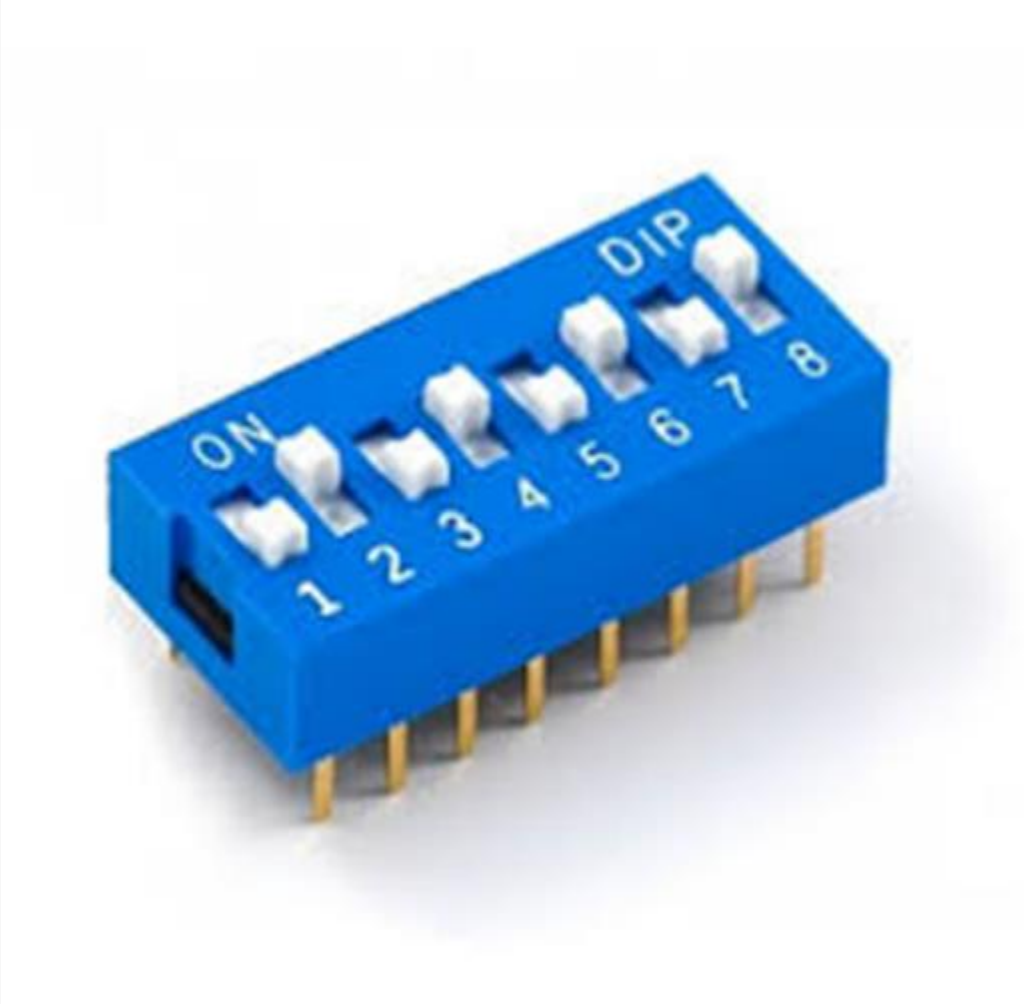
AMIT
Beyond Education

# Rocker Switch

# Toggle Switch


AMIT
Beyond Education

# DIP Switch




AMIT
Beyond Education

# Thumbwheel Switch

AMIT
Beyond Education

# Limit Switch

AMIT
Beyond Education

# Slide Switch

AMIT
Beyond Education

# Rotary Switch

AMIT
Beyond Education

# Reed Switch

AMIT
Beyond Education

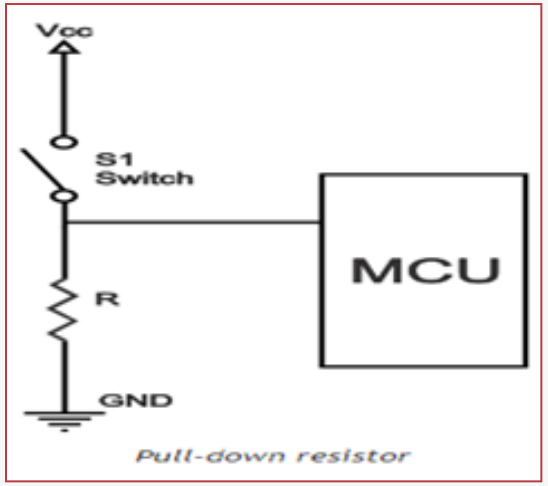
# Knife Switch

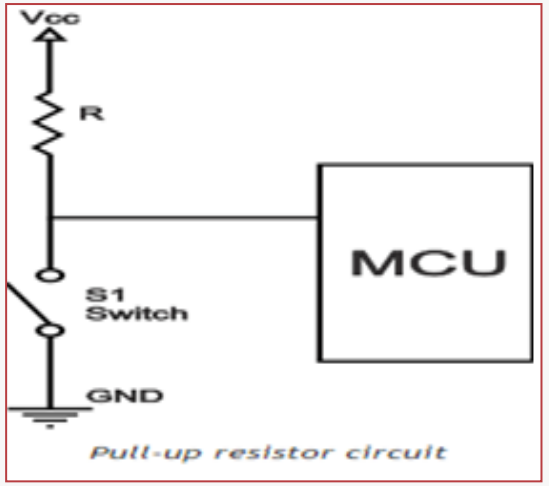
AMIT
Beyond Education

# Key Switch

# Interfacing Mechanical Switch

Switch shall be connected by pull up or pull down resistor to avoid short circuits.




AMIT
Beyond Education

# Interfacing Mechanical Switch

In AVR Microcontroller, all DIO pins have internal pull up resistors that can be activated or not.

**Input Digital Pin**

***Floating***

```
/* Configure PIN as input */
DDR -> 0
```

In this state, the DIO pin has 3 states, 0 when connected to GND, 1 when connected to VCC, floating when not connected to anything which may be read as 0 or as 1 !

***Note*** Never let an input pin as floating to avoid noise affection.

***Internal Pull Up***

```
DDR -> 0     /* Configure PIN as input    */
PORT -> 1    /* Activate Internal Pull up */
```

In this state, the DIO pin has 2 states only, 0 when connected to GND, 1 when connected to VCC or when not connected to anything.

AMIT Beyond Education

