



AKSHAYA INSTITUTE OF TECHNOLOGY, TUMKUR
Department of Electronics & Communication Engineering



Module 1 Notes for
“Introduction”
[BEC405A]

Prepared by:

Dr. Vijaya Kumar H R

Associate Professor

Department of ECE.

Akshaya Institute of Technology

Tumakuru

AKSHAYA INSTITUTE OF TECHNOLOGY

Lingapura, Obalapura Post, Koratagere Road, Tumakuru - 572106

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING



VISION

To produce competent engineering professionals in the field of Electronics and Communication Engineering by imparting value based quality technical education to meet the societal needs and to develop socially responsible citizens.



MISSION

M1: To provide strong fundamentals and technical skills in the field of Electronics and Communication Engineering through effective teaching learning process.

M2: Enhancing employability of the students by providing skills in the fields of VLSI, Embedded systems, Signal processing, etc., through Centre of Excellence.

M3: Encourage the students to participate in co-curricular and extra-curricular activities that creates a spirit of social responsibility and leadership qualities.



Program Specific Outcomes (PSOs)

After Successful Completion of Electronics and Communication Engineering Program Students will be able to

1. Apply fundamental knowledge of core. Electronics and Communication Engineering in the analysis, design and development of Electronics Systems as well as to interpret and synthesize experimental data leading to valid conclusions.
2. Exhibit the skills gathered to analyze, design, develop software applications and hardware products in the field of embedded systems and allied areas.



Program Educational Objectives (PEOs)

PEO1: Graduates exhibit their innovative ideas and management skills to meet the day to day technical challenges.

PEO2: Graduates utilize their knowledge and skills for the development of optimal solutions to the problems in the field of Electronics and Communication Engineering..

PEO3: Graduates exhibit good interpersonal skills, leadership qualities and adapt themselves for life-long Learning



MICROCONTROLLERS		Semester	4
Course Code	BEC405A	CIE Marks	50
Teaching Hours/Week(L:T:P)	3:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	3
Examination type(SEE)	Theory		

Course objectives:

This course will enable students to:

- Understand the difference between Microprocessor and Microcontroller and embedded microcontrollers.
- Analyze the basic architecture of 8051 microcontroller.
- Program 8051 microcontroller using Assembly Language and C.
- Understand the operation and use of inbuilt Timers/Counters and Serial port of 8051
- Understand the interrupt structure of 8051 and Interfacing I/O devices using I/O ports of 8051.

Teaching-Learning Process(General Instructions)

The samples strategies, which the teacher can use to accelerate the attainment of the various course outcomes are listed in the following:

1. Lecture method (L) does not mean only the traditional lecture method, but a different type of teaching method may be adopted to develop the outcomes.
 2. Show Video/animation films to explain the functioning of various techniques.
 3. Encourage collaborative(Group) Learning in the class
 4. Ask at least three HOTS(Higher-order Thinking) questions in the class, which promotes critical thinking
 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
 6. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.
 7. Discuss how every concept can be applied to the real world and when that's possible, it helps improve the students' understanding.
- Give Programming Assignments.

		LEVEL
Module-1 (8 Hrs)		
Microcontroller: Microprocessor Vs Microcontroller, Micro controller & Embedded Processors, Processor Architectures-Harvard Vs Princeton & RISC Vs CISC , 8051 Architecture- Registers, Pin diagram, I/O ports functions, Internal Memory organization. External Memory (ROM & RAM) interfacing. (Text book 1-1.1,Text book 2-1.0,1.1,3.0,3.1,3.2,3.3 Text book 3-Pg 5-9)		L1,L2
Module-2 (8 Hrs)		
Instruction Set: 8051 Addressing Modes, Data Transfer Instructions, Arithmetic instructions, Logical Instructions, Jump & Call Instructions Stack & Subroutine Instructions of 8051 (with examples in assembly Language). (Text book 2- Chapter 5,6,7,8, Additional reading Refer Textbook 3. Chapter 3 for complete understanding of instructions with		L1,L2

Module-3 (8 Hrs)	
Timers/Counters & Serial port programming: Basics of Timers & Counters, Data types & Time delay in the 8051 using C, Programming 8051 Timers, Mode 1 & Mode 2 Programming, Counter Programming (Assembly Language only). (Text book 2- 3.4, Text book 1- 7.1, 9.1,9.2) Basics of Serial Communication, 8051 Connection to RS232, Programming the 8051 to transfer data serially & to receive data serially using C.(Text book 2- 3.5, Text book 1- 10.1,10.2,10.3 except assembly language programs, 10.5)	L1,L2, L3
Module-4 (8 Hrs)	
Interrupt Programming: Basics of Interrupts, 8051 Interrupts, Programming Timer Interrupts, Programming Serial Communication Interrupts, Interrupt Priority in 8051(Assembly Language only) (Text book 2- 3.6, Text book 1- 11.1,11.2,11.4, 11.5)	L1,L2, L3
Module-5 (8 Hrs)	
I/O Port Interfacing & Programming: I/O Programming in 8051 C, LCD interfacing, DAC 0808 Interfacing, ADC 0804 interfacing, Stepper motor interfacing, DC motor control & Pulse Width Modulation (PWM) using C only. (Text book 1- 7.2, 12.1, 13.1, 13.2, 17.2, 17.3)	L1, L2, L3
Course outcome (Course Skill Set) At the end of the course, students will be able to: <ol style="list-style-type: none"> 1. Describe the difference between Microprocessor and Microcontroller, Types of Processor Architectures and Architecture of 8051Microcontroller. 2. Discuss the types of 8051 Microcontroller Addressing modes & Instructions with Assembly Language Programs. 3. Explain the programming operation of Timers/Counters and Serial port of 8051 Microcontroller. 4. Illustrate the Interrupt Structure of 8051 Microcontroller & its programming. 5. Develop C programs to interface I/O devices with 8051 Microcontroller. 	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). The student is declared as a pass in the course if he/she secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

- There are 25 marks for the CIE's Assignment component and 25 for the Internal Assessment Test component.
- Each test shall be conducted for 25 marks. The first test will be administered after 40-50% of the coverage of the syllabus, and the second test will be administered after 85-90% of the coverage of the syllabus. The average of the two tests shall be scaled down to 25 marks
- Any two assignment methods mentioned in the 22OB2.4, if an assignment is project-based then only one assignment for the courses shall be planned. The schedule for assignments shall be planned properly by the course teacher. The teacher should not conduct two assignments at the end of the semester if two assignments are planned. Each assignment shall be conducted for 25 marks. (If two assignments are conducted then the sum of the two assignments shall be scaled down to 25 marks)
- The final CIE marks of the course out of 50 will be the sum of the scale-down marks of tests and assignment/s marks.

Internal Assessment Test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester-End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the course (duration 03 hours).

1. The question paper will have ten questions. Each question is set for 20 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.
3. The students have to answer 5 full questions, selecting one full question from each module.
4. Marks scored shall be proportionally reduced to 50 marks

Suggested Learning Resources:

TEXT BOOKS

1. The “8051 Microcontroller and Embedded Systems – Using Assembly and C”, Muhammad Ali Mazidi and Janice Gillespie Mazidi and Rollind. Mckinlay; Phi, 2006 / Pearson, 2006.
2. “The 8051 Microcontroller”, Kenneth j. Ayala, 3rd edition, Thomson/Cengage Learning.
3. “Programming And Customizing The 8051 Microcontroller”.,Myke Predko Tata Mc Graw-Hill Edition 1999 (reprint 2003).

REFERENCEBOOKS:

1. “The 8051 Microcontroller Based Embedded Systems”, Manish K Patel, McGraw Hill, 2014, ISBN: 978-93-329-0125-4.
2. “Microcontrollers: Architecture, Programming, Interfacing and System Design”, Raj Kamal, Pearson Education, 2005.

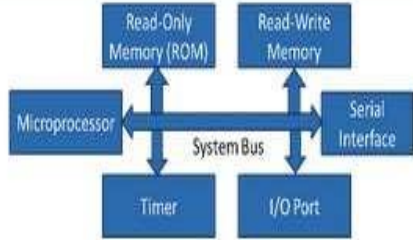
Web links and Video Lectures(e-Resources):

https://youtu.be/pA6K5NgWTow?si=zQqqgXQq50dVL_-s

8051 Microcontroller

Module 1: Introduction

Microprocessor



Micro Controller



Microprocessor is heart of Computer system.

Micro Controller is a heart of embedded system.

It is just a processor. Memory and I/O components have to be connected externally

Micro controller has external processor along with internal memory and i/O components

Since memory and I/O has to be connected externally, the circuit becomes large.

Since memory and I/O are present internally, the circuit is small.

Cannot be used in compact systems and hence inefficient

Can be used in compact systems and hence it is an efficient technique

Cost of the entire system increases

Cost of the entire system is low

Due to external components, the entire power consumption is high. Hence it is not suitable to be used with devices running on stored power like batteries.

Since external components are low, total power consumption is less and can be used with devices running on stored power like batteries.

Most of the microprocessors do not have power saving features.

Most of the micro controllers have power saving modes like idle mode and power saving mode. This helps to reduce power consumption even further.

Since memory and I/O components are all external, each instruction will need external operation, hence it is relatively slower.

Since components are internal, most of the operations are internal instruction, hence speed is fast.

Microprocessor have less number of registers, hence more operations are memory based.

Micro controller have more number of registers, hence the programs are easier to write.

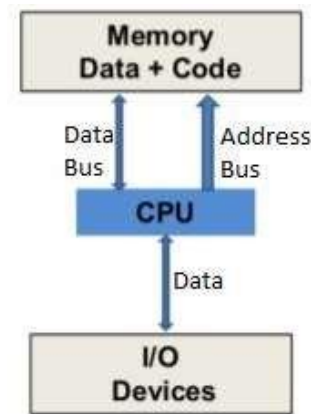
Microprocessors are based on von Neumann model/architecture where program and data are stored in same memory module

Micro controllers are based on Harvard architecture where program memory and Data memory are separate

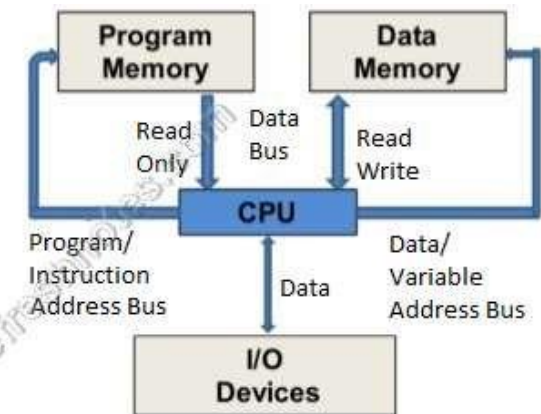
Mainly used in personal computers

Used mainly in washing machine, MP3 players

CISC	RISC
The original microprocessor ISA	Redesigned ISA that emerged in the early 1980s
Instructions can take several clock cycles	Single-cycle instructions
Hardware-centric design – the ISA does as much as possible using hardware circuitry	Software-centric design – High-level compilers take on most of the burden of coding many software steps from the programmer
More efficient use of RAM than RISC	Heavy use of RAM (can cause bottlenecks if RAM is limited)
Complex and variable length instructions	Simple, standardized instructions
May support microcode (micro-programming where instructions are treated like small programs)	Only one layer of instructions
Large number of instructions	Small number of fixed-length instructions
Compound addressing modes	Limited addressing modes



Von Neumann



Harvard

Von-Neuman	Harvard
First digital computer architecture. Introduced stored program concept	Computer architecture based on Harvard Mark 1
One memory module for data and instructions	Have different memory modules for data and instructions.
common bus for data and instructions	Individual buses for data and instructions
CPU takes 2 clocks to execute one instruction. Because fetch data before executing an instruction.	Can execute instruction one clock cycle.
CPU can not fetch instructions and data read/write at the same time.	CPU can not fetch instructions and data read/write at the same time.
Slow	Fast

Dr. Vijaya Kumar H R
Associate. Professor, Dept. of ECE, AIT, Tumakuru

Criteria for Choosing a Microcontroller

- Meeting the computing needs of the task at hand efficiently and cost effectively
- Speed
- Packaging
- Power consumption
- The amount of RAM and ROM on chip
- The number of I/O pins and the timer on chip
- How easy to upgrade to higher performance or lower power-consumption version
- Cost per unit

Various versions of Microcontroller

Feature	8051	8052	8031
ROM (on-chip program space in bytes)	4K	8K	0K
RAM (bytes)	128	256	128
Timers	2	3	2
I/O pins	32	32	32
Serial port	1	1	1
Interrupt sources	6	8	6

