Course Name: Micro Processors and Micro Controllers

Topic Name: 8051 Microcontrollers

Dr. Appalabathula Venkatesh M.Tech, Ph.D
Assistant Professor,
Department of EEE, ANITS

Cos

Course Objectives:

At the end of the course, students will be able to

- **CO1** (BL-2) Describe the architecture and various addressing modes of a typical 8085 microprocessor
- **CO2** (BL-4) Classify different memory devices to Discuss the interfacing between memory and 8085 microprocessor
- **CO3** (BL-3) Describe the architecture of a typical 8086 microprocessor to illustrate the general bus operations
- **CO4** (BL-3) Describe the various peripheral devices and show how the peripherals (8259,8251 & 8253) are interfaced with Microprocessor.
- **CO5** (BL-4) Use the architecture of 8051 microcontroller and illustrate how 8051 is interfaced with advanced applications.

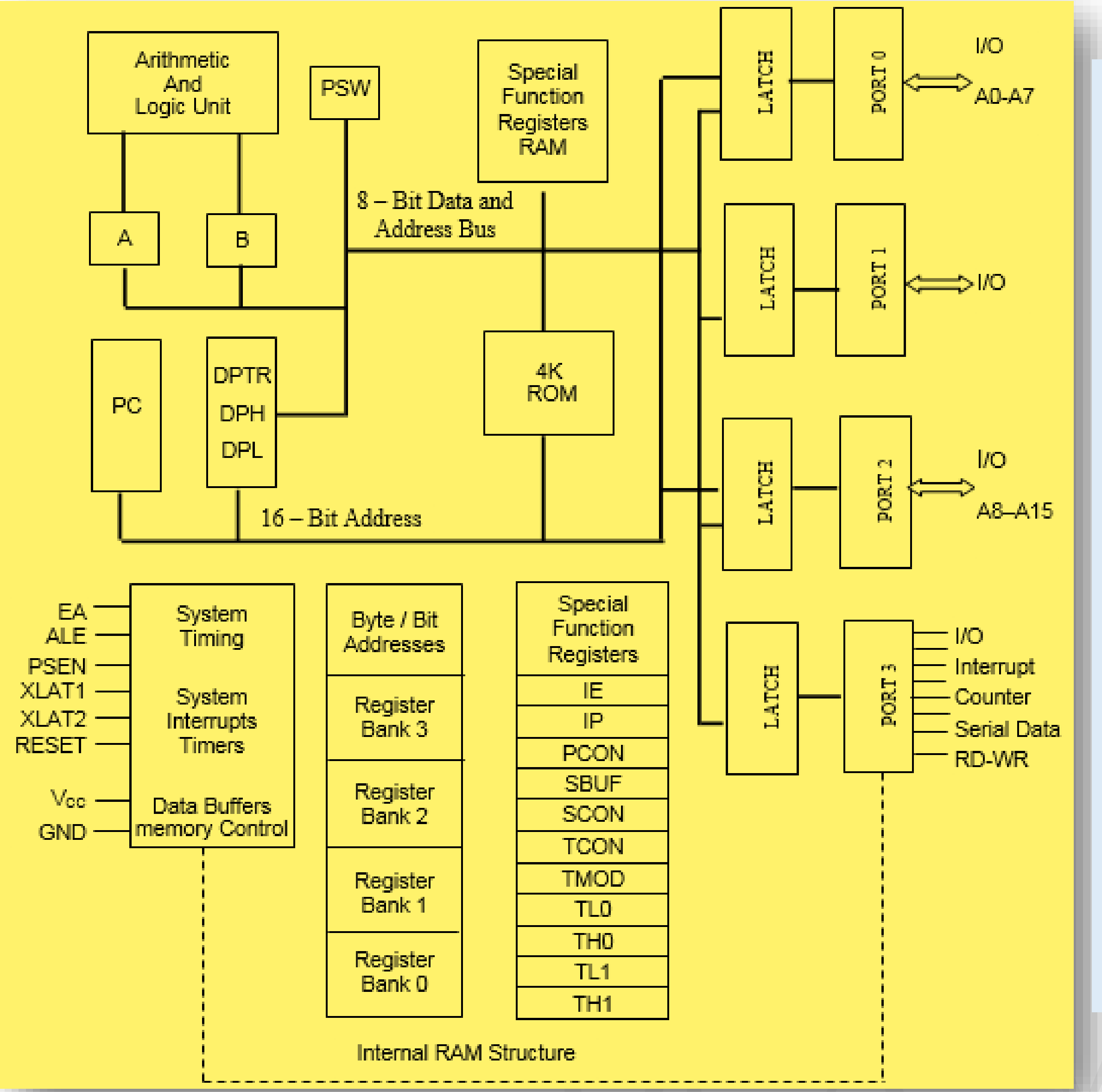
Syllabus

UNIT 5

8051 Microcontroller and Its Application

Pin Diagram and Architecture of 8051, Applications of microcontroller, Memory organization, Difference between Microprocessor and Microcontroller, Interfacing 8051 with LEDs, Seven segment display and Push button, ADC and DAC interfacing.

Architecture of 8051



1.CPU (Central Processing Unit):

1. The 8051 microcontroller is an 8-bit processor with a single accumulator (ACC) register and a B register.
2. It has a 16-bit program counter (PC) to keep track of the address of the next instruction to be fetched.

2.Registers:

1. Accumulator (ACC): Used for general-purpose arithmetic and logic operations.
2. B Register: Another general-purpose register for certain arithmetic operations.
3. Data Pointer (DPTR): A 16-bit register used as a pointer for data memory access.
4. Program Counter (PC): A 16-bit register that holds the address of the next instruction.
5. Stack Pointer (SP): An 8-bit register used to manage the program stack.

3.Memory Organization:

1. Program Memory (ROM): Typically 4 KB in size, where machine code instructions are stored. The lower 4 KB (0000H to 0FFFH) is used for standard 8051 variants.
2. Data Memory (RAM): 128 bytes of internal RAM for temporary data storage.
   1. General-Purpose RAM (from 00H to 7FH).
   2. Bit-Addressable RAM (from 20H to 2FH).
3. Special Function Registers (SFRs): These are used for controlling various functions of the microcontroller and are located within the data memory space.

4.I/O Ports:

1. The 8051 microcontroller has four parallel I/O ports (P0, P1, P2, P3) that can be configured as input or output.

Architecture of 8051

5. Interrupts:

1. The 8051 supports external and internal interrupts. It has two external interrupt pins (INT0 and INT1) and two internal interrupts (Timer 0 and Timer 1).
2. Interrupts can be enabled or disabled using specific bits in the interrupt enable registers.

6. Timers and Counters:

1. The 8051 microcontroller has two 16-bit timers/counters, Timer 0 and Timer 1. These can be used for timing and counting purposes.
2. The timers can be configured to operate in different modes, including timer mode and counter mode.

7. Serial Communication:

1. The 8051 microcontroller has a built-in UART (Universal Asynchronous Receiver/Transmitter) for serial communication.

8. Instruction Set:

1. The 8051 instruction set is rich and includes a variety of arithmetic, logic, and control instructions.

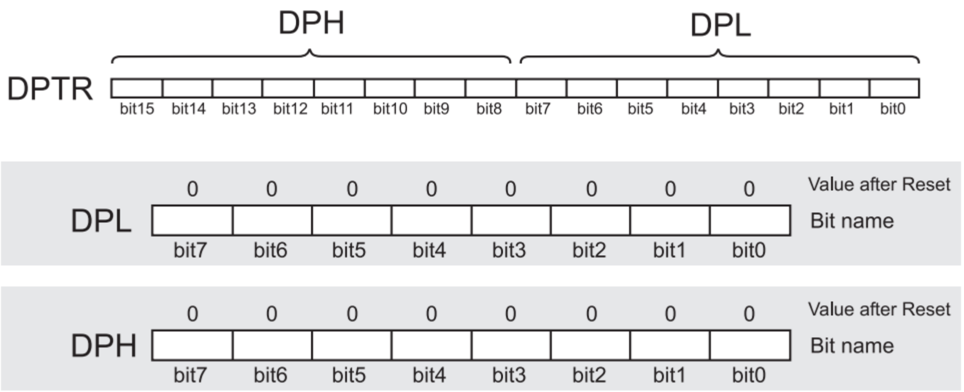
Data pointer (DPTR)

They consist of two separate registers DPH (data pointer high) and (data pointer low).

Their 16 bits are used for external memory addressing.

They may be handled as a 16 bit register or as two independent 8 bit registers.

The DPTR registers are usually used for storing data and immediate results which have nothing to do with memory locations



Architecture of 8051

Flags and PSW

Program Status Word (PSW)*

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	–	P
D7	D6	D5	D4	D3	D2	D1	D0

RS1	RS0	Bank Select
0	0	Bank – 0
0	1	Bank – 1
1	0	Bank – 2
1	1	Bank – 3

Bit	**Flag**	**Description**
7	CY	Carry flag; used in arithmetic, jump, rotate and Boolean instructions.
6	AC	Auxiliary carry; used for BCD arithmetic
5	F0	User defined flag 0
4	RS1	Register bank select bit-1
3	RS0	Register bank select bit-0
2	OV	Overflow flag; used in arithmetic instructions
1	-	Reserved for future use
0	P	Parity flag; shows parity of register A; 1=odd parity

PSEN (Program Store Enable) and EA (External Access)

SP→It is incremented before data is stored during PUSH and CALL instructions and decremented after data is restored during POP and RET instructions