# Reading Input PIN

The registers ***PINA***, ***PINB***, ***PINC*** and ***PIND*** are used to check the status of the input pins. If the corresponding bit for a certain pin is ***0***, then the pin is connected to ***GND***. If the corresponding bit for a certain pin is ***1***, then the pin is connected to ***VCC***.

```
/* Check if Pin A0 is conncted to GND  */
if ( (PINA & 0b00000001) == 0)

/* Check if Pin B3 is connected to VCC */
if ( (PINB & 0b00001000) != 0)
```

AMIT
Beyond Education

# LAB6

Write a code that uses a DIP switch to control a string of 8 LEDs. When the DIP switch is On the LED string shall be flashing every 500 ms. When the DIP switch off the LED string shall be also off.

AMIT
Beyond Education

# Assignment 1

**Write a C code that simulate the traffic lightening system:**

1- Turn On Green LED for 10 seconds

2- Turn On Yellow LED for 3 seconds

3- Turn On Red LED for 10 seconds

4- Apply these forever while counting the seconds down on a 2 7-segment displays.

AMIT
Beyond Education

# Assignment

**Write a C code that apply 8 different animations on 8 LED string based on the value of 3 way DIP Switch as following:**

| DIP value | LED Action |
|---|---|
| 1 | Flashing every 500 ms |
| 2 | Shifting Left every 250 ms |
| 3 | Shifting Right every 250 ms |
| 4 | 2-LEDs Converging every 300 ms |
| 5 | 2-LEDs Diverging every 300 ms |
| 6 | Ping Pong effect every 250 ms |
| 7 | Incrementing (Snake effect) every 300 ms |
| 8 | 2-LEDs Converging/Diverging every 300 ms |

AMIT
Beyond Education

# Displays

***1-Segment Display*** ***7segement Display***

***Alphanumeric***

These displays systems made to appear characters and numbers.