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

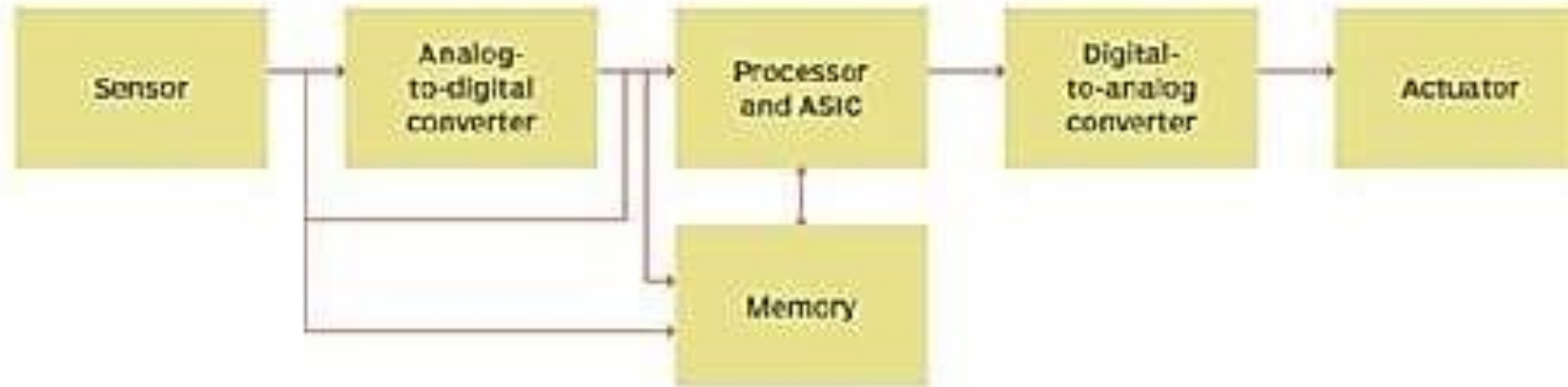
Applications of Microcontroller

1. Consumer electronics products: Toys, Camera, Robots, washing machine, Microwave ovens
2. Instrumentation and process control: Oscilloscopes, Multimeter, Leakage current tester, data acquisition and control.
3. Medical instruments: ECG, Blood measurements.
4. Communication: Cell phones, telephone sets, communication machines.
5. Office: Security system, FAX Machine, copier, printer, paging, intercom.
6. Auto: Engine control, air bag, security system, center locking system.
7. Others: Cellular phones, traffic controller, musical instruments, camera, E voting machine etc.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Embedded system structure diagram



- An embedded system is a combination of computer hardware and software designed for a specific function. Embedded systems may also function within a larger system.
- The systems can be programmable or have a fixed functionality. Industrial machines, consumer electronics, agricultural and processing industry devices, automobiles, medical equipment, cameras, digital watches, household appliances, airplanes, vending machines and toys, as well as mobile devices, are possible locations for an embedded system.
- While embedded systems are computing systems, they can range from having no user interface ([UI](#)) -- for example, on devices designed to perform a single task -- to complex graphical user interfaces ([GUIs](#)), such as in mobile devices. [User interfaces can include](#) buttons, LEDs (light-emitting diodes) and touchscreen sensing. Some systems use remote user interfaces as well.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Characteristics of embedded systems

- The main characteristic of embedded systems is that they are task-specific.
- Additionally, embedded systems can include the following characteristics: typically, consist of hardware, software and firmware;
- can be embedded in a larger system to perform a specific function, as they are built for specialized tasks within the system, not various tasks;
- can be either microprocessor-based or microcontroller-based -- both are integrated circuits that give the system compute power;
- are often used for sensing and real-time computing in internet of things ([IoT](#)) devices, which are devices that are internet-connected and do not require a user to operate;
- can vary in complexity and in function, which affects the type of software, firmware and hardware they use; and
- are often required to perform their function under a time constraint to keep the larger system functioning properly.

Structure of embedded systems

Embedded systems vary in complexity but, generally, consist of three main elements:

- **Hardware.** The hardware of embedded systems is based around microprocessors and microcontrollers. Microprocessors are very similar to microcontrollers and, typically, refer to a CPU (central processing unit) that is integrated with other basic computing components such as memory chips and digital signal processors ([DSPs](#)). Microcontrollers have those components built into one chip.
- **Software and firmware.** Software for embedded systems can vary in complexity. However, industrial-grade microcontrollers and embedded IoT systems usually run very simple software that requires little memory.
- **Real-time operating system.** These are not always included in embedded systems, especially smaller-scale systems. RTOS define how the system works by supervising the software and setting rules during program execution.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Basic embedded system would consist of the following elements:

- **Sensors** convert physical sense data into an electrical signal.
- **Analog-to-digital (A-D) converters** change an analog electrical signal into a digital one.
- **Processors** process digital signals and store them in memory.
- **Digital-to-analog (D-A) converters** change the digital data from the processor into analog data.
- **Actuators** compare actual output to memory-stored output and choose the correct one.

The sensor reads external inputs, the converters make that input readable to the processor, and the processor turns that information into useful output for the embedded system.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Types of embedded systems

There are a few basic embedded system types, which differ in their functional requirements. They are:

- **Mobile embedded systems** are small-sized systems that are designed to be portable. Digital cameras are an example of this.
- **Networked embedded systems** are connected to a network to provide output to other systems. Examples include home security systems and point of sale (POS) systems.
- **Standalone embedded systems** are not reliant on a host system. Like any embedded system, they perform a specialized task. However, they do not necessarily belong to a host system, unlike other embedded systems. A calculator or MP3 player is an example of this.
- **Real-time embedded systems** give the required output in a defined time interval. They are often used in medical, industrial and military sectors because they are responsible for time-critical tasks. A traffic control system is an example of this.

Embedded Microcontroller

- ❑ **An embedded product uses a microprocessor (or microcontroller) to do one task and one task only**
 - There is only one application software that is typically burned into ROM
- ❑ **A PC, in contrast with the embedded system, can be used for any number of applications**
 - It has RAM memory and an operating system that loads a variety of applications into RAM and lets the CPU run them
 - A PC contains or is connected to various embedded products
 - Each one peripheral has a microcontroller inside it that performs only one task

❑ **Home**

- Appliances, intercom, telephones, security systems, garage door openers, answering machines, fax machines, home computers, TVs, cable TV tuner, VCR, camcorder, remote controls, video games, cellular phones, musical instruments, sewing machines, lighting control, paging, camera, pinball machines, toys, exercise equipment

❑ **Office**

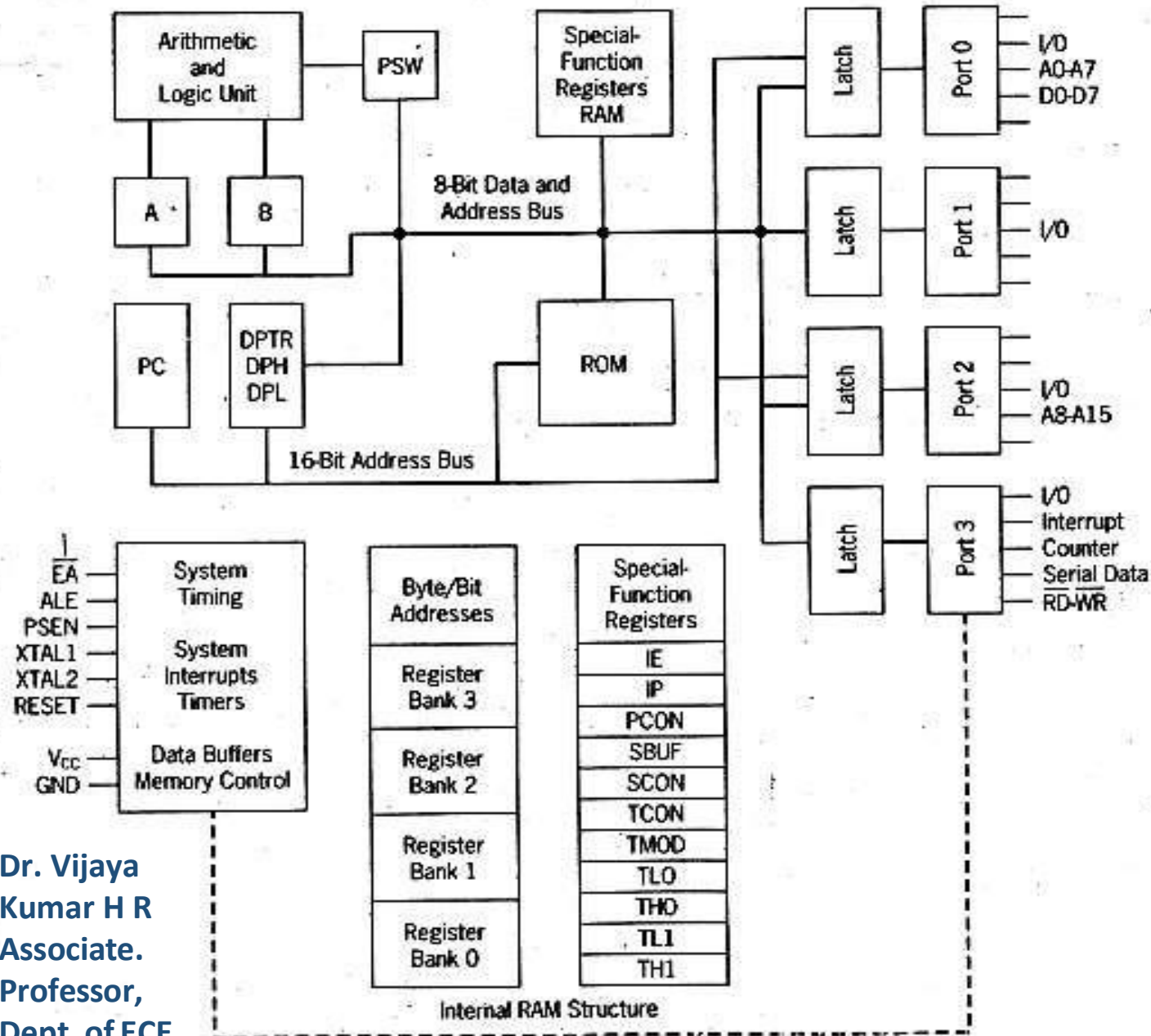
- Telephones, computers, security systems, fax machines, microwave, copier, laser printer, color printer, paging

❑ **Auto**

- Trip computer, engine control, air bag, ABS, instrumentation, security system, transmission control, entertainment, climate control, cellular phone, keyless entry

- ❑ **Many manufactures of general-purpose microprocessors have targeted their microprocessor for the high end of the embedded market**
 - There are times that a microcontroller is inadequate for the task
- ❑ **When a company targets a general-purpose microprocessor for the embedded market, it optimizes the processor used for embedded systems**
- ❑ **Very often the terms *embedded processor* and *microcontroller* are used interchangeably**
- ❑ **One of the most critical needs of an embedded system is to decrease power consumption and space**
- ❑ **In high-performance embedded processors, the trend is to integrate more functions on the CPU chip and let designer decide which features he/she wants to use**
- ❑ **In many cases using x86 PCs for the high-end embedded applications**
 - Saves money and shortens development time
 - A vast library of software already written
 - Windows is a widely used and well understood platform

8051 Block Diagram



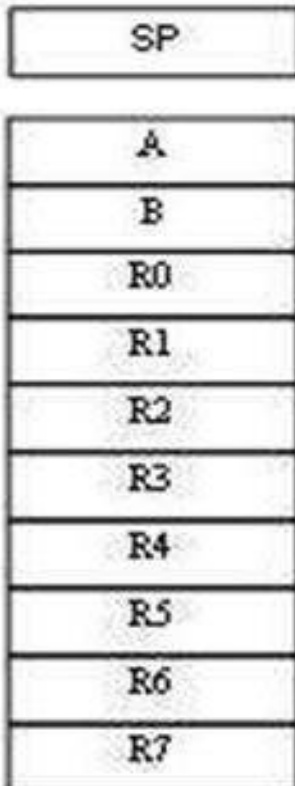
Features of 8051 Microcontroller:

- 4KB bytes on-chip program memory (ROM)
- 128 bytes on-chip data memory (RAM)
- Four register banks
- 128 user defined software flags
- 8-bit bidirectional data bus
- 16-bit unidirectional address bus
- 32 general purpose registers each of 8-bit
- 16 bit Timers (usually 2, but may have more or less)
- Three internal and two external Interrupts
- Four 8-bit ports, (short model have two 8-bit ports)
- 16-bit program counter and data pointer
- 8051 may also have a number of special features such as UARTs, ADC, Op-amp, etc.