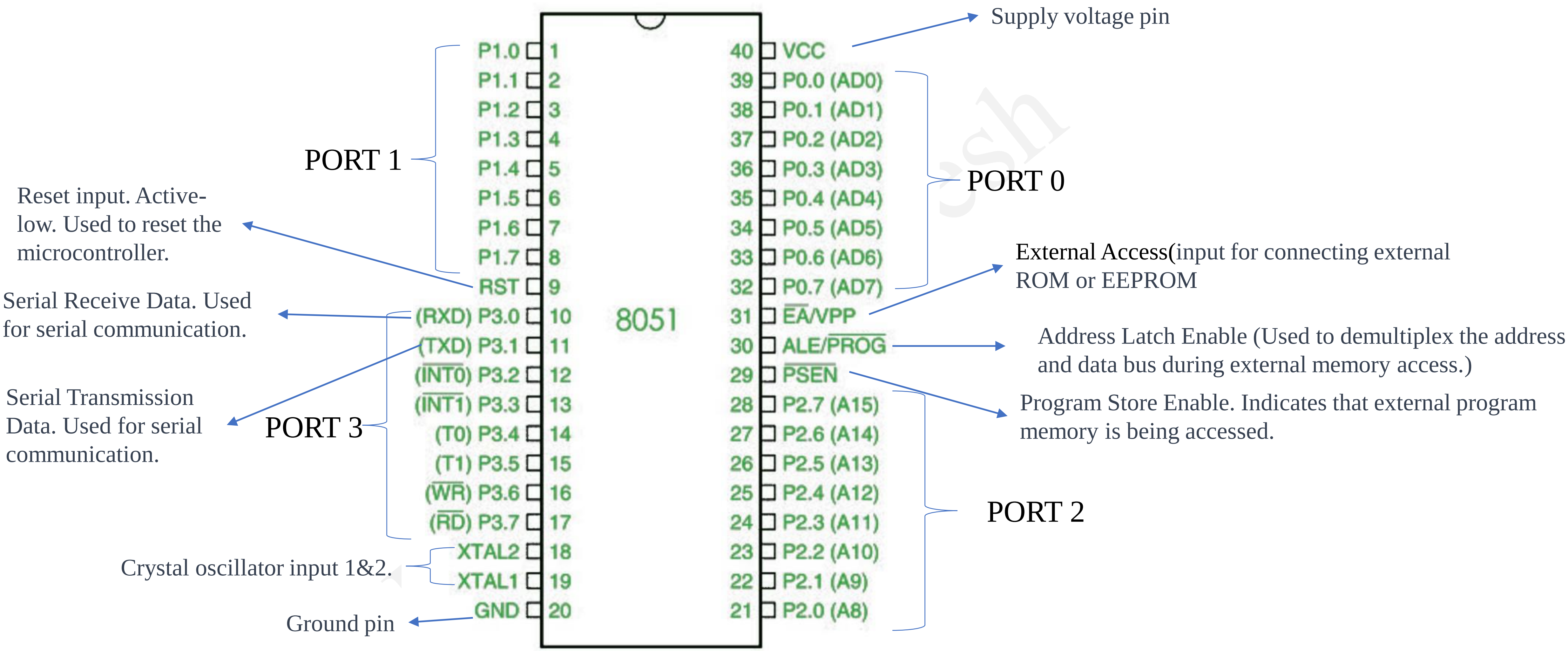
Special Function Registers (SFR's)

A→The Accumulator is an 8-bit register used for arithmetic and logic operations. It is one of the general-purpose registers in the CPU.
B Register→ 8-bit general-purpose register used for certain arithmetic operations.
DPTR→ 16-bit register used as a data pointer for accessing data in the external RAM. It is often used for accessing tables or arrays in the data memory.
PSW→The PSW is an 8-bit register that contains various flags and the stack pointer. Flags include carry, auxiliary carry, parity, overflow, and user-definable flags.
SP→It is an 8-bit register used to manage the program stack. It points to the top of the stack in the internal RAM.
PC→It is used to hold the address of memory location from which the next instruction is to be fetched

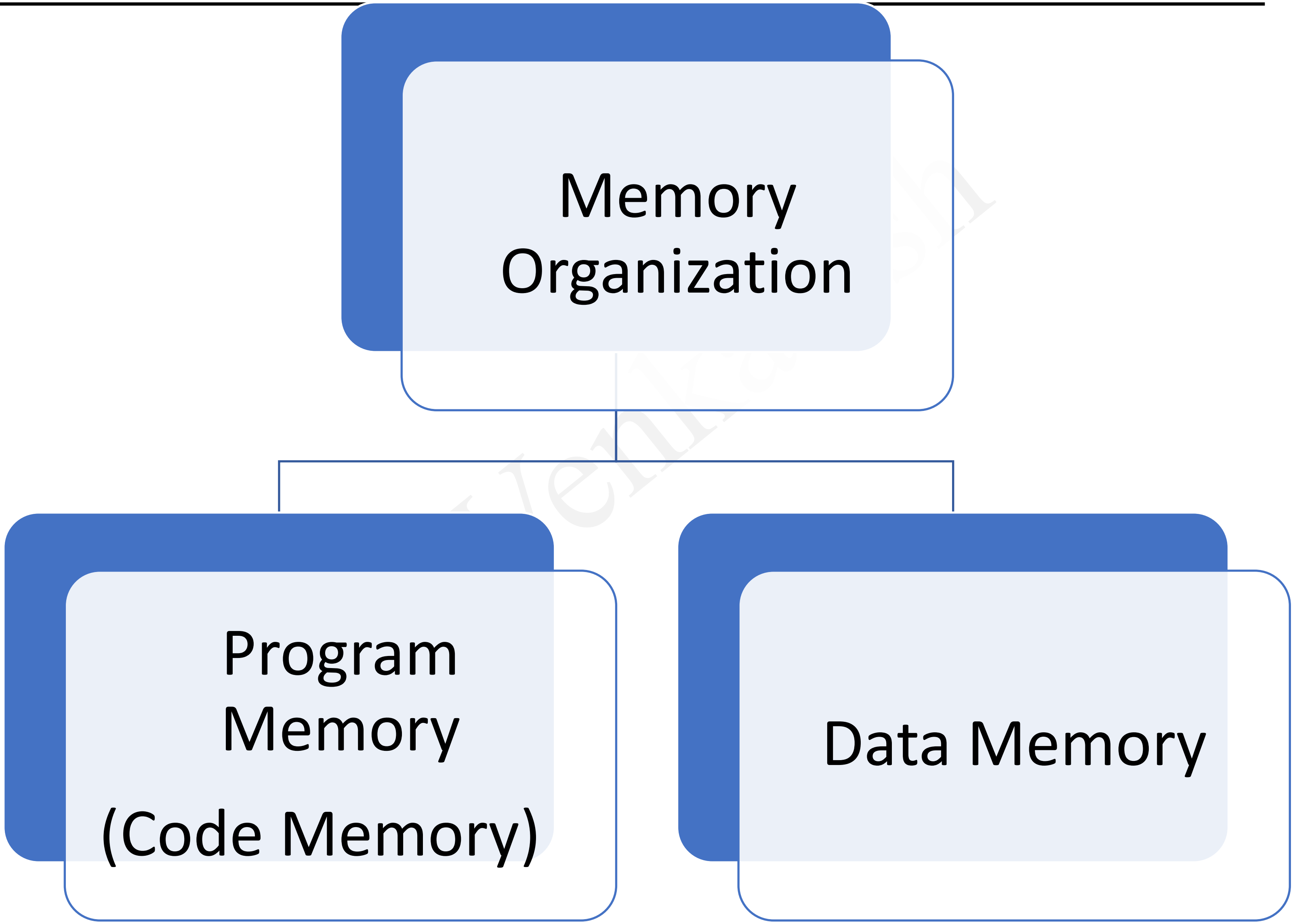
IE and IP registers→ control the interrupt system. IE enables or disables interrupts globally, while IP sets the priority level of each interrupt source.
TCON Register → TCON controls the operation and status of Timer 0 and Timer 1. It includes bits for timer control, external interrupts, and interrupt flags.
TMOD Register →TMOD sets the operating modes of Timer 0 and Timer 1, including timer or counter mode and various timer modes.
T0 and T1 Registers→These registers hold the values for Timer 0 and Timer 1 and are used in conjunction with the timers to generate timing and delay functions.
SCON and SBUF Registers:→SCON controls the serial communication modes, and SBUF holds the data to be transmitted or received

SFR	Description	Address
A*	Accumulator	E0
B*	B Register	F0
PC	Program Counter	—
DPTR		
DPH	Data Pointer Higher Byte	83
DPL	Data Pointer Lower Byte	82
SP	Stack Pointer	81
PSW*	Program Status Word	D0
P0*	Port 0	80
P1*	Port 1	90
P2*	Port 2	A0
P3*	Port 3	B0
TH0	Timer 0 Higher Byte	8C
TL0	Timer 0 Lower Byte	8A
TH1	Timer 1 Higher Byte	8D
TL1	Timer 1 Lower Byte	8B
TCON*	Timer Control	88
TMOD	Timer Mode Control	89
SBUF	Serial Data Buffer	99
SCON*	Serial Port Control	98
PCON	Power Mode Control	87
IE*	Interrupt Enable	A8
IP*	Interrupt Priority	B8

PIN description of 8051



8051 Microcontroller Memory Organization



PIN description of 8051

1.Program Memory (Code Memory):

1. The 8051 microcontroller has a 4 KB program memory (ROM) for storing the program code. This memory is used to store the machine code instructions that the microcontroller fetches and executes.
2. The program memory is organized as 8 KB, but only the lower 4 KB (addresses 0000H to 0FFFH) are used for program storage in standard 8051 variants.

2.Data Memory:

1. The data memory in the 8051 is used for storing data during program execution. It is divided into two parts: the Internal RAM and the bit-addressable area.
2. Internal RAM: The 8051 microcontroller has 128 bytes of internal RAM (from address 00H to 7FH) for general-purpose data storage.
3. Bit-Addressable Area: The addresses 20H to 2FH in the internal RAM are bit-addressable, allowing individual bits to be accessed and manipulated.

3.Special Function Registers (SFRs):

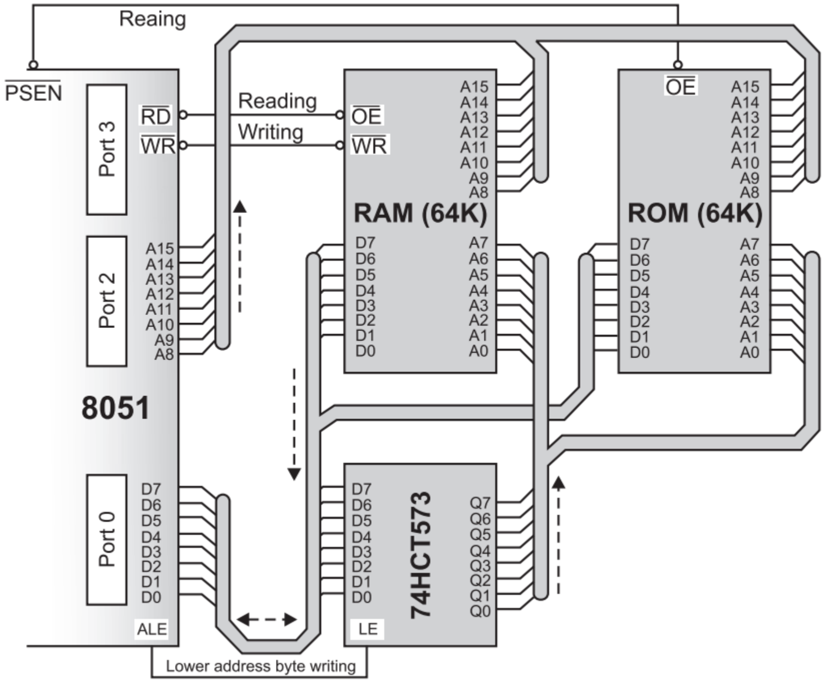
1. SFRs are a set of registers used for controlling and configuring the 8051 microcontroller's various functions. These registers are located in the data memory space.
2. Examples of SFRs include the Accumulator (ACC), B register, Data Pointer (DPTR), program status word (PSW), control registers for timers and interrupts, and more.