- ***Disadvantage :***

  1- consuming a lot of pins.

  2- limited in what it is display ( Display Characters and numbers only ).

  3- consuming high power because each segment is a LED.

- ***Advantage :***

  1- Attractive.

  2- Very easy software.

AMIT

Beyond Education

# Displays

***2-Dot Matrix Display:-***

you can draw letters , shapes and videos.

- ***Disadvantage :*** Consume High Power and a lot of pins and High Cost.
- ***Advantage :*** Very Attractive so we use it in Advertising.

# Displays

***3-Liquid crystal display***

***-Character LCD :-*** ***Drawing characters and numbers but in one color.***

***-Graphical LCD :-*** ***Drawing characters, numbers and Drawings but in one color.***

***-Colored LCD :-*** ***Drawing characters, numbers and Drawings but more than one color.***

AMIT
Beyond Education

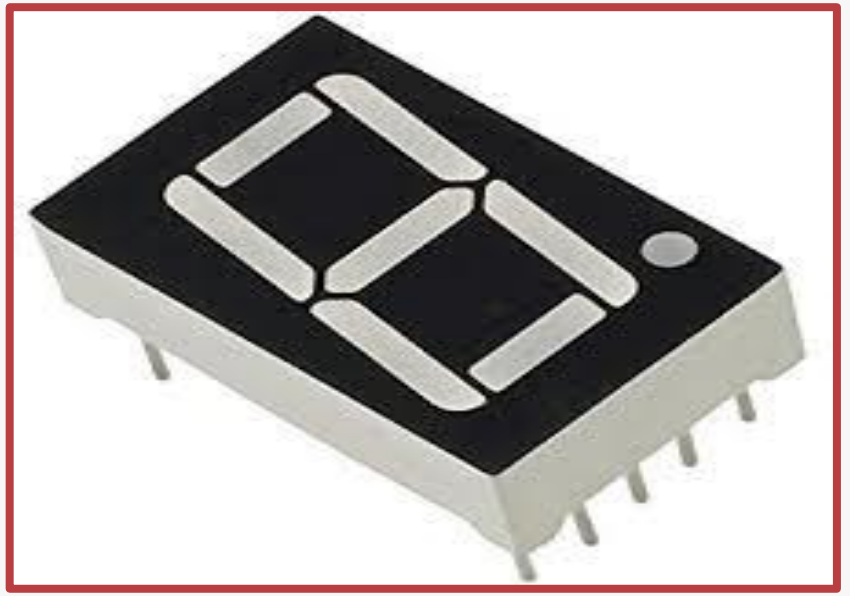
# Displays

In the previous Displays we did not  talk about touch screen because in the science there Is no thing called touch screen it is called touch panel it is transparent  panel that sense Where your finger pressed . You can put the touch panel on any thing and you can control This thing with pressing on the touch panel.

So they put touch panel on LCD and naming them touch screen but in real they are two Different modules.