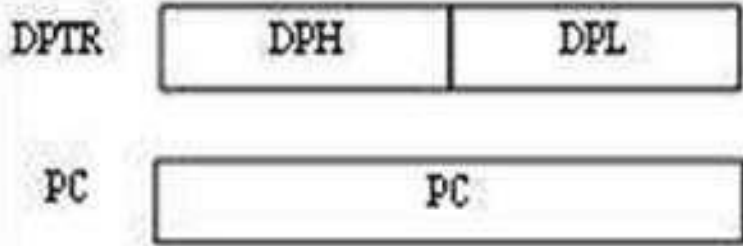
CPU of 8051: Registers are used in the CPU to store information on temporarily basis which could be data to be processed, or an address pointing to the data which is to be fetched. In 8051, there is one data type is of 8-bits, from the MSB (most significant bit) D7 to the LSB (least significant bit) D0. With 8-bit data type, any data type larger than 8-bits must be broken into 8-bit chunks before it is processed. The most widely used registers of the 8051 are A (accumulator), B, R0-R7, DPTR (data pointer), and PC (program counter). All these registers are of 8-bits, except DPTR and PC.

Accumulator: A register or Acc register: The accumulator, register A, is a 8 bit register and it is used for all arithmetic and logic operations. If the accumulator is not present, then every result of each calculation (addition, multiplication, shift, etc.) is to be stored into the main memory. Access to main memory is slower than access to a register like the accumulator because the technology used for the large main memory is slower (but cheaper) than that used for a register.

The "B" Register: The "B" register is very similar to the Accumulator in the sense that it may hold an 8-bit (1-byte) value. The "B" register is used only by two 8051 instructions: **MUL AB** and **DIV AB**. To quickly and easily multiply or divide A by another number, you may store the other number in "B" and make use of these two instructions. Apart from using MUL and DIV instructions, the "B" register is often used as yet another temporary storage register, much like a ninth R register.



8-bit Registers of 8051



16-bit Registers of 8051

Dr. Vijaya Kumar H R
Associate. Professor, Dept. of ECE, AIT, Tumakuru

The Data Pointer: The Data Pointer (DPTR) is the 8051's only user-accessible 16-bit (2-byte) register. The Accumulator, R0–R7 registers and B register are 1-byte value registers. DPTR is meant for pointing to data. It is used by the 8051 to access external memory using the address indicated by DPTR. DPTR is the only 16-bit register available and is often used to store 2-byte values.

The Program Counter: The Program Counter (PC) is a 2-byte address which tells the 8051 where the next instruction to execute can be found in the memory. PC starts at 0000h when the 8051 initializes and is incremented every time after an instruction is executed. PC is not always incremented by 1. Some instructions may require 2 or 3 bytes; in such cases, the PC will be incremented by 2 or 3.

The Stack Pointer (SP): The Stack Pointer, like all registers except DPTR and PC, may hold an 8-bit (1-byte) value. The Stack Pointer tells the location from where the next value is to be removed from the stack. When a value is pushed onto the stack, the value of SP is incremented and then the value is stored at the resulting memory location. When a value is popped off the stack, the value is returned from the memory location indicated by SP, and then the value of SP is decremented.

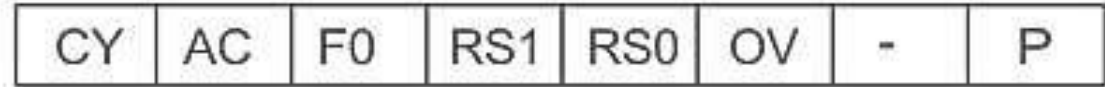
This order of operation is important. SP will be initialized to 07h when the 8051 is initialized. If a value is pushed onto the stack at the same time, the value will be stored in the internal RAM address 08h because the 8051 will first increment the value of SP (from 07h to 08h) and then will store the pushed value at that memory address (08h). SP is modified directly by the 8051 by six instructions: PUSH, POP, ACALL, LCALL, RET, and RETI.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

8051 Flag Bits and PSW Register

The program status word (PSW) register is an 8-bit register, also known as **flag register**. It is of 8-bit wide but only 6-bit of it is used. The two unused bits are **user-defined flags**. Four of the flags are called **conditional flags**, which means that they indicate a condition which results after an instruction is executed. These four are **CY** (Carry), **AC** (auxiliary carry), **P** (parity), and **OV** (overflow). The bits RS0 and RS1 are used to change the bank registers. The following figure shows the program status word register.



•**CY, the carry flag** – This carry flag is set (1) whenever there is a carry out from the D7 bit. It is affected after an 8-bit addition or subtraction operation. It can also be reset to 1 or 0 directly by an instruction such as "SETB C" and "CLR C" where "SETB" stands for set bit carry and "CLR" stands for clear carry.

•**AC, auxiliary carry flag** – If there is a carry from D3 and D4 during an ADD or SUB operation, the AC bit is set; otherwise, it is cleared. It is used for the instruction to perform binary coded decimal arithmetic.

•**P, the parity flag** – The parity flag represents the number of 1's in the accumulator register only. If the A register contains odd number of 1's, then P = 1; and for even number of 1's, P = 0.

•**OV, the overflow flag** – This flag is set whenever the result of a signed number operation is too large causing the high-order bit to overflow into the sign bit. It is used only to detect errors in signed arithmetic operations.

CY	PSW.7	Carry flag.
AC	PSW.6	Auxiliary carry flag.
F0	PSW.5	Available to the user for general purpose
RS1	PSW.4	Register Bank selector bit 1.
RS0	PSW.3	Register Bank selector bit 0.
OV	PSW.2	Overflow flag.
-	PSW.1	User-definable bit
P	PSW.0	Parity flag, set/cleared by hardware each instruction cycle to indicate an odd/even number of 1 bits in the accumulator.

RS1	RS0	Register Bank	Address
0	0	0	00H - 07H
0	1	1	08H - 0FH
1	0	2	10H - 17H
1	1	3	18H - 1FH

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Instructions that affect flag bits

Instruction	CY	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	0	X	
DIV	0	X	
DA	X		
RPC	X		
PLC	X		
SETB C	1		
CLR C	0		
CPL C	X		
ANL C, bit	X		
ANL C, /bit	X		
ORL C, bit	X		
ORL C, /bit	X		
MOV C, bit	X		
CJNE	X		

Dr. Vijaya Kumar H R
Associate. Professor, Dept. of ECE, AIT,
Tumakuru

Example 2-2
Show the status of the CY, AC, and P flags after the addition of 38H and 2FH in the following instructions.

```
MOV A, #38H
ADD A, #2FH      ;after the addition A=67H, CY=0
```

Solution:

```

      38      00111000
+     2F      00101111
-----
      67      01100111

```

CY = 0 since there is no carry beyond the D7 bit.

AC = 1 since there is a carry from the D3 to the D4 bit.

P = 1 since the accumulator has an odd number of 1s (it has five 1s).

Example 2-3

Show the status of the CY, AC, and P flags after the addition of 9CH and 64H in the following instructions.

```
MOV A, #9CH
ADD A, #64H      ;after addition A=00 and CY=1
```

Solution:

```

      9C      10011100
+     64      01100100
-----
     100     00000000

```

CY = 1 since there is a carry beyond the D7 bit.

AC = 1 since there is a carry from the D3 to the D4 bit.

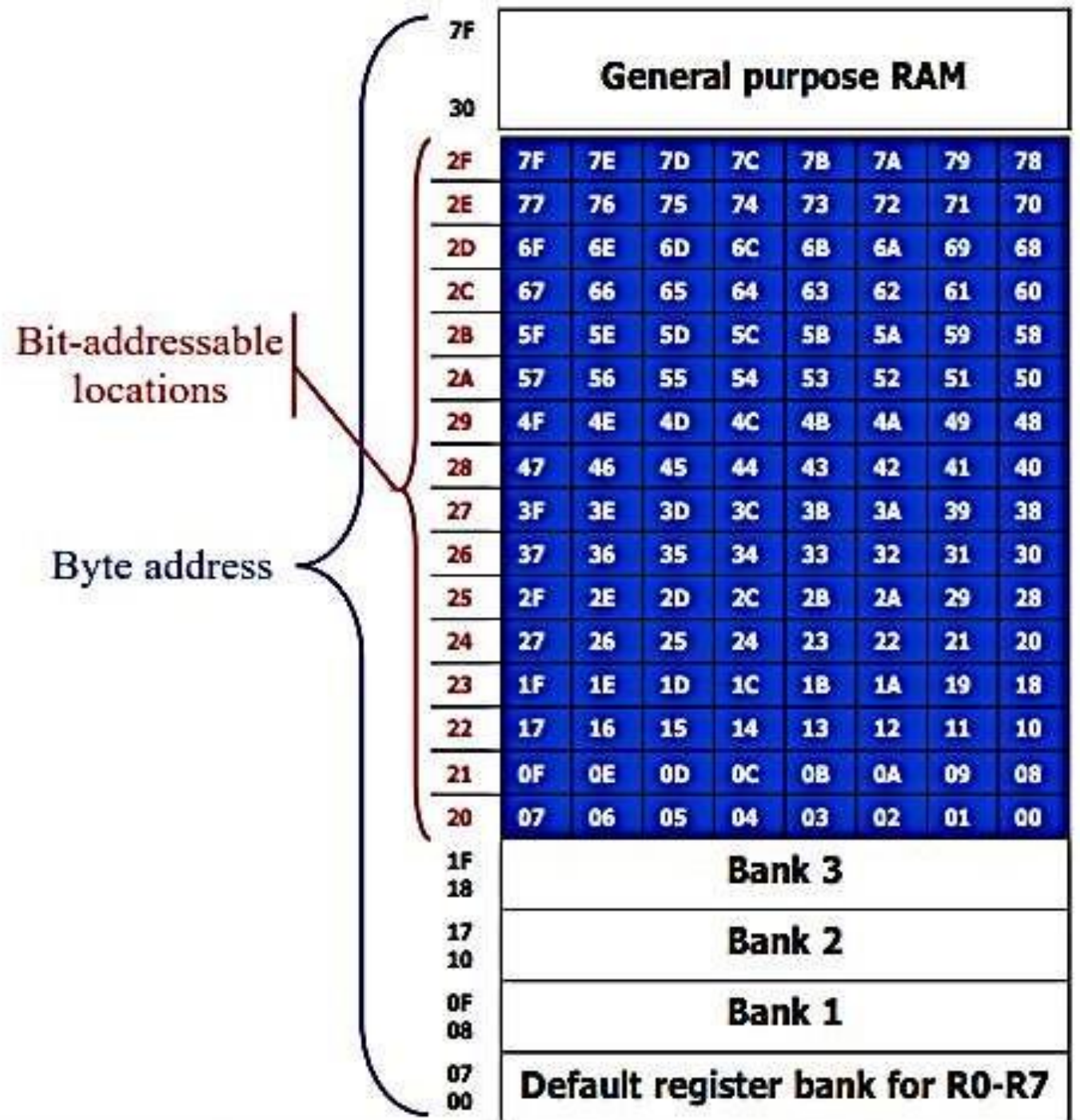
P = 0 since the accumulator has an even number of 1s (it has zero 1s).

Data Memory (RAM) or Internal RAM of 8051 Microcontroller: The Data Memory or RAM of the 8051 Microcontroller stores temporary data and intermediate results that are generated and used during the normal operation of the microcontroller. Original Intel's 8051 Microcontroller had 128B of internal RAM.

Three parts:

1. 4 Register bank.
2. Bit-addressable location or Special function registers
3. General purpose or Scratch pad registers.