4.External Data Memory:

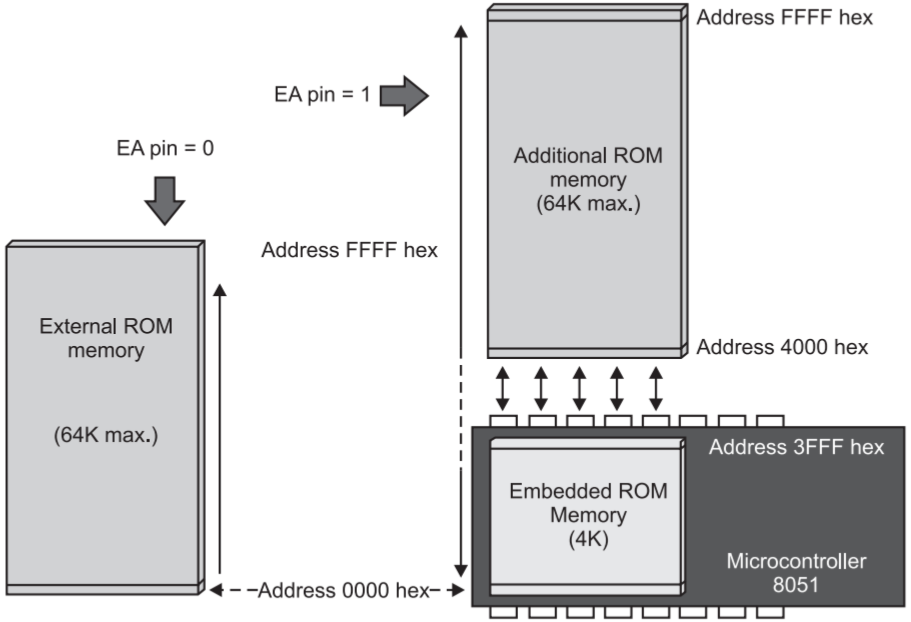
1. The 8051 microcontroller can also interface with external data memory devices through the MOVX (move external) instruction.
2. External data memory is not part of the internal memory of the 8051 but can be added to expand the data storage capacity.

External Memory

I/O ports P2 and P3 are used for the two external memory chips addressing and data transmission.



How does the microcontroller handle external memory depend on the pin EA logic state:

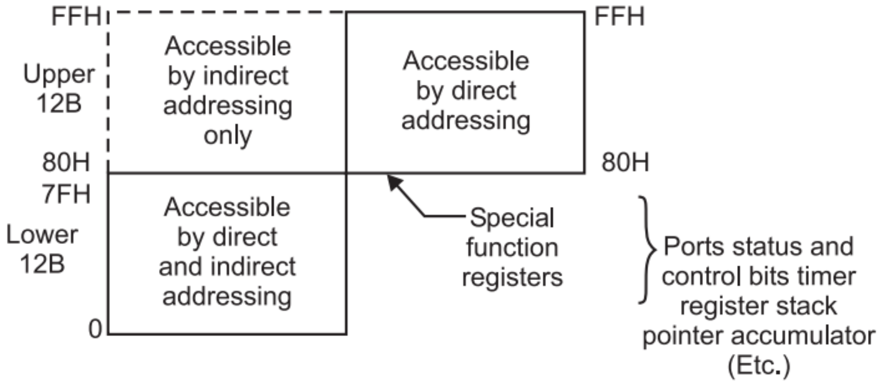


Additional Memory

EA = 0 In this case, internal program memory is completely ignored, only a program stored in external memory is to be executed.
EA = 1 In this case, a program from builtin ROM is to be executed first (to the last location). Afterwards, the execution is continued by reading additional memory.

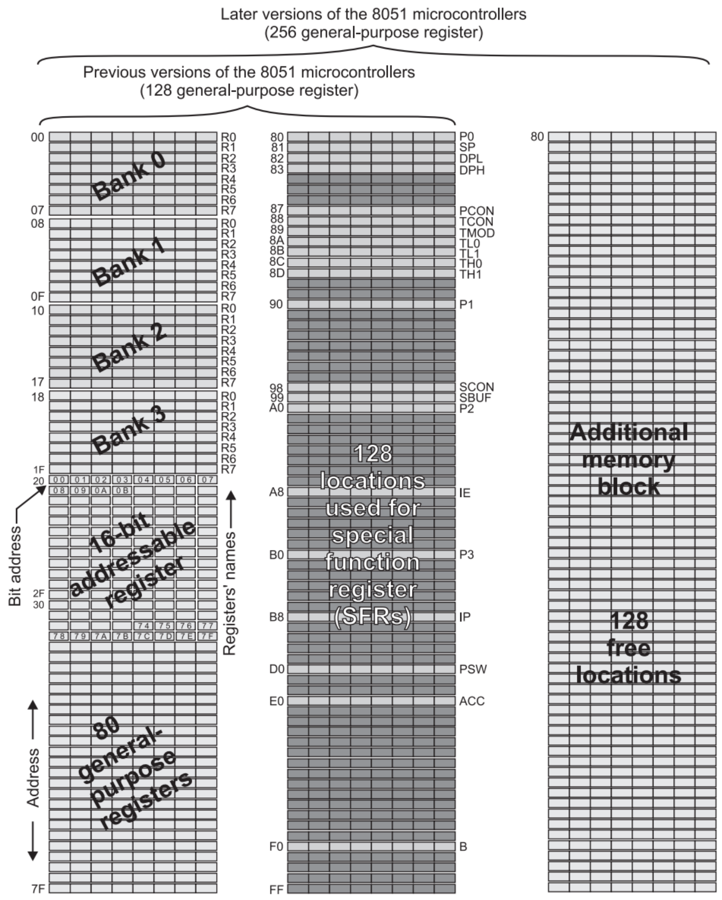
Internal Memory

First 128 register this part of ram is divided in several blocks.



Internal RAM

Additional Memory Block of Data Memory



The first bank consists of four banks each including 8 registers designated as R0–R7, the four register banks are bank0, bank1, bank2 and bank3.location available to the user occupy memory space with addresses from 00–FF. Next memory block is bit addressable which means that each bit being there has its own address from 00–7FH. Since, there are 16 such registers, this block contain in total of 128 bits with separate addresses. The third group of registers occupies addresses 2FH–7FH and does not have any special purpose or features.

LED Interfacing with Microcontroller(With Port1)

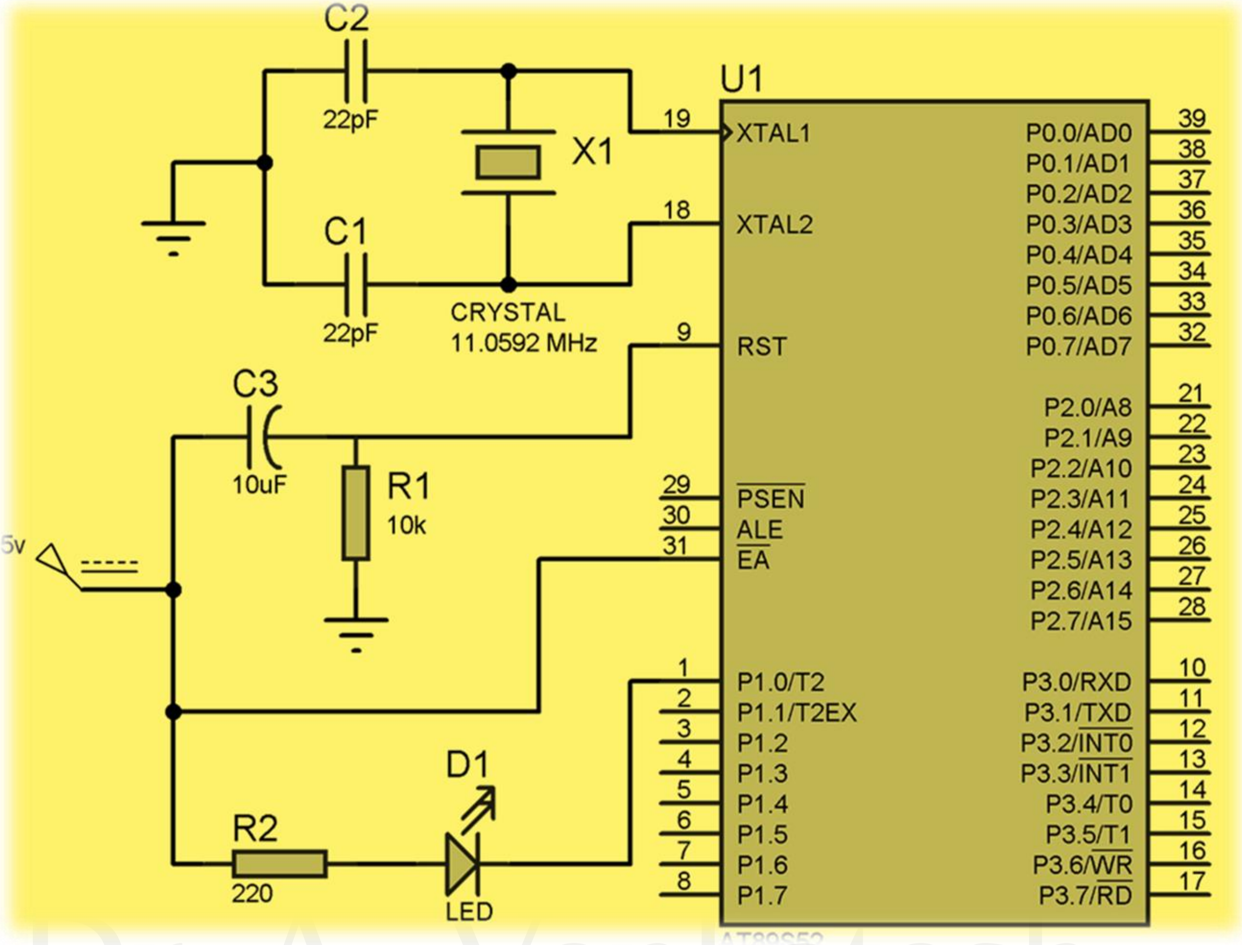
Pin 31 (EA) → Vcc, which is an active low pin. This should be connected to Vcc when we are not using the any external memory.

Pin 30(ALE) and pin 29 (PSEN) → to connect microcontroller to the external memory.

Pin 9 (RST)→ reset PIN, used to reset the microcontroller and program again starts from the beginning. It resets the microcontroller when connected to HIGH. We have used standard reset circuitry, 10k ohm resistor and 1uF Capacitor to connect the RST pin.