# Interfacing 7-Segments

***1-7Segment : -*** ***it is a display for displaying numbers only***



***It is consist of a few LEDs and connected them together***

***in a certain way to do the 7segment.***

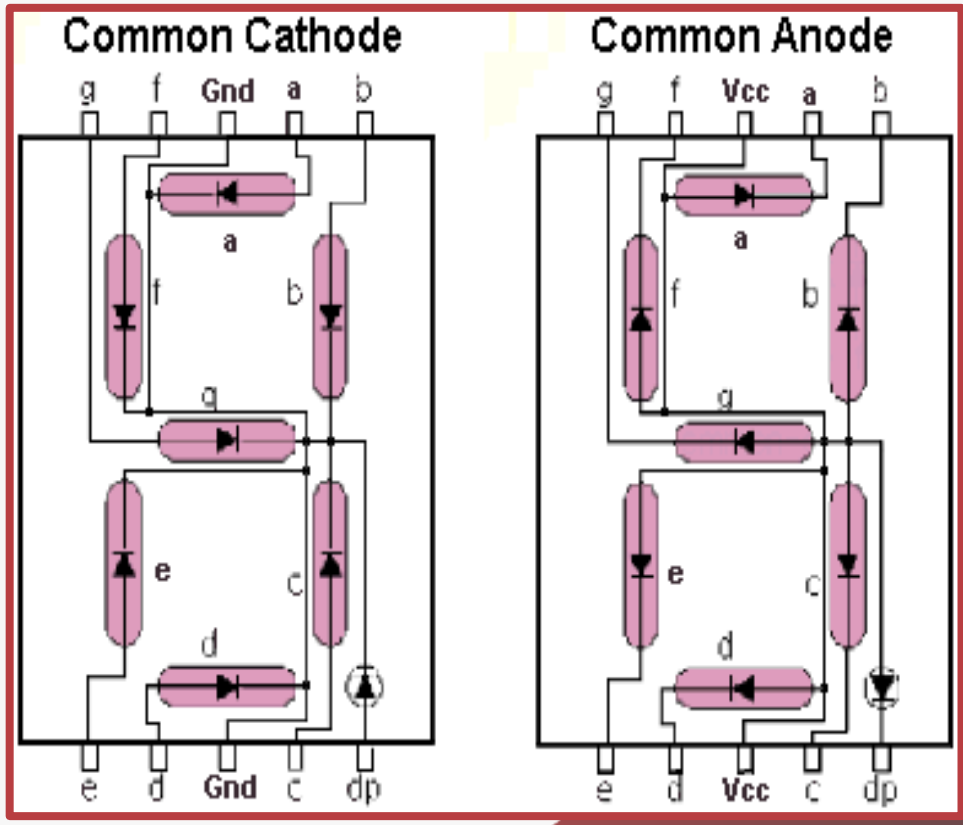
if I want to display a number I will light up All LEDs that represent this number.

AMIT
Beyond Education

# Interfacing 7-Segments

We have ***7*** LEDs and Each LED has ***two*** parties So now Each 7Seg has ***14*** Party . We will take all ***–ve*** of the LEDs and connected them together And enable one party for ***–ve*** and naming it ***Common*** and connect this party to the ***ground.***

So this 7Seg now has ***7*** parties for ***+ve*** and ***one Common*** party called ***Common Cathode.***