Dr. Vijaya Kumar H R
Associate. Professor, Dept. of ECE, AIT, Tumakuru



- The 4 banks are named as Bank0, Bank1, Bank2 and Bank3. Each Bank consists of 8 registers named as R0 – R7. Each Register can be addressed in two ways: either by name or by address. To address the register by name, first the corresponding Bank must be selected. In order to select the bank, we have to use the RS0 and RS1 bits of the Program Status Word (PSW) Register (RS0 and RS1 are 3rd and 4th bits in the PSW Register).
- When addressing the Register using its address i.e., 12H for example, the corresponding Bank may or may not be selected. (12H corresponds to R2 in Bank2). The next 16B of the RAM i.e., from 20H to 2FH are Bit – Addressable memory locations.
- There are totally 128 bits that can be addressed individually using 00H to 7FH or the entire byte can be addressed as 20H to 2FH. For example 32H is the bit 2 of the internal RAM location 26H.
- The final 80B of the internal RAM i.e., addresses from 30H to 7FH, is the general purpose RAM area which are byte addressable. These lower 128B of RAM can be addressed directly or indirectly.
- The upper 128B of the RAM i.e., memory addresses from 80H to FFH is allocated for Special Function Registers (SFRs). SFRs control specific functions of the 8051 Microcontroller. Some of the SFRs are I/O Port Registers (P0, P1, P2 and P3), PSW (Program Status Word), A (Accumulator), IE (Interrupt Enable), PCON (Power Control), etc.
- SFRs Memory addresses are only direct addressable. Even though some of the addresses between 80H and FFH are not assigned to any SFR, they cannot be used as additional RAM area. In some microcontrollers, there is an additional 128B of RAM, which share the memory address with SFRs i.e., 80H to FFH. But, this additional RAM block is only accessed by indirect addressing.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

<i>Name of the Register</i>	<i>Function</i>	<i>Internal RAM Address (HEX)</i>
ACC	Accumulator	E0H
B	B Register (for Arithmetic)	F0H
DPH	Addressing External Memory	83H
DPL	Addressing External Memory	82H
IE	Interrupt Enable Control	A8H
IP	Interrupt Priority	B8H
P0	PORT 0 Latch	80H
P1	PORT 1 Latch	90H
P2	PORT 2 Latch	A0H
P3	PORT 3 Latch	B0H
PCON	Power Control	87H
PSW	Program Status Word	D0H
SCON	Serial Port Control	98H
SBUF	Serial Port Data Buffer	99H
SP	Stack Pointer	81H
TMOD	Timer / Counter Mode Control	89H
TCON	Timer / Counter Control	88H
TL0	Timer 0 LOW Byte	8AH
TH0	Timer 0 HIGH Byte	8CH
TL1	Timer 1 LOW Byte	8BH
TH1	Timer 1 HIGH Byte	8DH

Dr. Vijaya Kumar H R
Associate. Professor,
Dept. of ECE, AIT,
Tumakuru

Program Memory (ROM) of 8051 Microcontroller

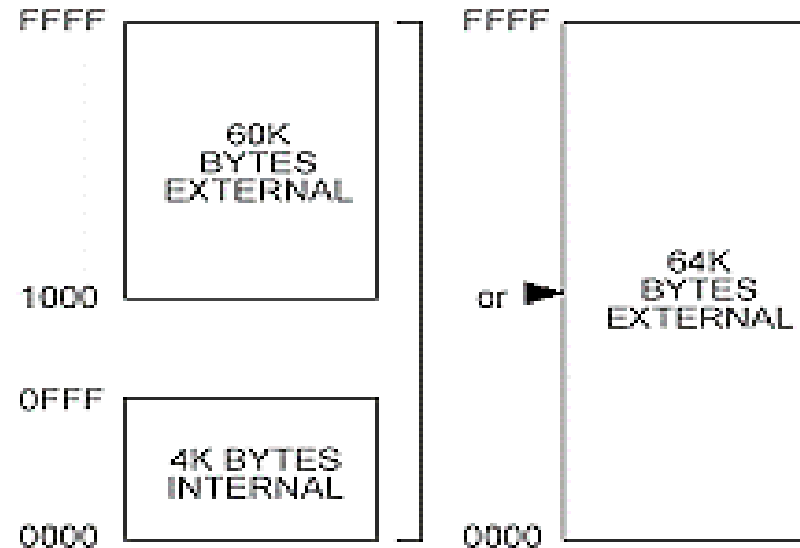
In 8051 Microcontroller, the code or instructions to be executed are stored in the Program Memory, which is also called as the ROM of the Microcontroller. The original 8051 Microcontroller by Intel has 4KB of internal ROM.

Some variants of 8051 like the 8031 and 8032 series doesn't have any internal ROM (Program Memory) and must be interfaced with external Program Memory with instructions loaded in it.

Almost all modern 8051 Microcontrollers, like 8052 Series, have 8KB of Internal Program Memory (ROM) in the form of Flash Memory (ROM) and provide the option of reprogramming the memory.

In case of 4KB of Internal ROM, the address space is 0000H to 0FFFH. If the address space i.e., the program addresses exceed this value, then the CPU will automatically fetch the code from the external Program Memory.

The 80C51 Program Memory.



Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

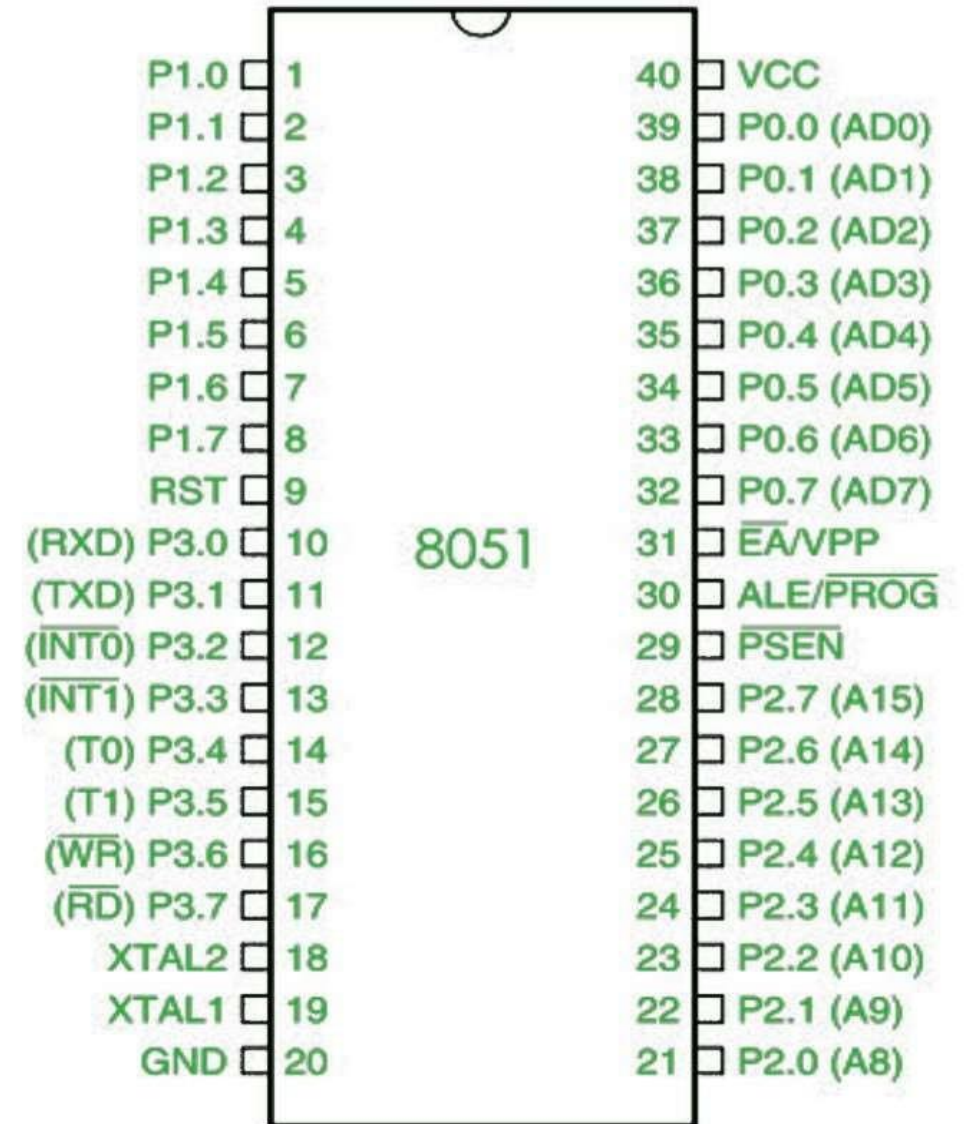
Description of the Pins :

•Pin 1 to Pin 8 (Port 1) –

Pin 1 to Pin 8 are assigned to Port 1 for simple I/O operations. They can be configured as input or output pins depending on the logic control i.e. if logic zero (0) is applied to the I/O port it will act as an output pin and if logic one (1) is applied the pin will act as an input pin. These pins are also referred to as P1.0 to P1.7 (where P1 indicates that it is a pin in port 1 and the number after ‘.’ tells the pin number i.e. 0 indicates first pin of the port. So, P1.0 means first pin of port 1, P1.1 means second pin of the port 1 and so on). These pins are bidirectional pins.

•Pin 9 (RST) –

Reset pin. It is an active-high, input pin. Therefore if the RST pin is high for a minimum of 2 machine cycles, the microcontroller will reset i.e. it will close and terminate all activities. It is often referred as “power-on-reset” pin because it is used to reset the microcontroller to its initial values when power is on (high).



40 - PIN DIP

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

•Pin 10 to Pin 17 (Port 3) –

Pin 10 to pin 17 are port 3 pins which are also referred to as P3.0 to P3.7. These pins are similar to port 1 and can be used as universal input or output pins. These pins are bidirectional pins. These pins also have some additional functions which are as follows:

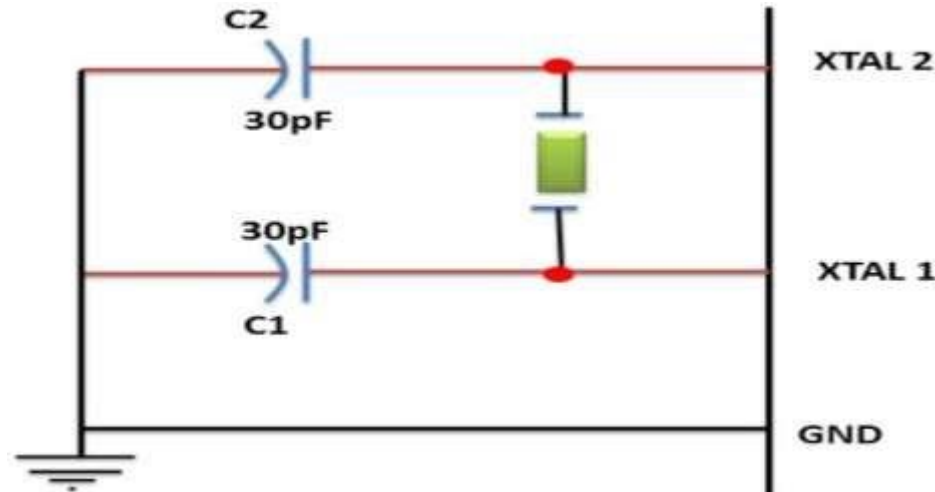
- **P3.0 (RXD) :**
10th pin is RXD (serial data receive pin) which is for serial input. Through this input signal microcontroller receives data for serial communication.
- **P3.1 (TXD) :**
11th pin is TXD (serial data transmit pin) which is serial output pin. Through this output signal microcontroller transmits data for serial communication.
- **P3.2 and P3.3 (INT0', INT1') :**
12th and 13th pins are for External Hardware Interrupt 0 and Interrupt 1 respectively. When this interrupt is activated(i.e. when it is low), 8051 gets interrupted in whatever it is doing and jumps to the vector value of the interrupt (0003H for INT0 and 0013H for INT1) and starts performing Interrupt Service Routine (ISR) from that vector location.
- **P3.4 and P3.5 (T0 and T1) :**
14th and 15th pin are for Timer 0 and Timer 1 external input. They can be connected with 16 bit timer/counter.
- **P3.6 (WR') :**
16th pin is for external memory write i.e. writing data to the external memory.
- **P3.7 (RD') :**
17th pin is for external memory read i.e. reading data from external memory.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

- **Pin 18 and Pin 19 (XTAL2 And XTAL1) –**

These pins are connected to an external oscillator which is generally a quartz crystal oscillator. They are used to provide an external clock frequency of 4MHz to 30MHz.



- **Pin 20 (GND) –**

This pin is connected to the ground. It has to be provided with 0V power supply. Hence it is connected to the negative terminal of the power supply.

- **Pin 21 to Pin 28 (Port 2) –**

Pin 21 to pin 28 are port 2 pins also referred to as P2.0 to P2.7. When additional external memory is interfaced with the 8051 microcontroller, pins of port 2 act as higher-order address bytes. These pins are bidirectional.

- **Pin 29 (PSEN) –**

PSEN stands for Program Store Enable. It is output, active-low pin. This is used to read external memory. In 8031 based system where external ROM holds the program code, this pin is connected to the OE pin of the ROM.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

•Pin 30 (ALE/ PROG) –

ALE stands for Address Latch Enable. It is input, active-high pin. This pin is used to distinguish between memory chips when multiple memory chips are used. It is also used to de-multiplex the multiplexed address and data signals available at port 0. During flash programming i.e. Programming of EPROM, this pin acts as program pulse input (PROG).