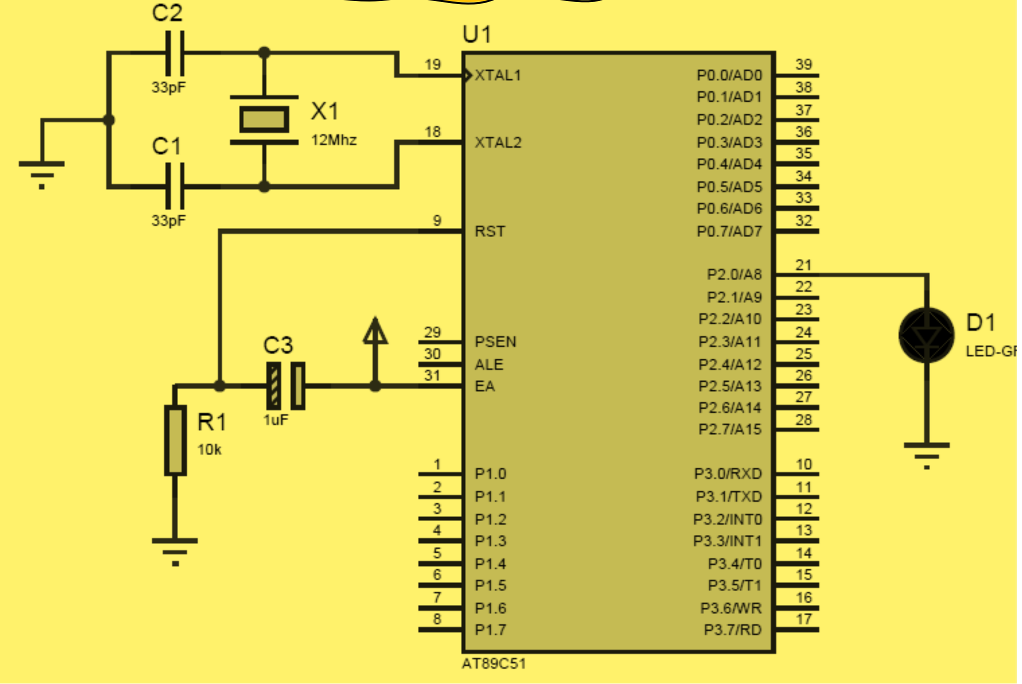
Here the LED run on the negative logic like when, pin P1_0 is 1 then LED will be tuned OFF and when pin output is 0 then LED will be turned ON. When PIN output is 0, it behaves like ground and LED glows.

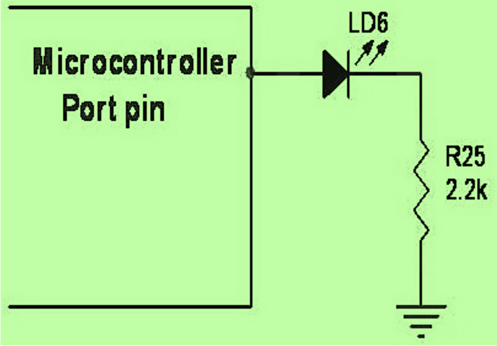
XTAL1&XTAL2 →Crystal Oscillator → to synchronise all the events.



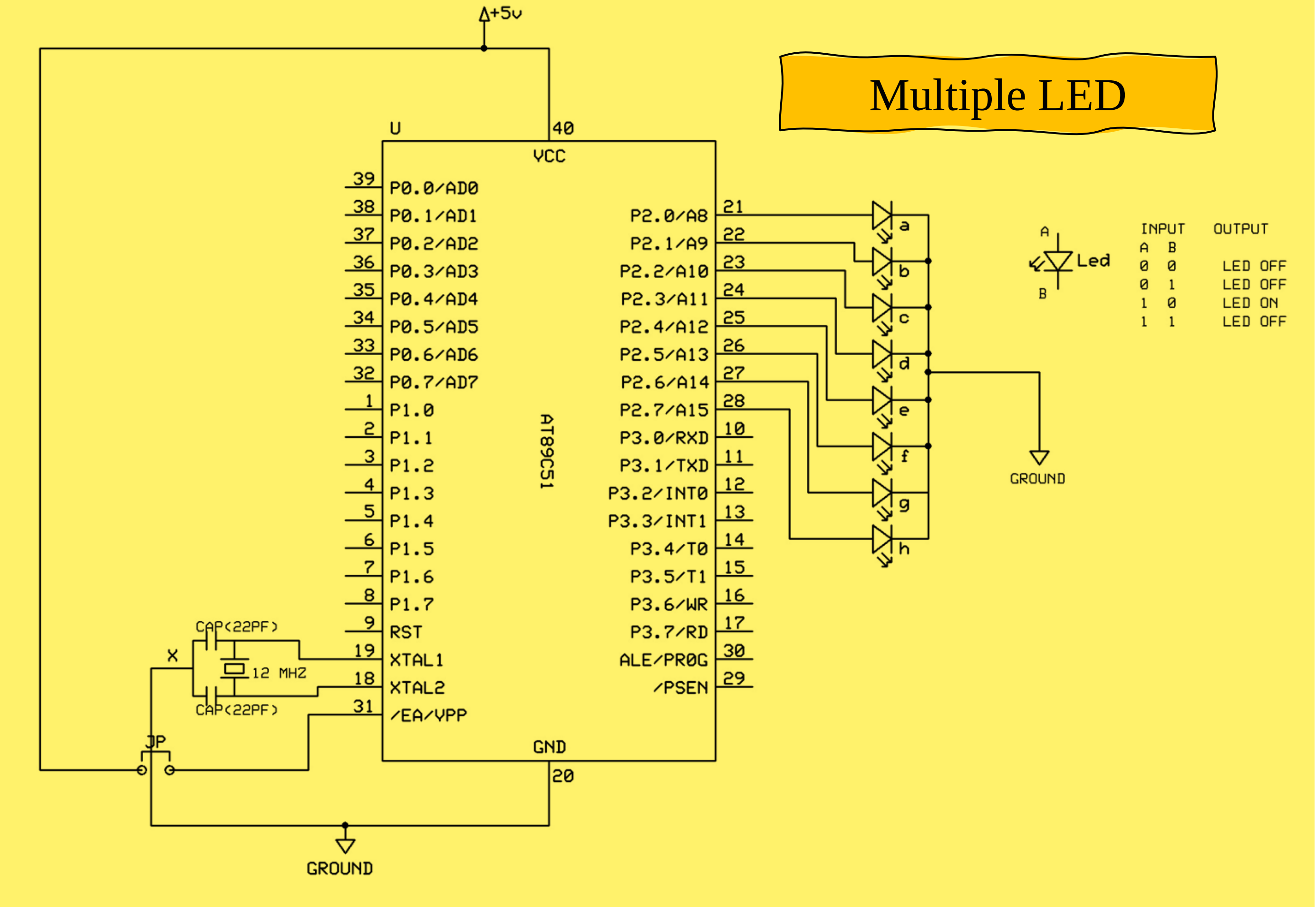
LED Interfacing with Microcontroller (with Port2)