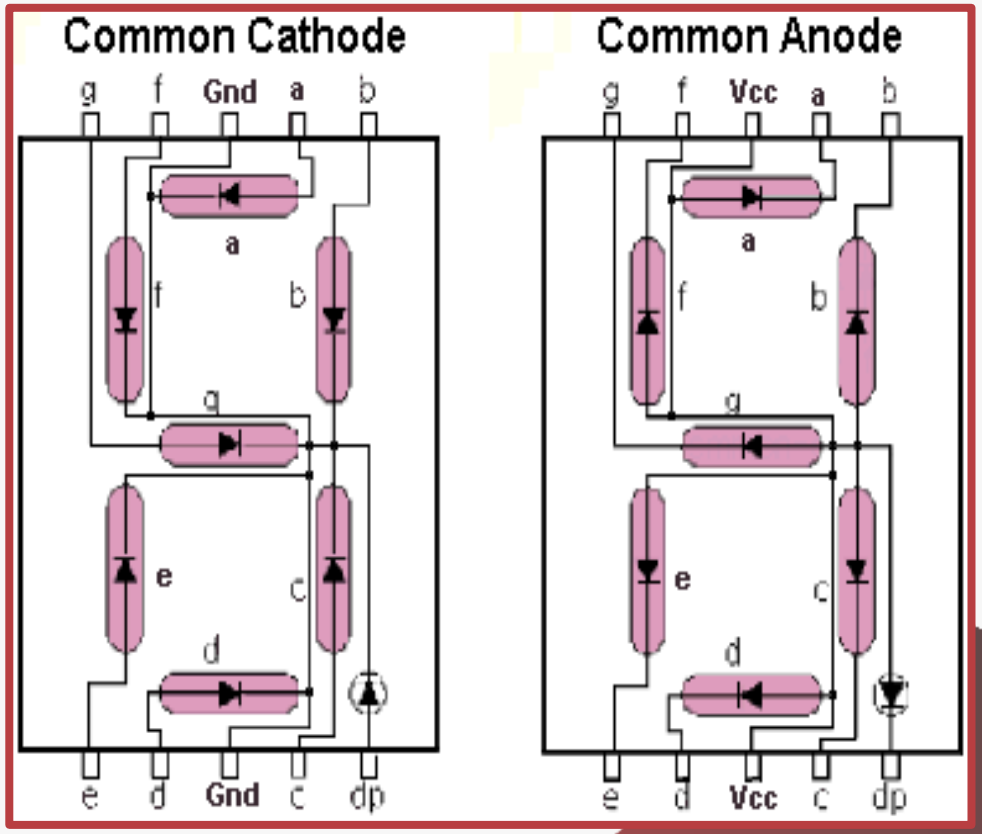



AMIT
Beyond Education

# Interfacing 7-Segments

On another way we will take all ***+ve*** of the LEDs and connected them together and enable one party for ***+ve*** and naming it ***Common*** and connect this party to the ***VCC.***

So this 7Seg now has ***7*** parties for ***–ve*** and ***one Common*** party called ***Common Anode.***




AMIT
Beyond Education

# Interfacing 7-Segments

In common cathode I will connect the one common party ***( -ve )*** that connected to the ***ground*** to the ***ground*** of the micro and each LED I want to control it I will connect it to any pin from micro.

In common anode I will connect the one common party ***( +ve )*** that connected to the ***VCC*** to any pin in the micro and always  put high on this pin and Each LED I want to control it i will connect it to any pin from micro.

AMIT
Beyond Education

# Interfacing 7-Segments

How to check the type of my 7Segment (common anode or common cathode ) ?

***suppose that your 7SEG is common anode:-***

and connect the common party (+ve) to 5v and connect any other pin to the ground if any segment light up then your 7SEG is common anode if the LED won`t light up connect the common(-ve) to the ground and connect any other pin to the 5v if the segment light up then your 7SEG is common cathode.

# Interfacing 7-Segments

***Note that:-***

The 7SEG is same as the LED and the max power that can bear it is 100mW so you can not use the VCC of the kit because it out 350mA then the power will be

5v * 350mA > 100mW then the LED will Burn .

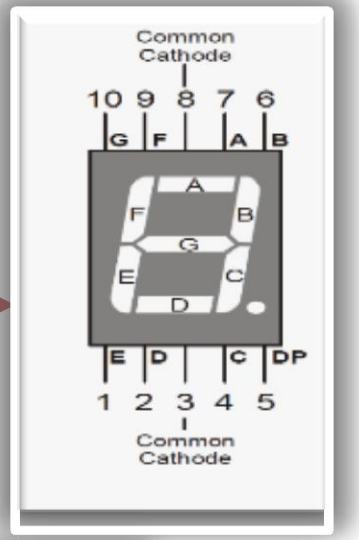
So I will out 5v from any pin of the micro .

# Interfacing 7-Segments

**Coding of 7-Segments :**

7–segment is common cathode :

Assuming Connecting the 7-Segment lines ( a to g ) to PD0 to PD6 in the microcontroller kit



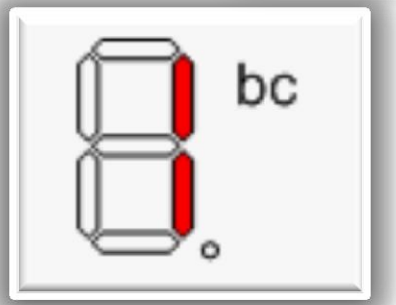

DDRD=255 ;

Configure PORTD in output mode

PORTD=0b00000110 ;

Set second and third pin in PORTD to carry 5v.

output on 7-segement




**then**

PORTD=0b01011011 ;

Set D0 , D1 , D3 , D4 & D6 in PORTD to carry 5v that connected by A,B,D,E,G

output on 7-segement


AMIT
Beyond Education

# 7-Segment Truth Table

| s | BCD | G | F | E | D | C | B | A |
|---|---|---|---|---|---|---|---|---|
| 0 | 0000 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0001 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 2 | 0010 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 3 | 0011 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 4 | 0100 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 5 | 0101 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 6 | 0110 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 7 | 0111 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 8 | 1000 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 9 | 1001 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |

# LAB 5

write a code to display on 7-segement numbers from 0 to 9 with delay 1 second before changing number.

AMIT
Beyond Education

# THANK YOU!

AMIT
Beyond Education