•Pin 31 (EA/ VPP) –

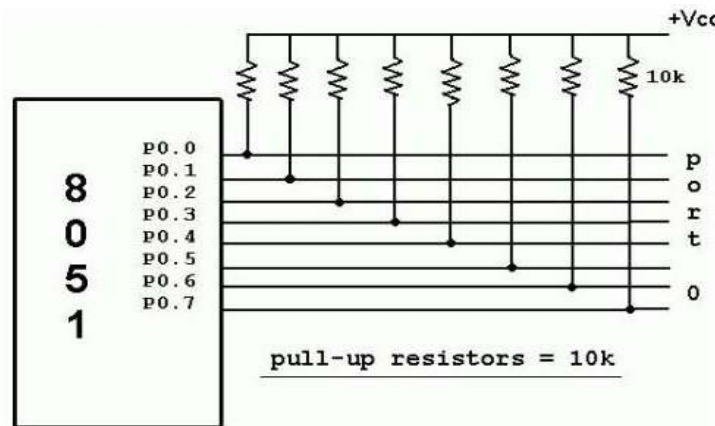
EA is active low pin and stands for External Access input. It is used to enable/disable external memory interfacing. In 8051, EA is connected to Vcc as it comes with on-chip ROM to store programs. For other family members such as 8031 and 8032 in which there is no on-chip ROM, the EA pin is connected to the GND

•Pin 32 to Pin 39 (Port 0) –

Pin 32 to pin 39 are port 0 pins also referred to as P0.0 to P0.7. They are bidirectional input/output pins. They don't have any internal pull-ups. Hence, 10 K Ω pull-up resistors are used as external pull-ups. Port 0 is also designated as AD0-AD7 because 8051 multiplexes address and data through port 0 to save pins.

•Pin 40 (VCC) –

This pin provides power supply voltage i.e. +5 Volts to the circuit.

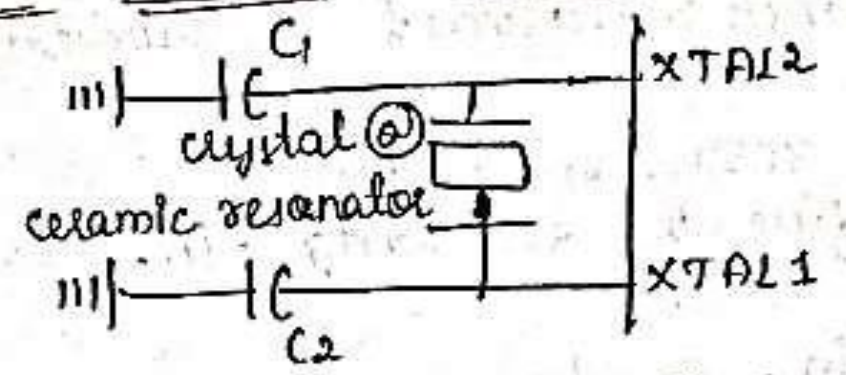


8051 doesn't have internal pull-up resistor. Generally we use external pull up resistor on Port0 of 8051 microcontroller. We use resistor of 10k connected to pins and VCC. Or we use 10k resistor array to do that, the reason is because 8051 microcontroller doesn't have internal pull up resistor on port0

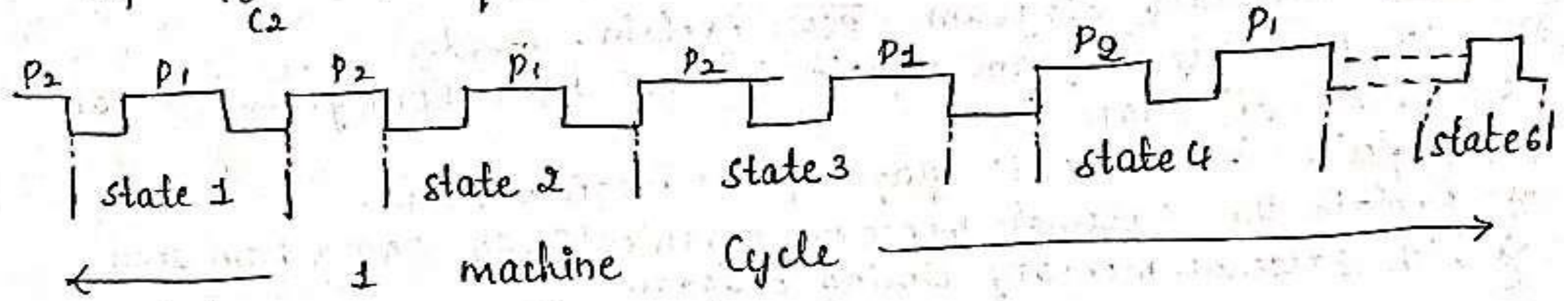
Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

8051 Oscillator & CLK:-



$$T = \frac{CX12d}{\text{crystal freq.}}$$



* Crystal freq = 16 MHz.

$$* T = \frac{1 \times 12}{16 \times M} = 0.75 \mu s$$

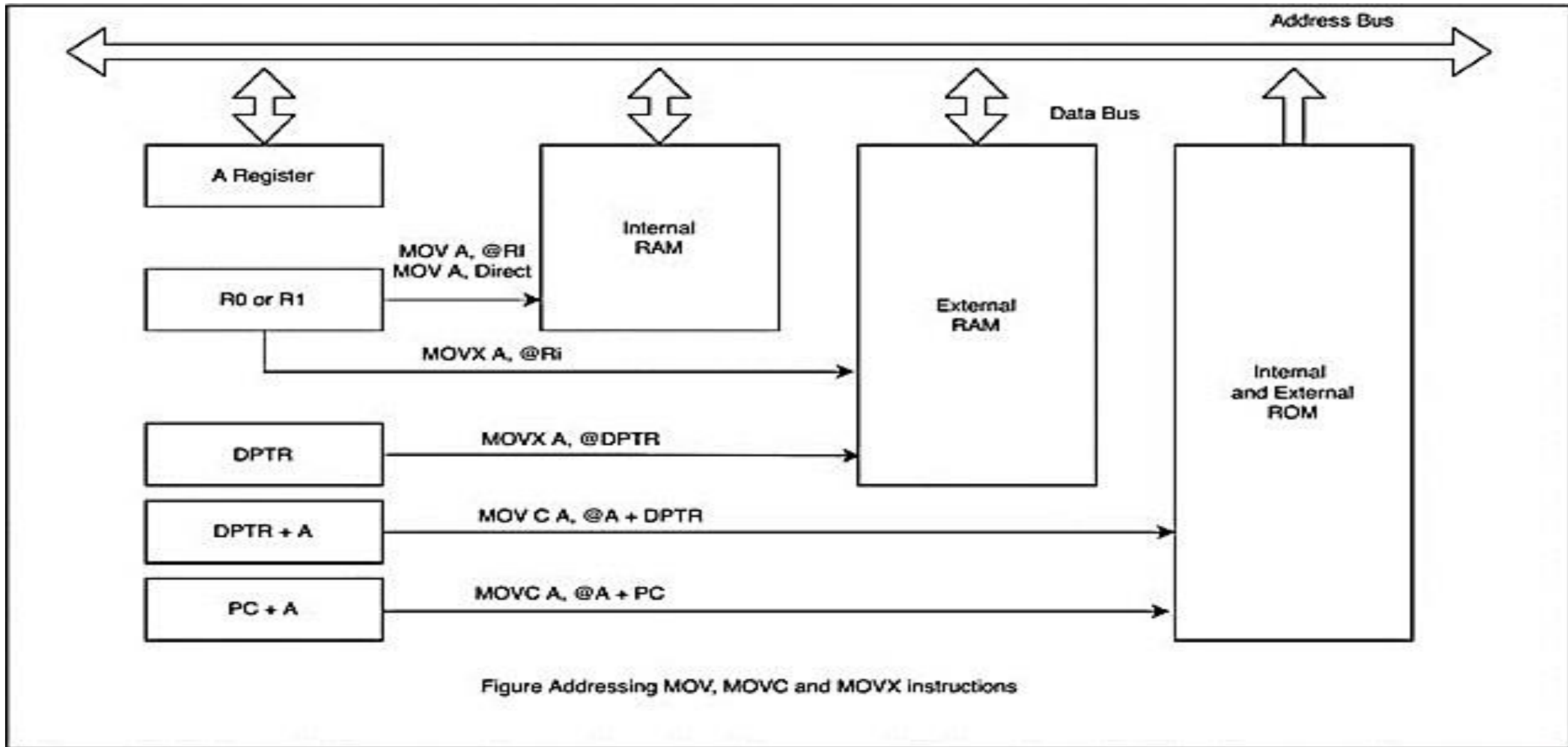
$$* T = \frac{1 \times 12}{11.0592 \times 10^6} = 1.085 \times 10^{-6} \text{ sec.}$$

Interfacing External Memory with 8051 Microcontroller

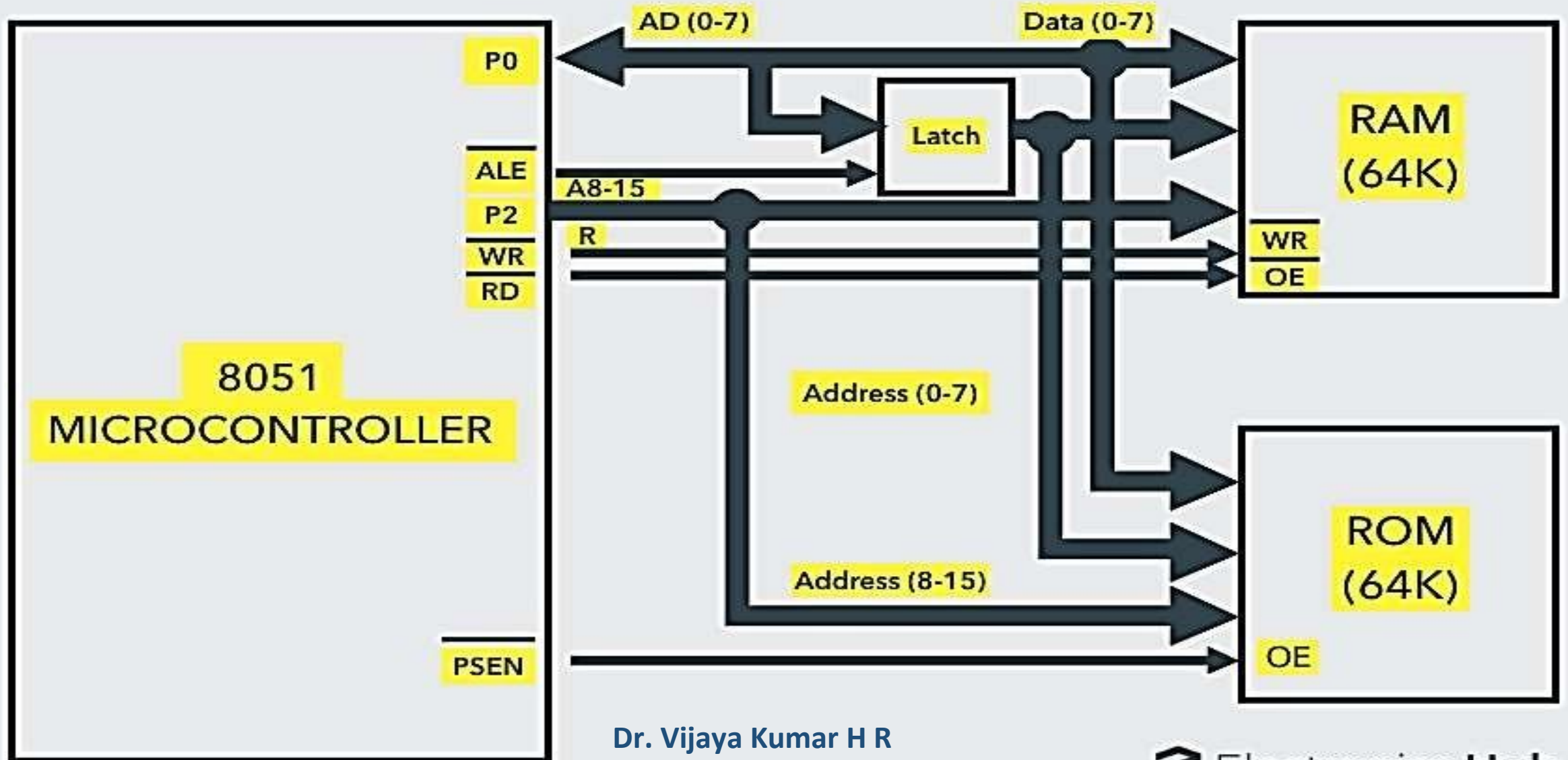
- It is always good to have an option to expand the capabilities of a Microcontroller, whether it is in terms of Memory or IO or anything else. Such expansion will be useful to avoid design throttling. We have seen that a typical 8051 Microcontroller has 4KB of ROM and 128B of RAM (most modern 8051 Microcontroller variants have 8K ROM and 256B of RAM).
- The designer of an 8051 Microcontroller based system is not limited to the internal RAM and ROM present in the 8051 Microcontroller. There is a provision of connecting both external RAM and ROM i.e., Data Memory and Program.
- The reason for interfacing external Program Memory or ROM is that complex programs written in high – level languages often tend to be larger and occupy more memory.
- Another important reason is that chips like 8031 or 8032, which doesn't have any internal ROM, have to be interfaced with external ROM.
- A maximum of 64KB of Program Memory (ROM) and Data Memory (RAM) each can be interface with the 8051 Microcontroller.
- An important point to remember when interfacing external memory with 8051 Microcontroller is that Port 0 (P0) cannot be used as an IO Port as it will be used for multiplexed address and data bus (A0 – A7 and D0 – D7). Not always, but Port 2 may be used as higher byte of the address bus.
- In this tutorial, we have seen the 8051 Microcontroller Memory Organization, Program Memory, Data Memory, Internal ROM and RAM and how to interface external Memory (ROM and RAM) with 8051 Microcontroller.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru



Interfacing External Memory (Ram And Rom) With 8051

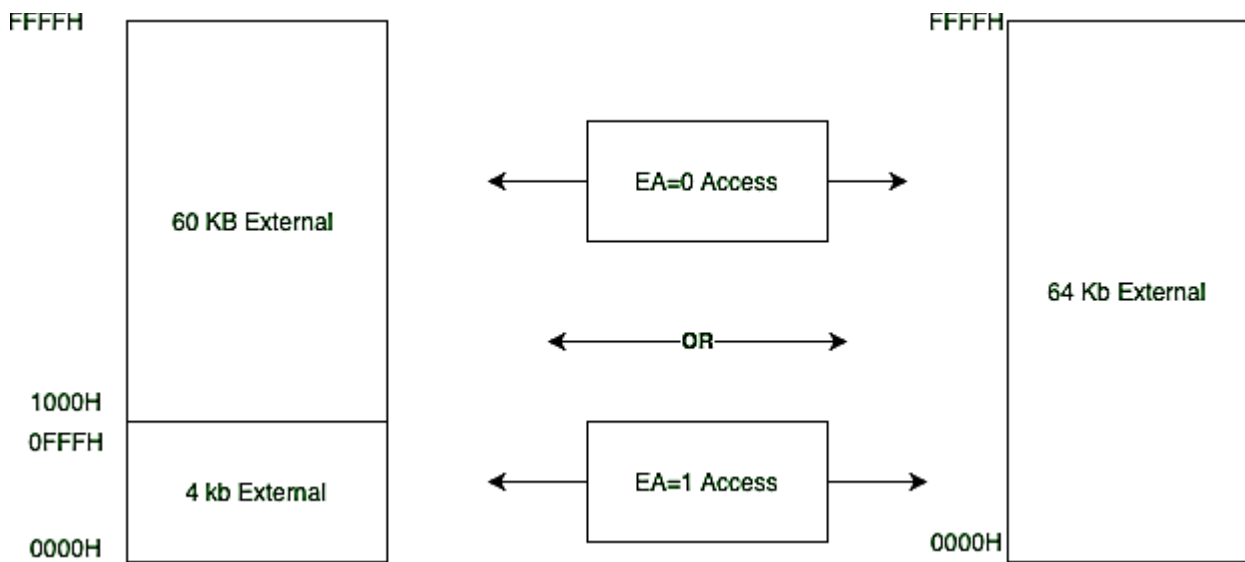


Dr. Vijaya Kumar H R
Associate. Professor, Dept. of
ECE, AIT, Tumakuru

- The memory types are illustrated in the following graphic. They are: On-Chip Memory, External Code Memory, and External RAM.
 - **On-Chip Memory** refers to any memory (Code (4K), RAM(128)BYTES, or other) that physically exists on the microcontroller itself. On-chip memory can be of several types, but we'll get into that shortly.
 - **External Code Memory is code (or program) memory** that resides off-chip. This is often in the form of an external EPROM.
 - **External RAM** is RAM memory that resides off-chip. This is often in the form of standard static RAM or flash RAM.
- **Code Memory:** Code memory is the memory that holds the actual 8051 program that is to be run. This memory is limited to 64K and comes in many shapes and sizes: Code memory may be found on-chip, either burned into the microcontroller as ROM or EPROM. Code may also be stored completely off-chip in an external ROM or, more commonly, an external EPROM. Flash RAM is also another popular method of storing a program. Various combinations of these memory types may also be used--that is to say, it is possible to have 4K of code memory on-chip and 64k of code memory off-chip in an EPROM. When the program is stored on-chip the 64K maximum is often reduced to 4k, 8k, or 16k. This varies depending on the version of the chip that is being used. Each version offers specific capabilities and one of the distinguishing factors from chip to chip is how much ROM/EPROM space the chip has. However, code memory is most commonly implemented as off-chip EPROM. This is especially true in low-cost development systems and in systems developed by students. Programming Tip: Since code memory is restricted to 64K, 8051 programs are limited to 64K. Some assemblers and compilers offer ways to get around this limit when used with specially wired hardware. However, without such special compilers and hardware, programs are limited to 64K.
- **External RAM** As an obvious opposite of Internal RAM, the 8051 also supports what is called External RAM. As the name suggests, External RAM is any random access memory which is found off-chip. Since the memory is off-chip it is not as flexible in terms of accessing, and is also slower. For example, to increment an Internal RAM location by 1 requires only 1 instruction and 1 instruction cycle. To increment a 1-byte value stored in External RAM requires 4 instructions and 7 instruction cycles. In this case, external memory is 7 times slower! What External RAM loses in speed and flexibility it gains in quantity. While Internal RAM is limited to 128 bytes (256 bytes with an 8052), the 8051 supports External RAM up to 64K

Dr. Vijaya Kumar H R

Associate Professor, Dept. of ECE, AIT, Tumakuru



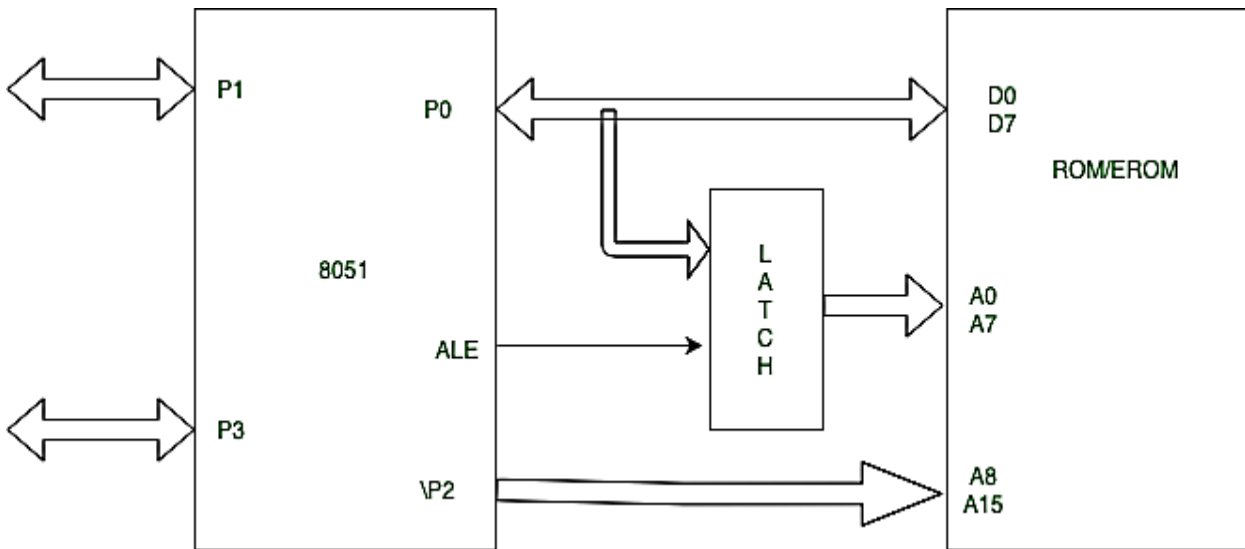
- The program fetches to addresses 0000H through 0FFFH are directed to the internal ROM in the 8051 when the EA pin is attached to Vcc, and program fetches to addresses 1000H through FFFFH are directed to the external ROM/EPROM. When the EA pin is grounded, all addresses fetched by the program (0000H to FFFFH) are led to it.. ROM/EPROM that is external to the device.

- As seen in Fig. NEXT PAGES, the PSEN signal is used to trigger output e external ROM/EPROM. Port 0 is used as a multiplexed address/bus, as seen in Fig.

- In the initial T-cycle, it provides a lower order 8-bit address, and later it is used as a data bus. The external latch and the ALE signal provided by the 8051 are used to latch the 8-bit address.

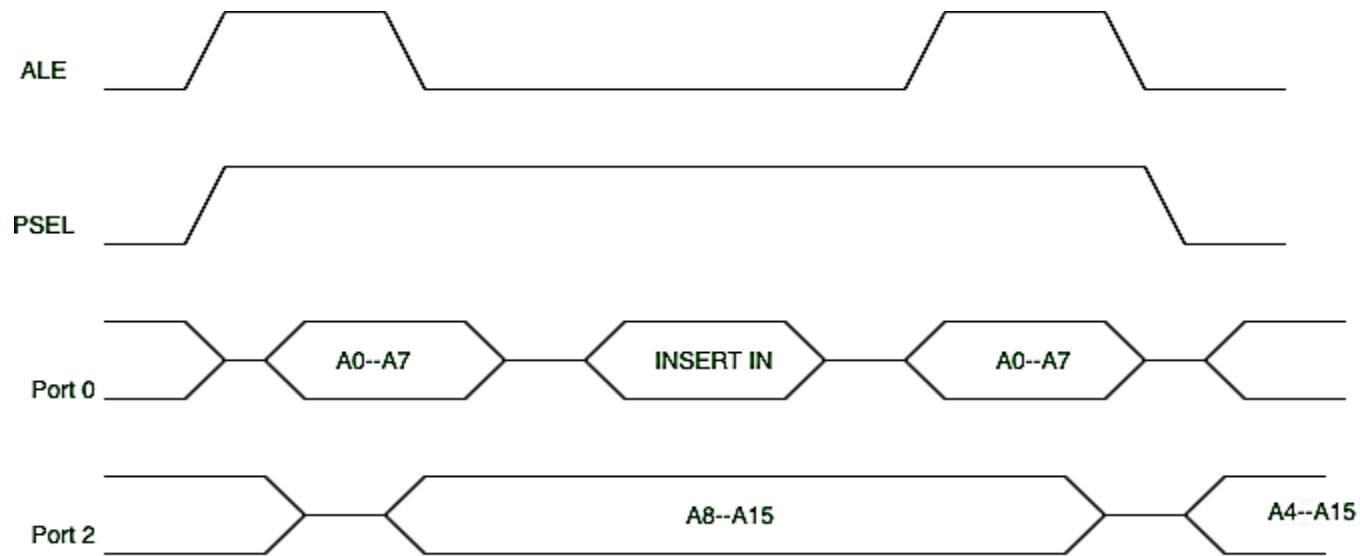
- Remote ROM/EPROM(Read Only Memory/Electronic Programmable ROM/Electronic Program The PSEN signal is used to activate output e external ROM/EPROM.

- ort 0 is used as a multiplexed address/bus. It supplies a lower-order 8-bit address in the first T-cycle and later serves as a data bus. The 8-bit address is latched using the external latch and the ALE signal given by the 8051.