Single LED



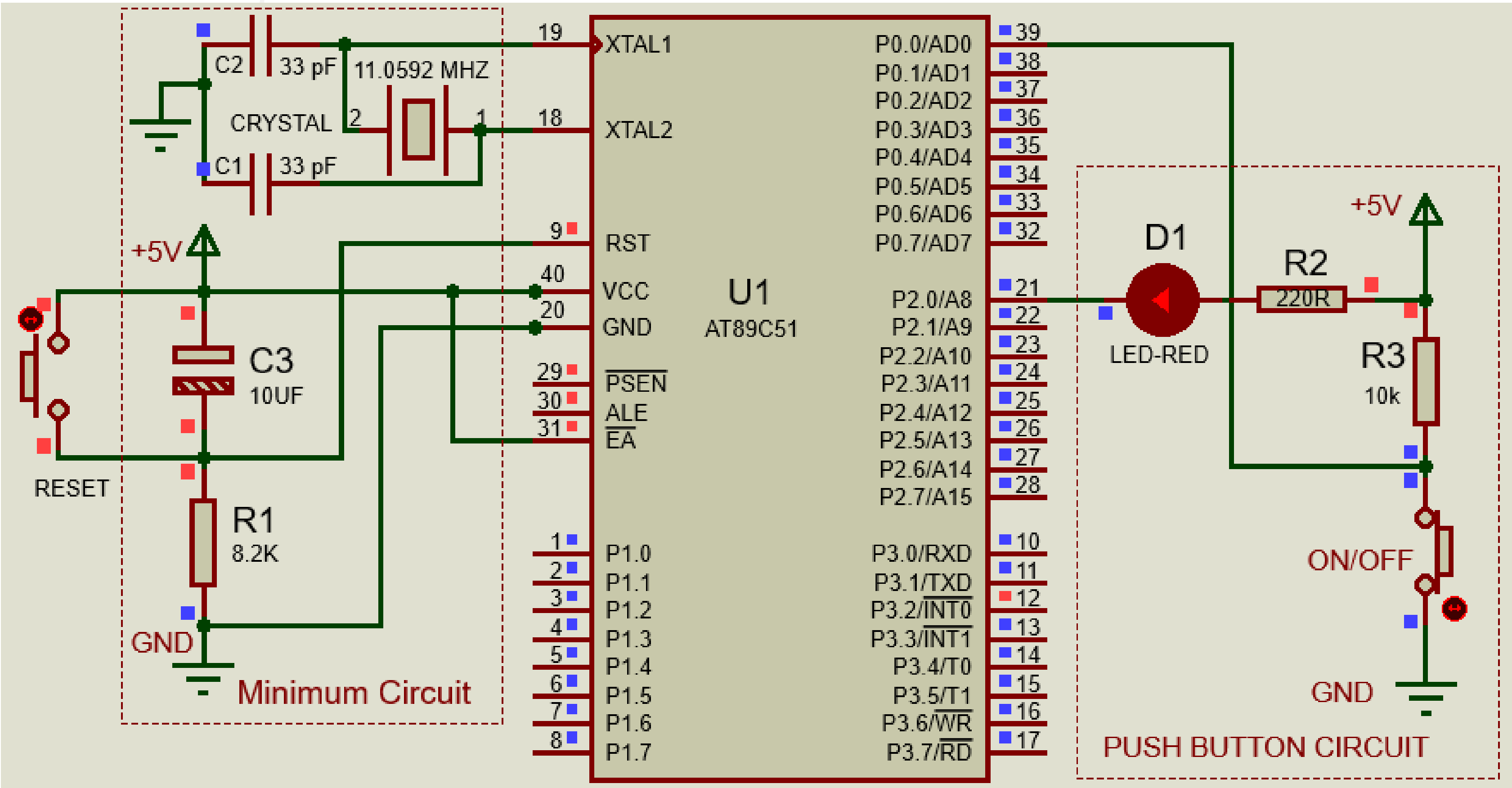


Multiple LED



Push button Interfacing with Microcontroller

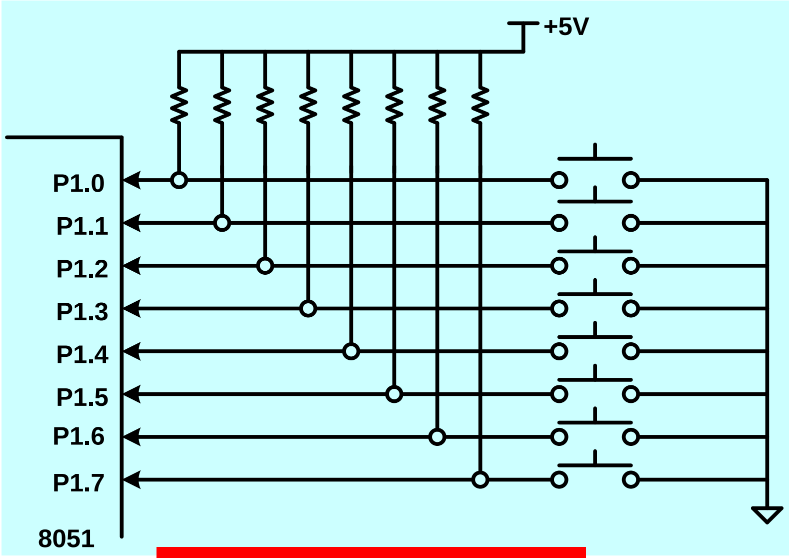
A **push-button** (also spelled pushbutton) or simply button or tactile switch is a component that connects two points in a circuit when you press it.



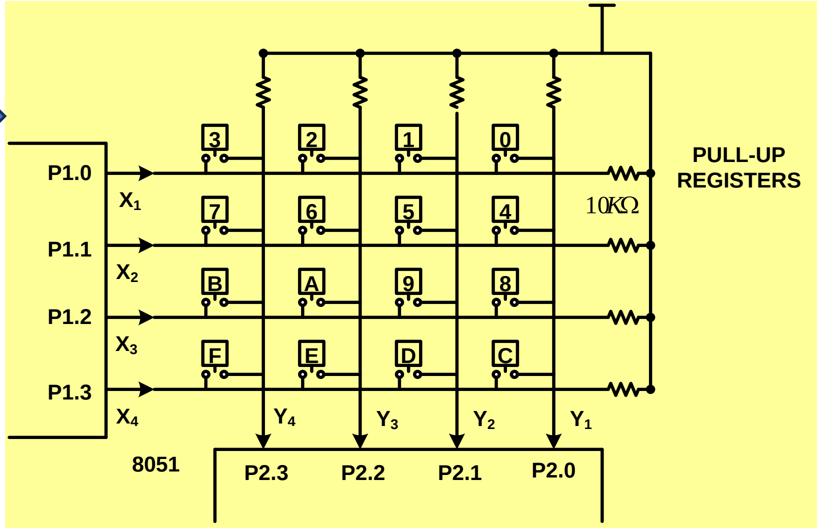
A push button switch is connected to PORT-0 PIN-0. The SW is connected externally with a pull-up resistor since there is no internal pull-up resistor for port 0. The other end of the push button is connected to ground.

The voltages at the pins will be VCC (Approx) when the button is not pushed. The input pin will get grounded on pressing the button.

Keyboard Interfacing with Microcontroller



Solution-1



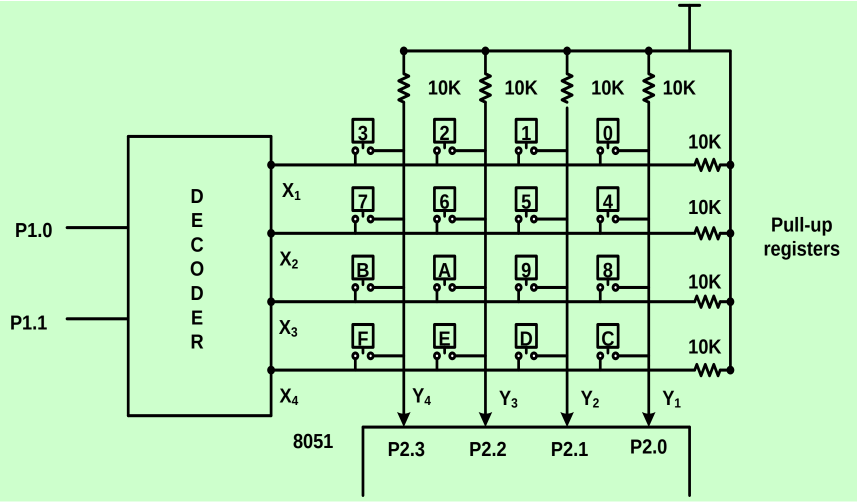
- Keys are organized in two-dimensional matrix to minimize the number of ports required for interfacing

Drawback?

- One key per port line

Solution-2

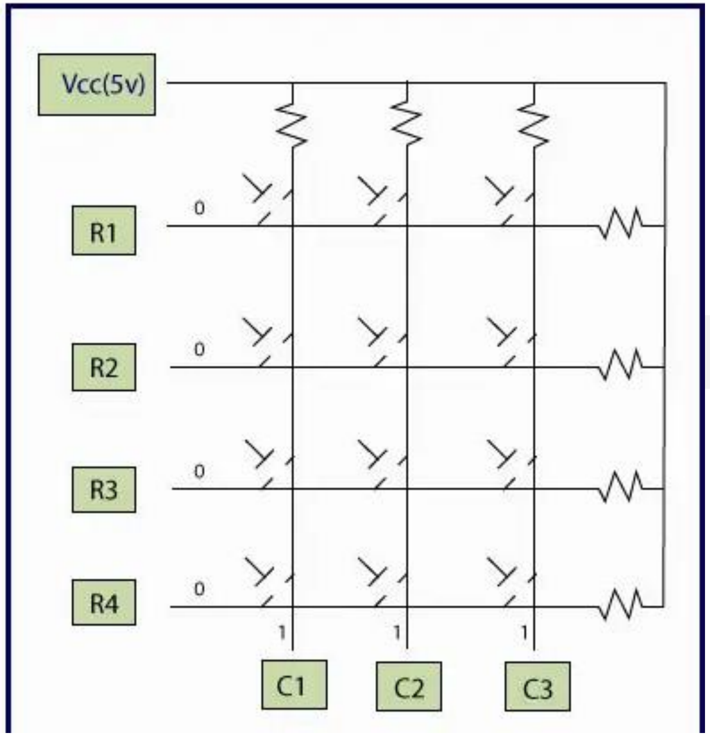
- Use of decoder further reduces the number of port lines required



Key Board Interfacing with Microcontroller

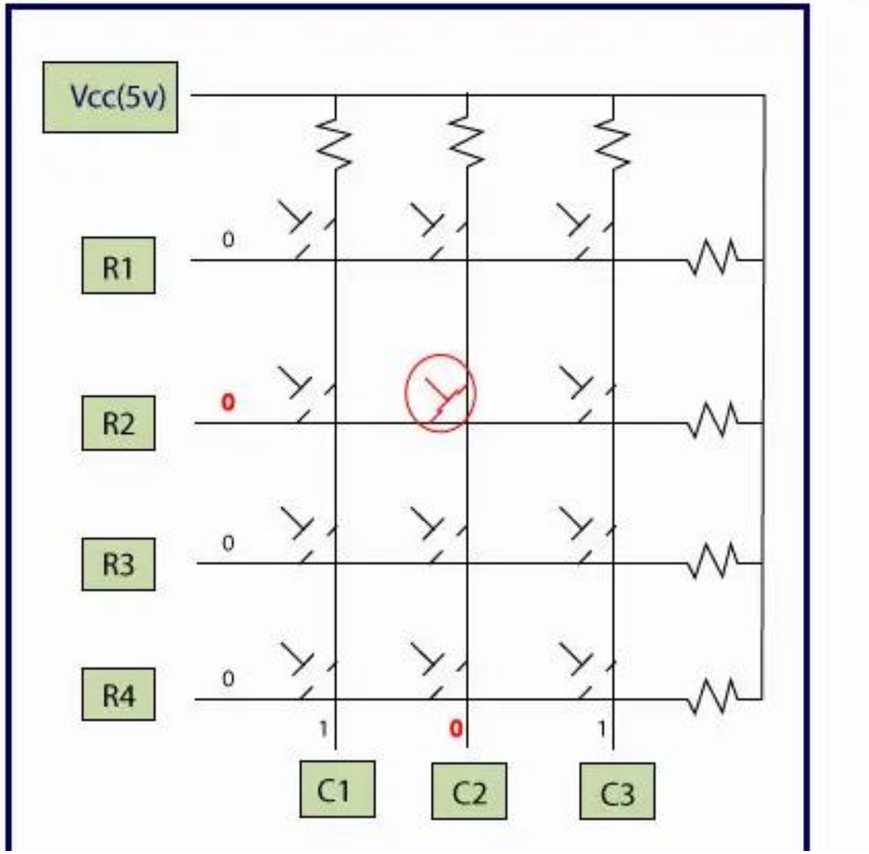
Initial State of Rows and Columns?

Initially all the rows are set to zero by the controller and the columns are scanned to check if any key is pressed. In case no key is pressed the output of all the columns will be high.



Locating Columns No of Pressed Key?