External data memory (RAM)

- External data memory is read or write memory. Since external data memory is indirectly accessed through a data pointer register, DPTR (which must be loaded with an address), it is a slower process compared to accessing the internal data memory.
- Note that the additional 128 bytes of memory is interfaced with RAM by the use of RD read signal, RD = 1 (is active) when reading bytes from the external data memory (RAM). The command used to access data from external RAM is,



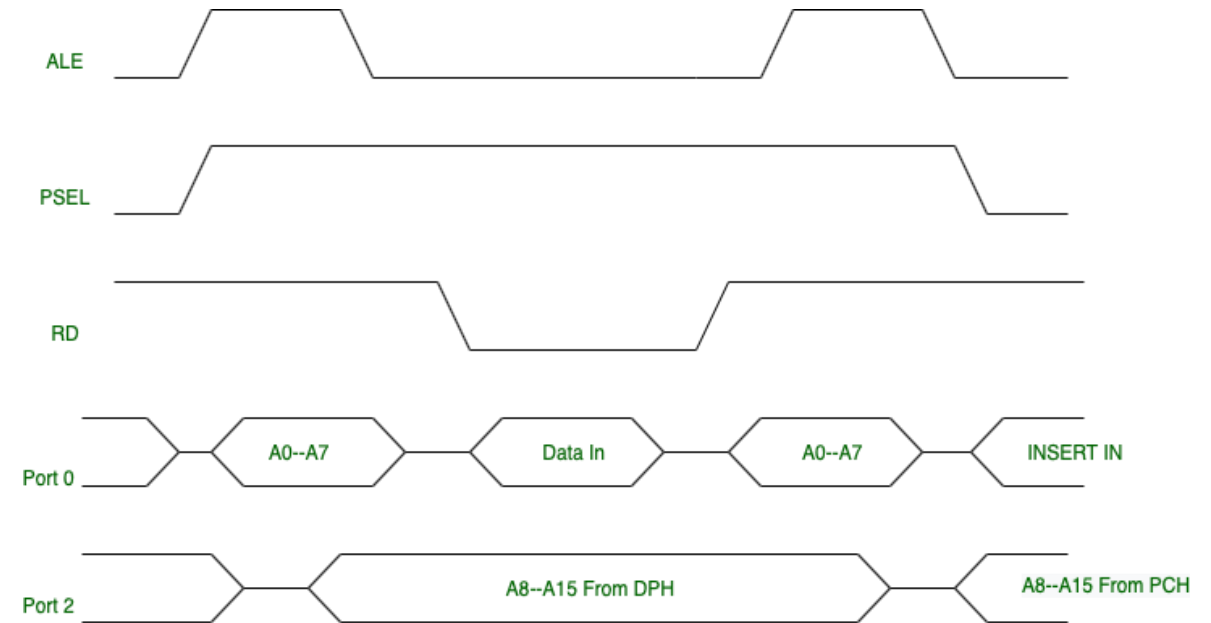
timing waveform for external data memory write cycle

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

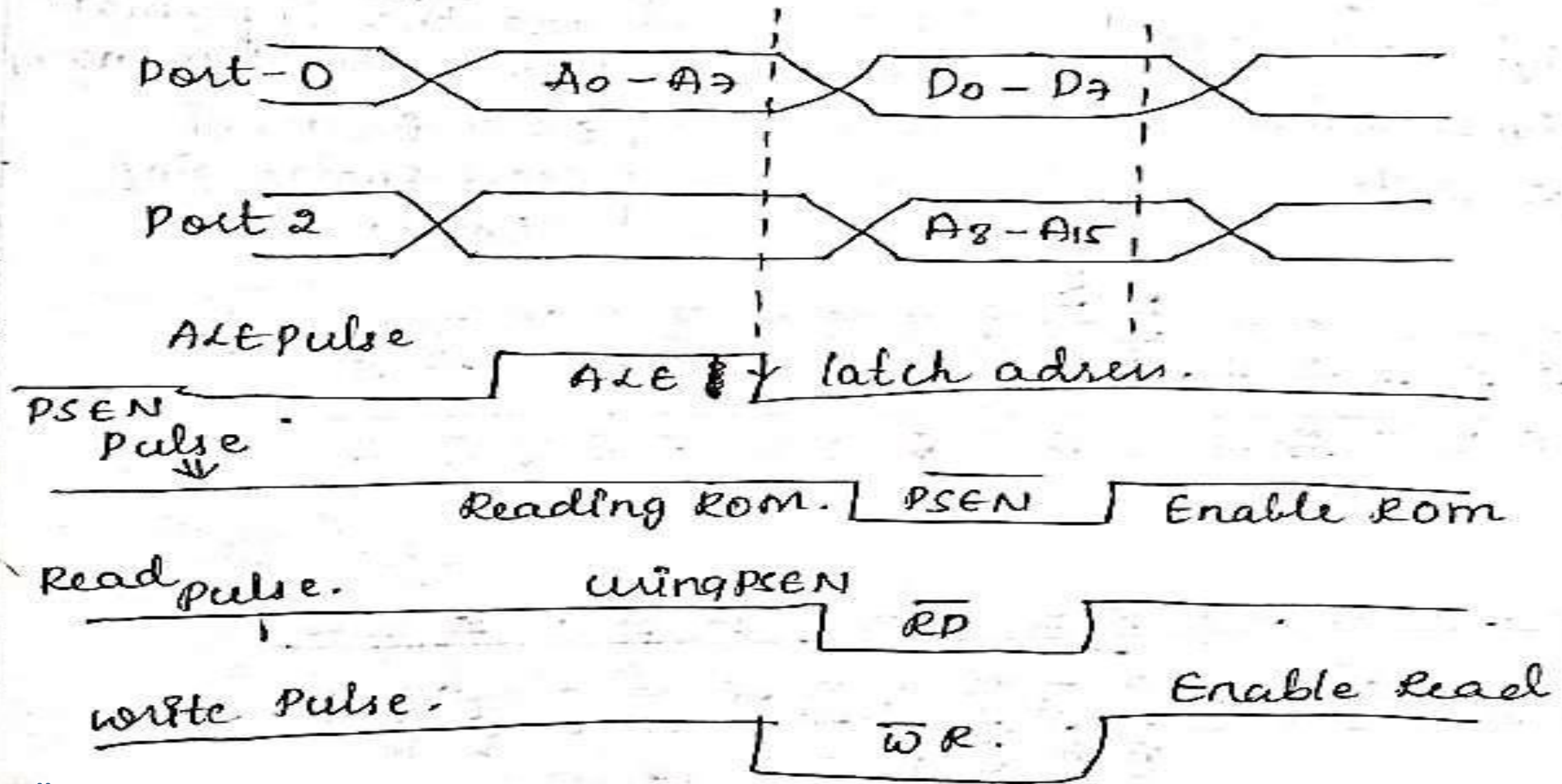
• External Interrupt 0 is 0003H, Timer 0 is 000BH, External Interrupt 1 is 0013H, Timer 1 is 001BH, and so on. If an interrupt is to be used, the operation routine for it must be in the same place as the interrupt. If the interrupt isn't used, the service location may be used as general-purpose program memory.

MOVX A, @Rp	In this operation, it will copy the contents of the external address in Rp to A.
MOVX A, @DPTR	Copy the contents of the external address in DPTR to A.
MOVX @Rp, A	Copy data from A to the external address in Rp
MOVX DPTR, A	Copy data from A to the external address in DPTR



timing waveform for external data memory read cycle

Connecting to External Memory:-



i. External ROM (program memory) Interfacing

Dr. Vijaya Kumar H R
Associate Professor, Dept. of ECE, AIT, Tumakuru

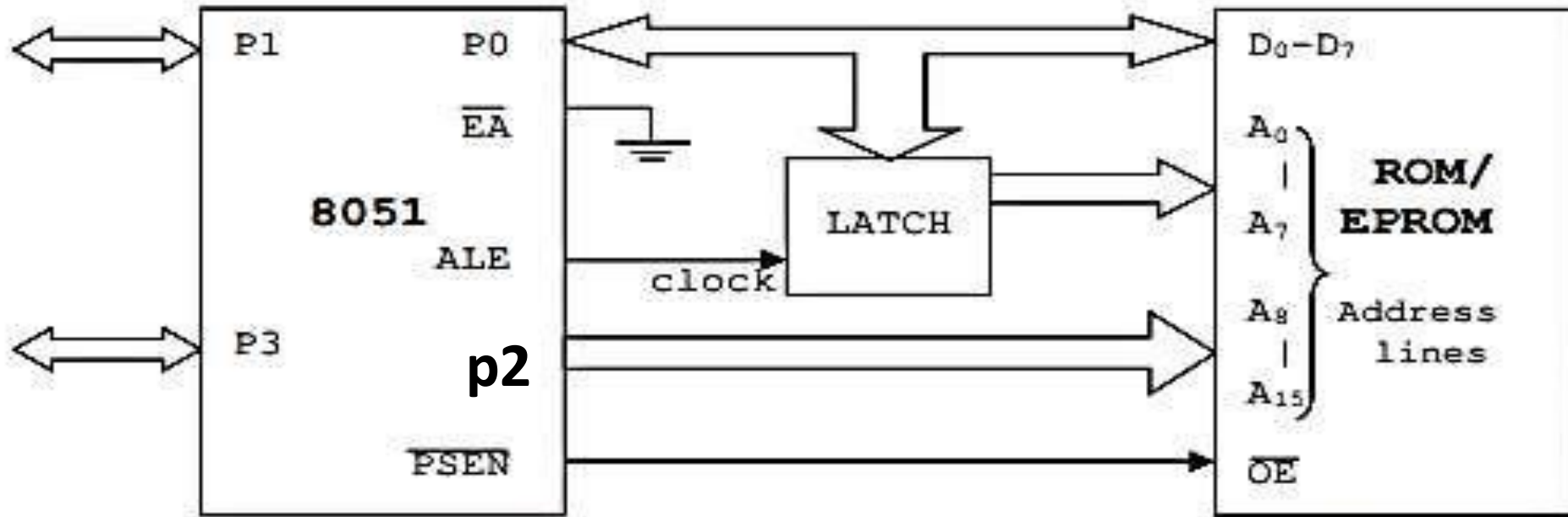


FIGURE 1 INTERFACING OF ROM/EPROM TO μC 8051.

above figure shows how to access or interface ROM to 8051.
port 0 is used as multiplexed data & address lines.
it gives lower order (A₇-A₀) 8 bit address in initial T cycle & higher order (A₈-A₁₅) used as data bus.
8 bit address is latched using external latch & ALE signal from 8051.
port 2 provides higher order (A₁₅-A₈) 8 bit address.
PSEN is used to activate the output enable signal of external ROM/EPROM.

Example 2: Design a μ Controller system using 8051. Interface the external ROM of size 4k x 8.

Solution: Given, Memory size: 4k

that means we require $2^n = 4k :: n$ address lines

here $n=12 :: A_0$ to A_{11} address lines are required.

remaining lines A_0, A_0, A_0, A_0 & PSEN are connected through OR gate to CS & RD of external ROM.

when A_0 to A_0 are low (logic '0'), only then external ROM is selected.

Address Decoding (Memory Map) for 4k x 8 RAM.

Address	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	HEX adrs.
starting	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
end	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0FFFH

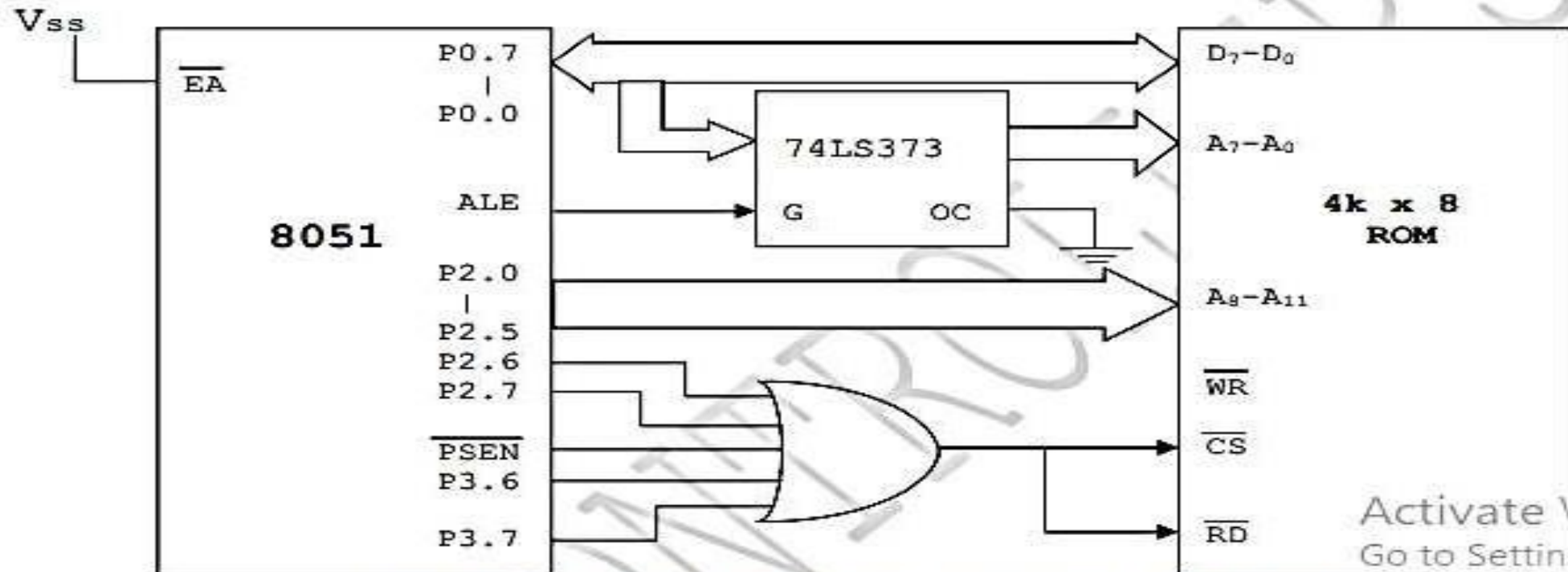


FIGURE 6 4K X 8 MEMORY (ROM) INTERFACING TO μ C 8051.

ii. External RAM (data memory) Interfacing

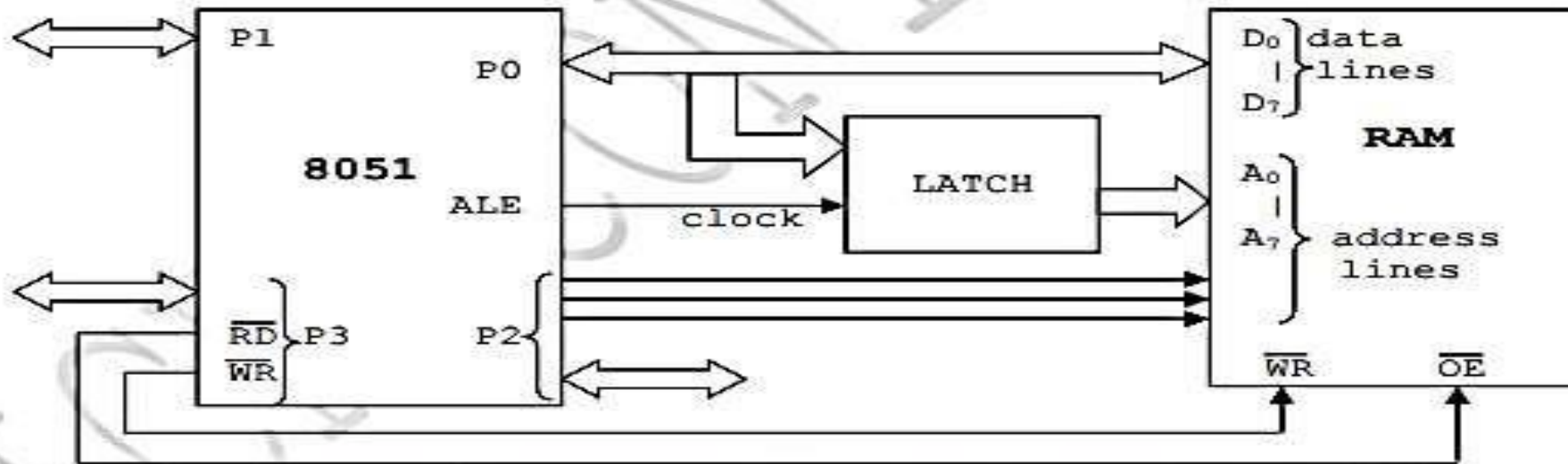


FIGURE 2 INTERFACING OF RAM(DATA MEMORY) TO μ C 8051.

above figure shows how to connect or interface external RAM(data memory) to 8051.

port 0 is used as multiplexed data & address lines.

address lines are decoded using external latch & ALE signal from 8051 to provide lower order (A7-A0) address lines.

port 2 gives higher order address lines.

RD & WR signals from 8051 selects the memory read & memory write operations respectively.

RD & WR signals: generally P3.6 & P3.7 pins of port 3 are used to generate memory read and memory write signals.
remaining pins of port 3 i.e. P3.0-P3.5 can be used for other functions.

Example 1: Design a μ Controller system using 8051. Interface the external RAM of size $16k \times 8$.

Solution: Given, Memory size: $16k$

that means we require $2^n = 16k :: n$ address lines

here $n=14 :: A_0$ to A_{13} address lines are required.

A_{14} and A_{15} are connected through OR gate to CS pin of external RAM.

when A_{14} and A_{15} both are low (logic '0'), external data memory (RAM) is selected.

Address Decoding (Memory Map) for $16k \times 8$ RAM.

Address	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	HEX adrs.
starting	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
end	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3FFFH

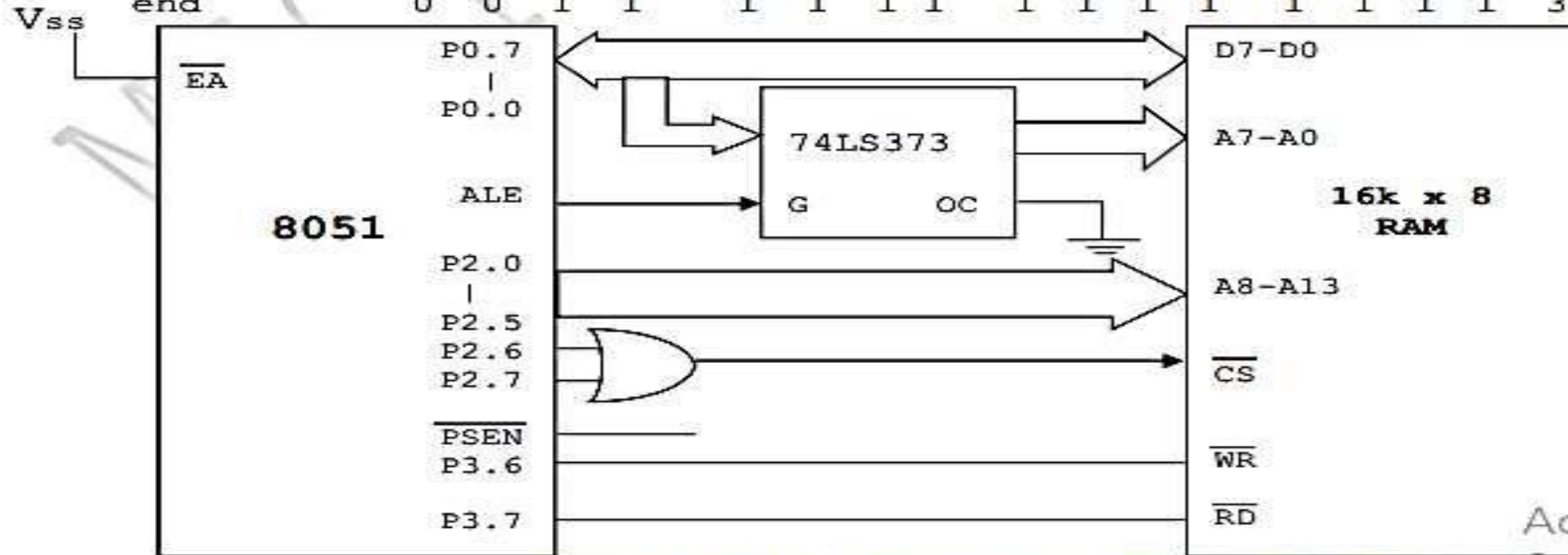


FIGURE 5 16K X 8 MEMORY (RAM) INTERFACING TO μ C 8051

Example 3: Design a μ Controller system using 8051, 16k bytes of ROM & 32k bytes of RAM. Interface the memory such that starting address for ROM is 0000H & RAM is 8000H.

Solution: Given, Memory size- ROM : 16k
that means we require $2^n=16k :: n$ address lines
here $n=14 :: A_0$ to A_{13} address lines are required.

$A_{14}, A_{15}, \overline{PSEN} \rightarrow \text{ORed} \rightarrow \overline{CS}$

when low - ROM is selected.

Dr. Vijaya Kumar H R
Associate. Professor, Dept. of ECE, AIT,
Tumakuru

Memory size- RAM : 32k
that means we require $2^n=32k :: n$ address lines
here $n=15 :: A_0$ to A_{15} address lines are required.

$A_{15} \rightarrow \text{inverted (NOT Gate)} \rightarrow \overline{CS}$

when high- RAM is selected.

\overline{PSEN} is used as chip select pin ROM.
 \overline{RD} is used as read control signal pin.
 \overline{WR} is used as write control signal pin.

for RAM
selection.

Example 4: Design a μ Controller system using 8051, 8k bytes of program ROM & 8k bytes of data RAM. Interface the memory such that starting address for ROM is 0000H & RAM is E000H.

Solution: Given, Memory size- ROM : 8k

that means we require $2^n=8k :: n$ address lines

here $n=13 :: A_0$ to A_{12} address lines are required.

$A_{13}, A_{14}, A_{15}, \overline{PSEN}$ \longrightarrow ORed $\longrightarrow \overline{CS}$

when low - program ROM is selected.

Dr. Vijaya Kumar H R

Associate. Professor, Dept. of ECE, AIT, Tumakuru

Memory size- RAM : 8k

that means we require $2^n=8k :: n$ address lines

here $n=13 :: A_0$ to A_{12} address lines are required.

A_{13}, A_{14}, A_{15} \longrightarrow NANDed $\longrightarrow \overline{CS}$

when high- data RAM is selected.

\overline{PSEN} is used as chip select pin ROM.

\overline{RD} is used as read control signal pin.

\overline{WR} is used as write control signal pin.

for RAM selection.

Address Decoding (Memory Map) for 8k x 8 ROM.

Address	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	HEX adrs.
starting	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000H
end	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFFH

Address Decoding (Memory Map) for 8k x 8 RAM.

Address	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	HEX adrs.
starting	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	E000H
end	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	FFFFH

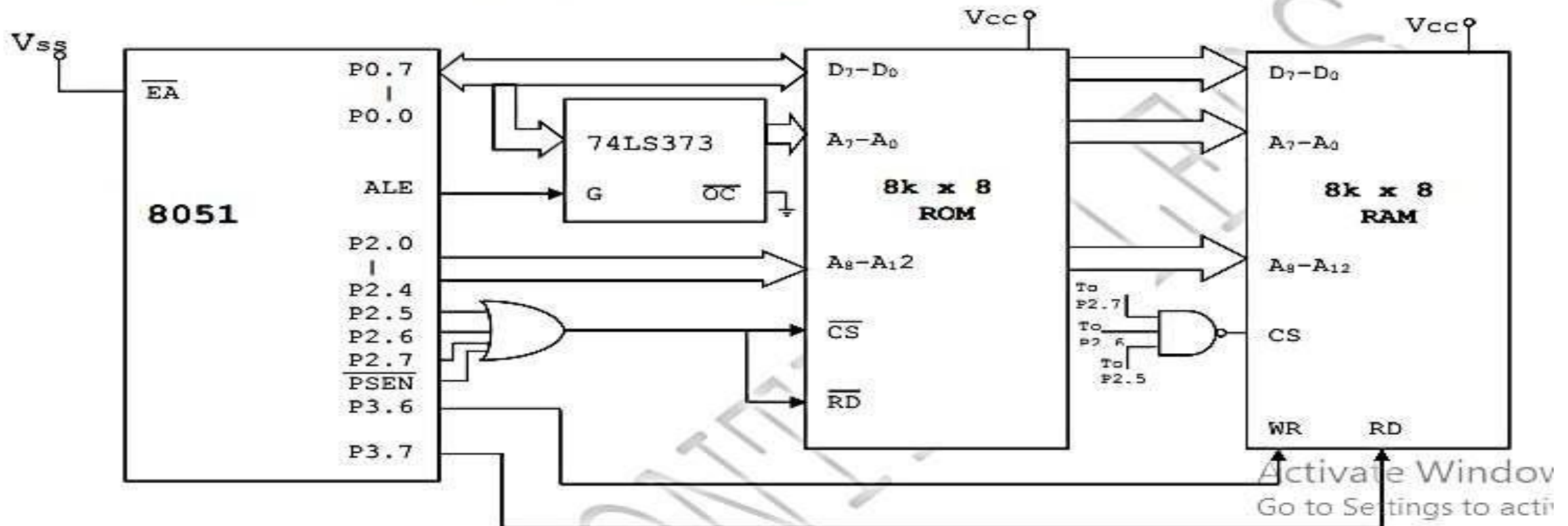


FIGURE 8 8K X 8 ROM AND 8K X 8 RAM INTERFACING TO μ C 8051.

END OF MODULE 1