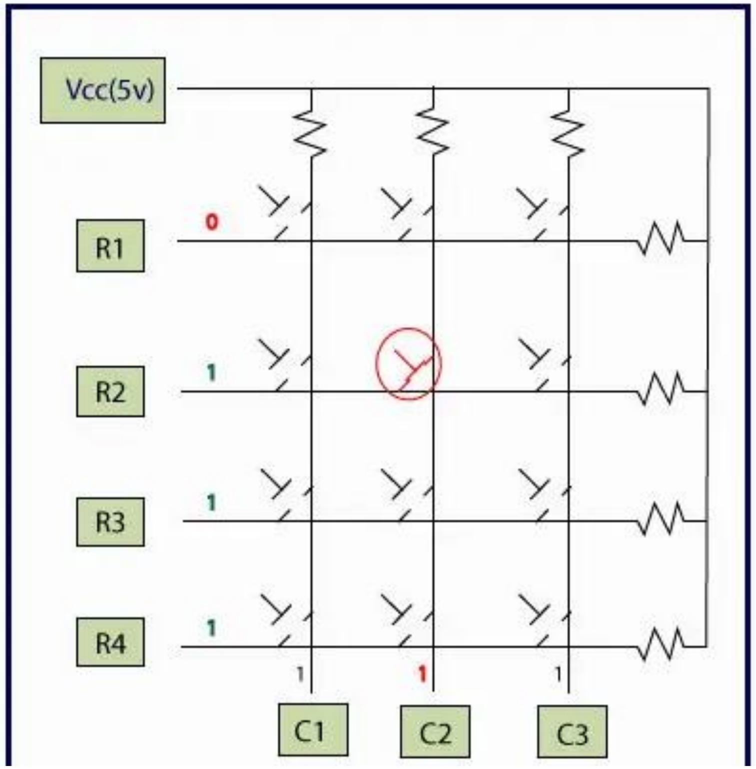
Whenever a key is pressed the row and column corresponding to the key will get short, resulting in the output of the corresponding column goes to go low (since we have made all the rows zero). This gives the column number of the pressed key



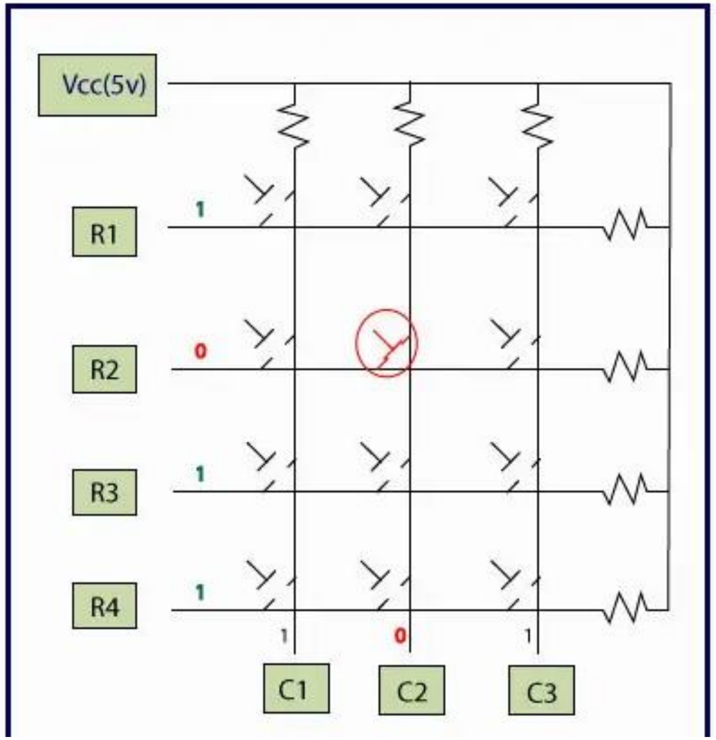
Key Board Interfacing with Microprocessor-8279

Row No. Checking of Pressed Digit?

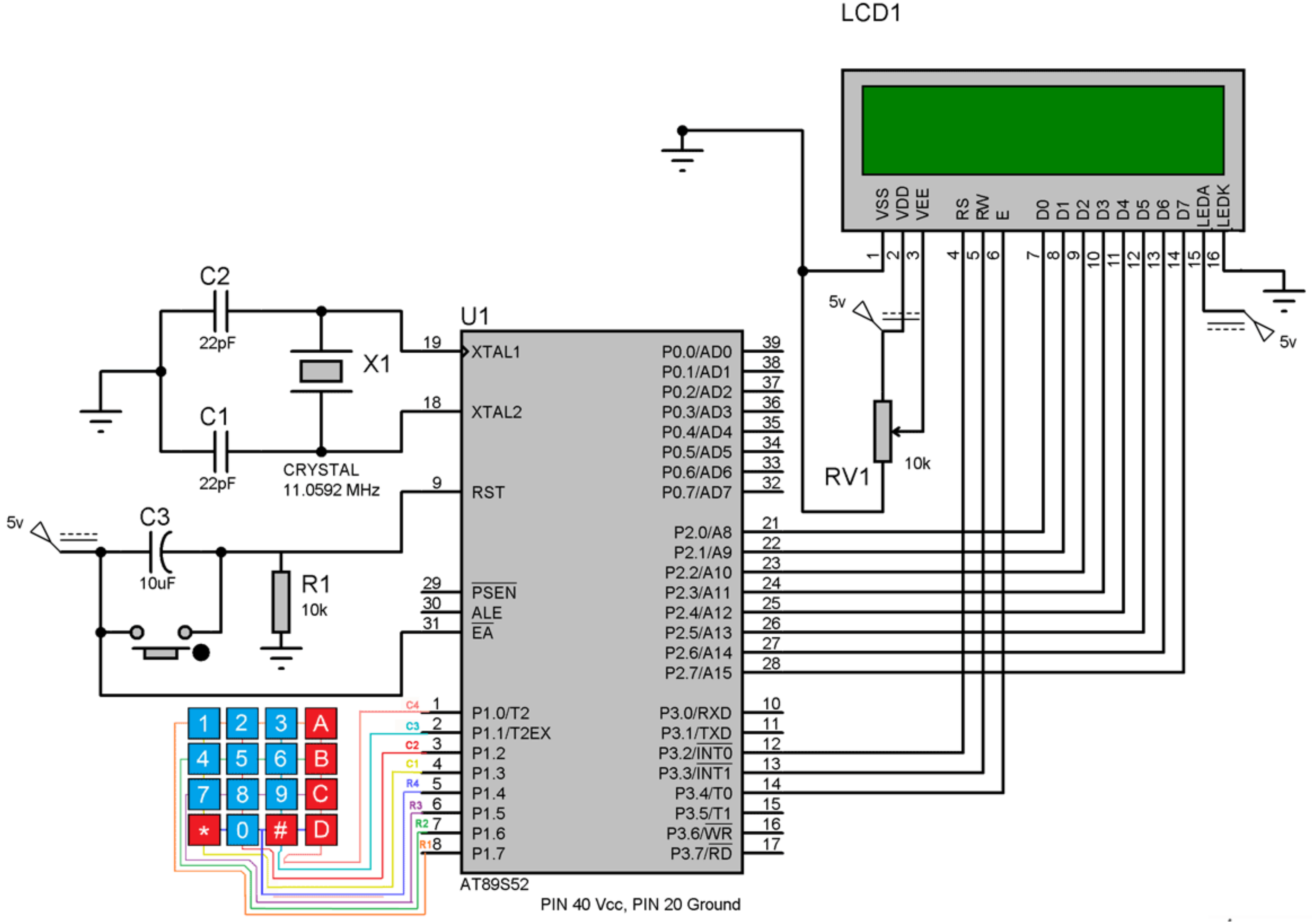
Once the column number is detected, the controller set's all the rows to high. Now one by one each row is set to zero by controller and the earlier detected column is checked if it becomes zero. The row corresponding to which the column gets zero is the row number of the digit.



Displaying Row and Column No. of Pressed Key?

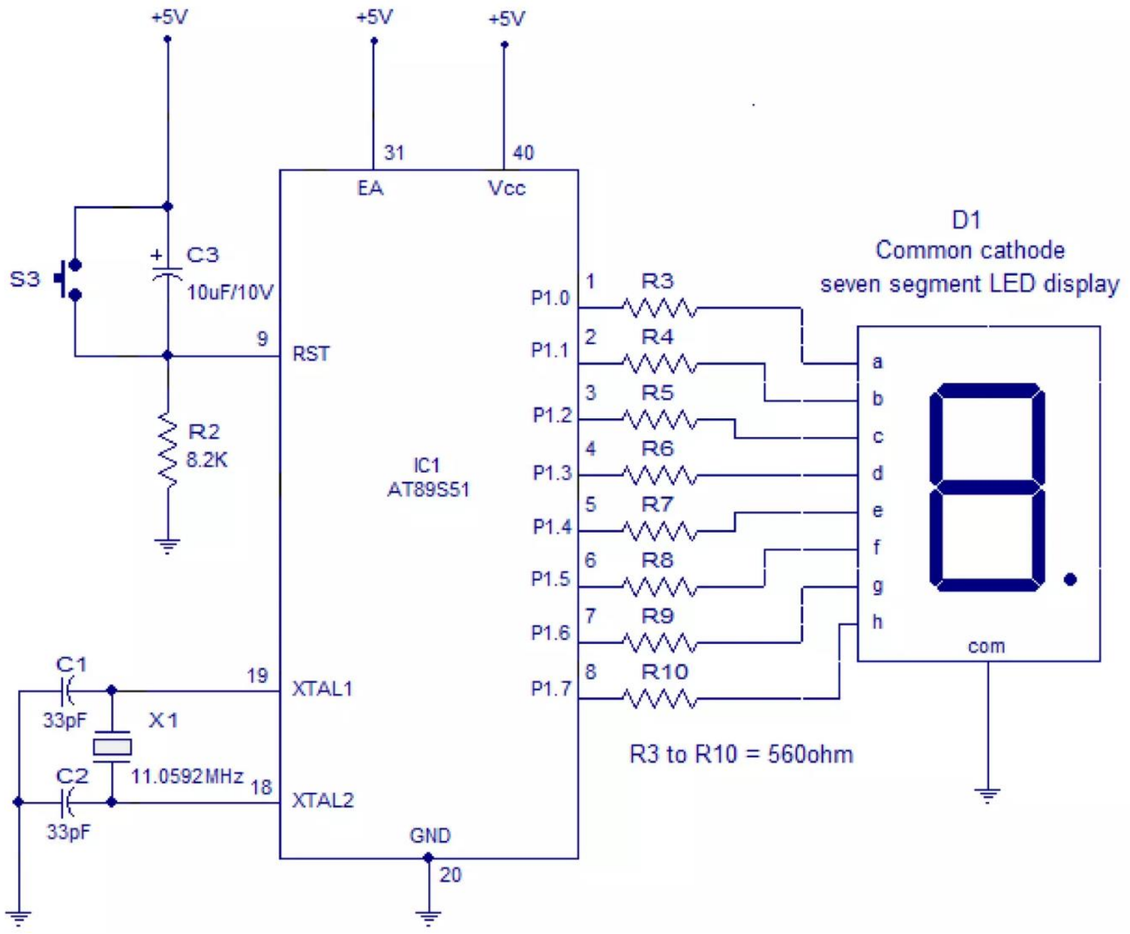


Key Board Interfacing with Microcontroller



Dr. Appalabathula Venkatesh, Assistant Professor,
EEE Dept., ANITS

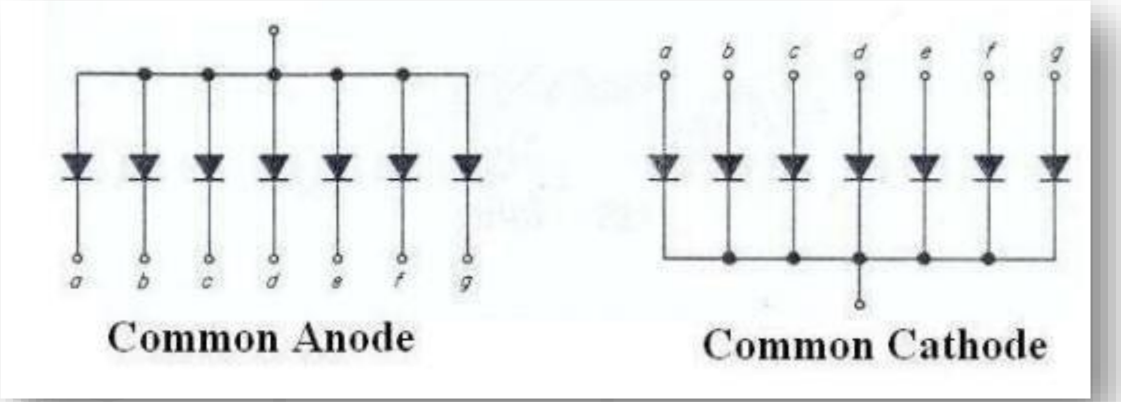
7Segment Interfacing with Microcontroller

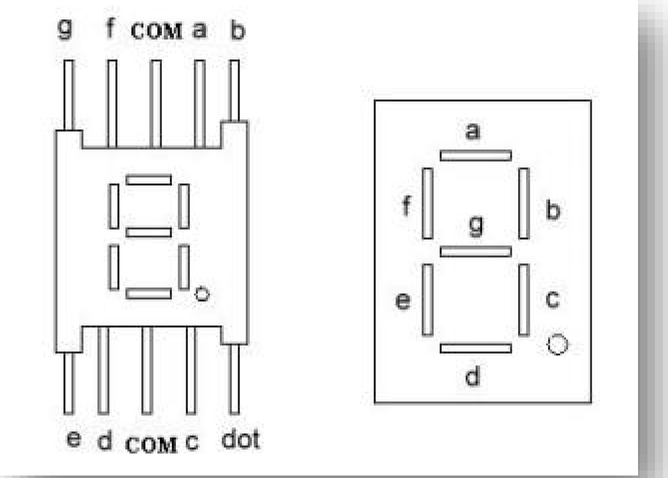


- ➢ Seven segment displays are used to indicate numerical information.
- ➢ Seven segment display mainly consists of 8 LEDs internally.
- ➢ In these LEDs, 7LEDs are used to indicate the digits 0 to 9
- ➢ 1LED is used to indicate decimal point.

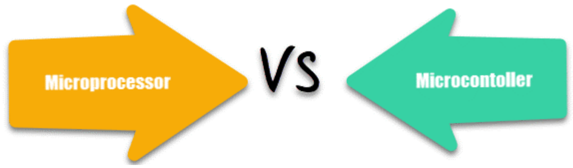
7Segment Interfacing with Microcontroller

h	g	f	e	d	c	b	a	Digit
0	0	1	1	1	1	1	1	0
0	0	0	0	0	1	1	0	1
0	1	0	1	1	0	1	1	2
0	1	0	0	1	1	1	1	3
0	1	1	0	0	1	1	0	4
0	1	1	0	1	1	0	1	5
0	1	1	1	1	1	0	1	6
0	0	0	0	0	1	1	1	7
0	1	1	1	1	1	1	1	8
0	1	1	0	0	1	1	1	9





Differences between Microprocessor and Microcontroller



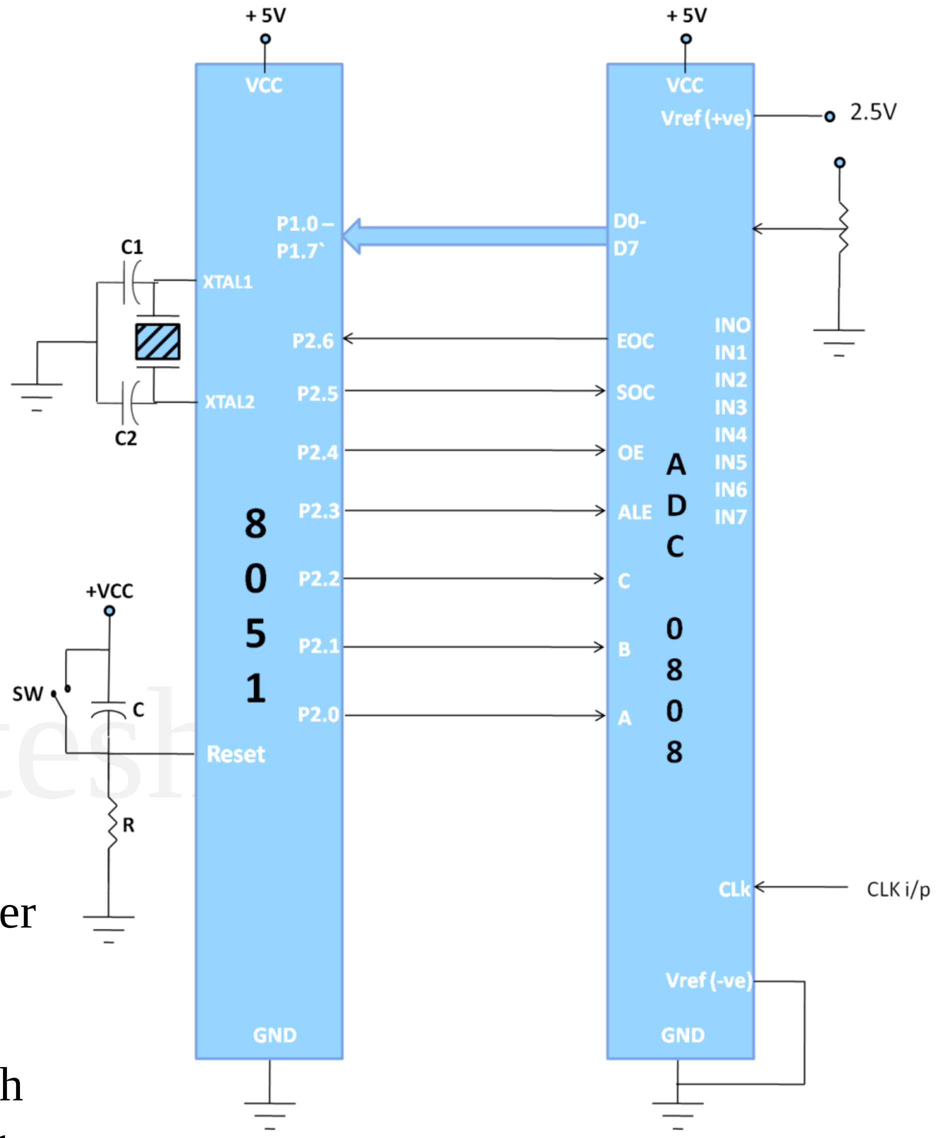
	Microprocessor	Microcontroller
Definition	A processing device implemented on a single chip is called a microprocessor.	A microcontroller is an electronic system which consists of a processing element, a small memory (RAM, ROM, EPROM), I/O ports, etc. on a single chip.
System Architecture	The microprocessor is the heart of a Computer system.	The microcontroller is the heart of an embedded system.
Interfacing of components	Memory and I/O component are connected externally.	Memory and I/O output component are present internally.
Circuit complexity	Since memory and I/O have to be connected externally, the circuit becomes large	Since memory and I/0 are present internally, the circuit is small
Cost	It increases the overall cost of the system high.	The overall cost of the system is less.
Power Saving feature	Most of the microprocessors do not have power saving features	Most of the microcontrollers offer power-saving mode.
Speed of operation	Since memory and I/O components are all external, each instruction will need an external operation, hence it is relatively slower.	Since components are internal, most of the operations are internal instruction, hence speed is fast
Clock speed	Higher clock speed, typically greater than 1 GHz	Lower clock speed, typically less than 100 MHz
Component Integration	Less component integration.	High component integration.
Resisters	Microprocessors have less number of registers, hence more operations are memory based	Microcontrollers have more registers, hence the programs are easier to write.
Instruction set	Fixed instruction set-It's complex and expensive, with a large number of instructions to process.	More flexible-It's simple and inexpensive with less number of instructions to process.
Application	It is mainly used in personal computers.	It is used mainly in a washing machine, MP3 players, and embedded systems
Handling Boolean functions	Not capable of handling Boolean function	Capable of handling Boolean functions
Multifunction of pins	Less number of pins are multi functioned.	More number of pins are multifunctiona

ADC interfacing with 8051 Microcontroller

- The ADC 0808/0809 is the 8-bit analog to digital converter.
- It has 8 channels IN0-IN7

Pin A,B,C and ALE are used to select one of the analog channel of total 8 channels as shown below:

A	B	C	Channel
0	0	0	IN0
0	0	1	IN1
0	1	0	IN2
0	1	1	IN3
1	0	0	IN4
1	0	1	IN5
1	1	0	IN6
1	1	1	IN7



ADC 0808 has 3 control signals

SOC [Start of conversion]: When High to low signal is appears to this pin of ADC, ADC then starts conversion

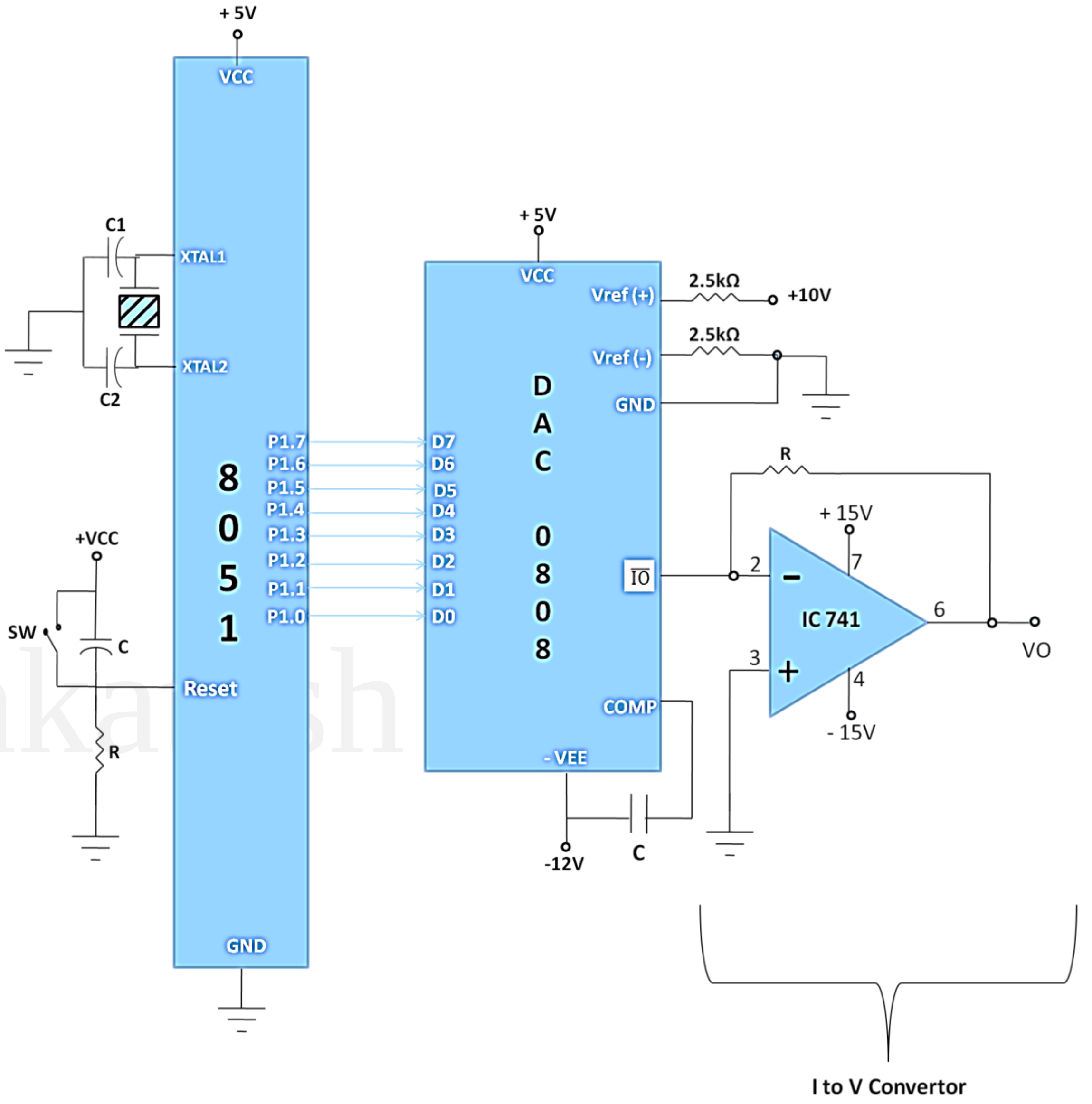
EOC [End of conversion]: ADC sends this high EOC signal to Micro-Controller to indicate completion of conversion.

OE [Output Enable]: When a high signal is applied to this pin, The output latch of ADC get enables and the converted data is then available to Micro-Controller.

DAC interfacing with 8051 Microcontroller

There are two methods of converting digital signals to analog signals.

These two methods are binary weighted method and R/2R ladder method



MC1408 (DAC0808) Digital to Analog Converter

In this chip the digital inputs are converted to current. The output current is known as I_{out} by connecting a resistor to the output to convert into voltage.

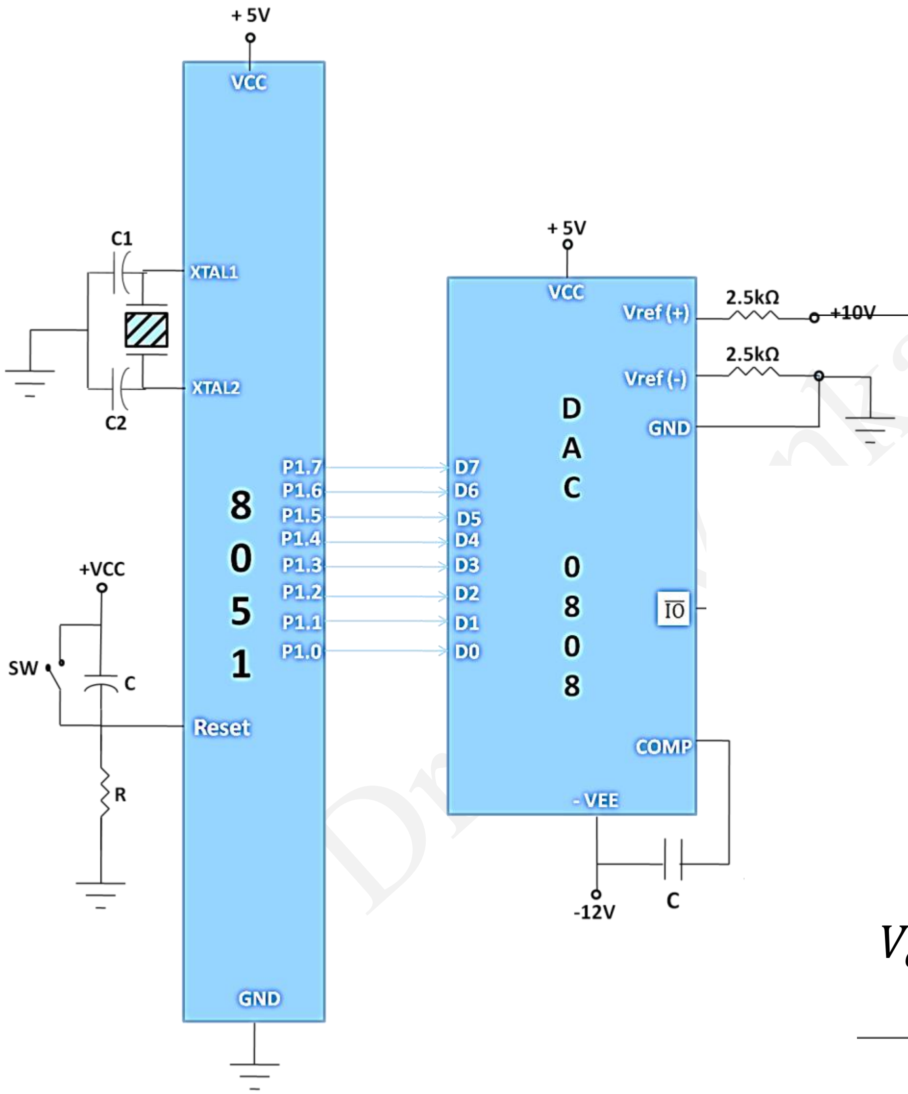
The total current provided by the I_{out} pin is basically a function of the binary numbers at the input pins D_0 - D_7 (D_0 is the LSB and D_7 is the MSB) of DAC0808 and the reference current I_{ref}.

The following formula is showing the function of I_{out}

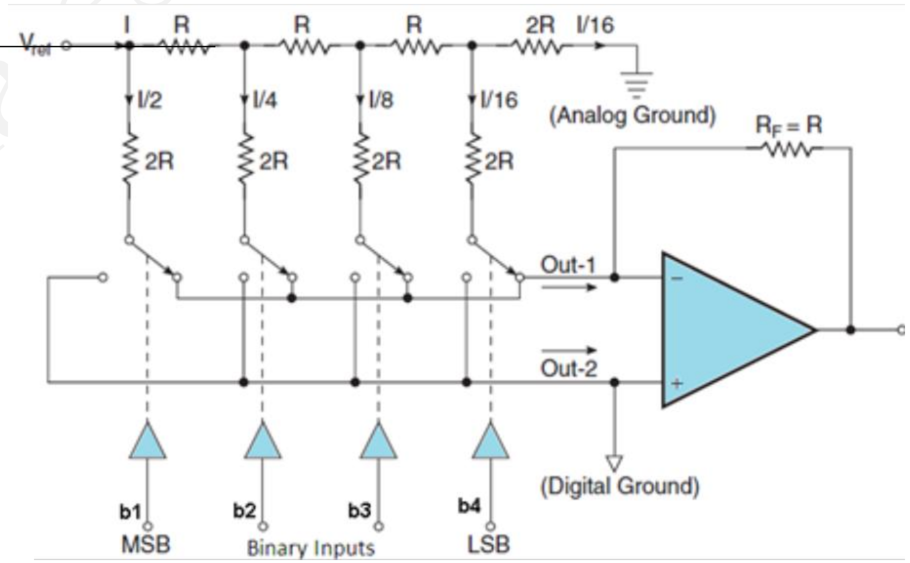
$$I_{out} = I_{ref}\left(\frac{D_7}{2} + \frac{D_6}{2^2} + \frac{D_5}{2^3} + \frac{D_4}{2^4} + \frac{D_3}{2^5} + \frac{D_2}{2^6} + \frac{D_1}{2^7} + \frac{D_0}{2^8}\right)$$

The I_{ref} is the input current. This must be provided into the pin 1.4. Generally 2.0mA is used as I_{ref}

DAC interfacing with 8051



R-2R ladder DAC



$$V_{out} = \frac{-R_f}{R} * V_{ref}\left(\frac{D_7}{2} + \frac{D_6}{2^2} + \frac{D_5}{2^3} + \frac{D_4}{2^4} + \frac{D_3}{2^5} + \frac{D_2}{2^6} + \frac{D_1}{2^7} + \frac{D_0}{2^8}\right)$$

Applications of 8051 Microcontroller

1. **Embedded Systems:** It is widely used as the core microcontroller in various embedded systems, including home appliances, industrial control systems, and consumer electronics.
2. **Automotive Electronics:** In automotive applications, the 8051 is used in engine control units (ECUs), dashboard displays, airbag systems, anti-lock braking systems (ABS), and other vehicle control systems.
3. **Home Automation:** It is employed in home automation systems for controlling lighting, heating, ventilation, air conditioning (HVAC), security systems, and other household devices.
4. **Industrial Automation:** In industrial settings, the 8051 microcontroller is used for process control, monitoring systems, and automation of manufacturing processes.
5. **Medical Devices:** The 8051 is utilized in medical devices such as infusion pumps, blood pressure monitors, glucose meters, and other healthcare equipment.
6. **Consumer Electronics:** Various consumer electronics devices, including TV remote controls, DVD players, washing machines, microwave ovens, and electronic toys, use the 8051 microcontroller.
7. **Communication Systems:** It is employed in communication systems, such as modems, routers, and network switches, for controlling and managing data communication.
8. **Security Systems:** Security and surveillance systems use the 8051 microcontroller for tasks such as controlling access systems, monitoring sensors, and managing security cameras.
9. **Instrumentation and Measurement:** It is utilized in instrumentation and measurement equipment for tasks like data acquisition, signal processing, and control in laboratories and industrial environments.
10. **Robotics:** It is used for controlling the movement, sensors, and other functions of robots in various applications, including industrial automation and educational robotics.
11. **Educational Platforms:** It is often used as an educational tool for teaching embedded systems and microcontroller programming due to its simplicity and widespread use.
12. **Smart Cards:** It is used in smart card applications, where it manages communication between the card and the card reader for tasks such as authentication and secure transactions.