BDeBooks.com

E-BOOK

www.BDeBooks.com
FB.com/BDeBooksCom
BDeBooks.Com@gmail.com

বাংলাদেশ কারিগরি শিক্ষা বোর্ডের নতুন সিলেবাস অনুযায়ী প্রকৌশল ডিপ্লোমা শিক্ষাক্রমের ৭ম পর্ব : ইলেকট্রনিক্স, ইলেকট্রোমেডিক্যাল এবং মেকাট্রনিক্স টেকনোলজির ছাত্র/ছাত্রীদের জন্য প্রণয়ন করা হয়েছে।

## বিষয় কোড : ৬৮৭১

# মাইক্রোকন্ট্রোলার অ্যান্ড ইমবেডেড সিস্টেমস্

# Microcontroller & Embedded Systems

**কেন এ বইটি অনন্য :**

- বাংলাদেশ কারিগরি শিক্ষা বোর্ডের সর্বশেষ সিলেবাস ও প্রশ্নের ধারা অনুযায়ী প্রণীত।
- কারিগরি শিক্ষা বোর্ডের বিগত বছরের প্রশ্নের সমাধান।
- সিলেবাস অনুসারে ব্যবহারিক সংযোজন করা হয়েছে।
- গবেষণামূলক সাজেশন সংযোজন করা হয়েছে।


**রচনায়**

| | |
|---|---|
| **অনিমেশ চন্দ্র সূত্রধর**<br>ইনস্ট্রাক্টর ও বিভাগীয় প্রধান (ইলেকট্রনিক্স)<br>কুমিল্লা পলিটেকনিক ইনস্টিটিউট, কুমিল্লা। | **গাজী সাইফুল ইসলাম**<br>বিএসসি ইঞ্জিনিয়ারিং (ইইই)<br>বিভাগীয় প্রধান ও জুনিয়র ইনস্ট্রাক্টর (ইলেকট্রনিক্স)<br>পটুয়াখালী পলিটেকনিক ইনস্টিটিউট, পটুয়াখালী। |
| **শামীম আহমদ**<br>এমএসসি ইন: টেকনিক্যাল এডুকেশন (ইইই) ও আইসি<br>জুনিয়র ইনস্ট্রাক্টর (ইলেকট্রনিক্স)<br>চট্টগ্রাম পলিটেকনিক ইনস্টিটিউট, চট্টগ্রাম। | **শারমিন সুলতানা**<br>জুনিয়র ইনস্ট্রাক্টর (ইলেকট্রনিক্স)<br>বরিশাল পলিটেকনিক ইনস্টিটিউট<br>বরিশাল। |
| **মোঃ মকছুদ আলম**<br>জুনিয়র ইনস্ট্রাক্টর (ইলেকট্রনিক্স)<br>হবিগঞ্জ পলিটেকনিক ইনস্টিটিউট, হবিগঞ্জ। | **হাসনাত রেহানা**<br>জুনিয়র ইনস্ট্রাক্টর (ইলেকট্রনিক্স)<br>কুমিল্লা পলিটেকনিক ইনস্টিটিউট, কুমিল্লা। |

**সম্পাদনায়**

**মোছাঃ রুমানা তাবাসুম**
বিএসসি-ইন ইলেকট্রনিক্স অ্যান্ড টেলিকমিউনিকেশন
ইনস্ট্রাক্টর (ইলেকট্রনিক্স)
কুড়িগ্রাম পলিটেকনিক ইনস্টিটিউট, কুড়িগ্রাম।



**প্রকাশনায়**

**টেকনিক্যাল প্রকাশনী**
৩৮ বাংলাবাজার (৪র্থ তলা), ঢাকা- ১১০০।

প্রকাশক : টেকনিক্যাল প্রকাশনীর পক্ষে
মোঃ তালহা ইসলাম (বাবু)
বেগম তাসলিমা ইসলাম
৩৮, বাংলাবাজার (৪র্থ তলা), ঢাকা– ১১০০।
ফোন : ০২-৯৫৬৩২৩২, ০২-৯৫১৪২৯০।

ম্যানেজার মোবাইল : ০১৯৪২২৭৪৭৪৬, ০১৬৮৭৭২৪২৬২, ০১৯২৩৭০৪০২৬, ০১৯৬৩৮২৫৬৬৬, ০১৭১৭১৭১৮৭৩, ০১৯৫৪০০০৯৫৪।

বিকাশ মোবাইল : ০১৯৮৪৭৫৫১১৫, ০১৯৫৯৯৯৫৫৪০ (পার্সোনাল), ০১৭১২৫৮৪৪৫৩ (মার্চেন্ট, পেমেন্ট)।

মোবাইল ম্যাসেজ অর্ডার : ০১৯৪২২৭৪৭৪৬, ০১৯২৩৭০৪০২৬।

ডেলিভারি কাউন্টার : ০১৯৬৩৮২৬৬৯৯, ০১৭১৭১৭১৮৭৩।
ই-মেইল : technicalprocashoni@gmail.com
http://www.facebook.com/technicalprocashoni/
www.technicalprokashoni.com

গ্রন্থস্বত্ব : প্রকাশক কর্তৃক গ্রন্থস্বত্ব সংরক্ষিত

প্রথম সংস্করণ : ২০০৬ ইং
দ্বিতীয় সংস্করণ : ২০১১ ইং
তৃতীয় পরিমার্জিত সংস্করণ : ২০১৬ ইং
চতুর্থ পরিমার্জিত সংস্করণ : ২০১৭ ইং

প্রচ্ছদ পরিকল্পনায় : জনাব আলহাজ্ব মোঃ সাইফুল ইসলাম
মোবাইল : ০১৭১২৫৮৪৪৫৩, ০১৯১৭৬৬৬০৫৮

বর্ণবিন্যাসে : শাপলা অ্যান্ড তালহা কম্পিউটার সেন্টার
৩৮, বাংলাবাজার (৪র্থ তলা)
ঢাকা– ১১০০।

মুদ্রণে : টেকনিক্যাল প্রিন্টিং প্রেস
বাংলাবাজার, ঢাকা।

মূল্য : ২০০.০০/- (দুইশত) টাকা মাত্র।

৬৮৭১- মাই: ইম: সিস্টে: - ১ (খ)

# উৎসর্গ

অসহায় ও দুঃখী
মানুষের জন্য

# লেখকগণের কথা

**6871 MICROCONTROLLER AND EMBEDDED SYSTEMS**

| T | P | C |
|---|---|---|
| 2 | 6 | 4 |

## Theory

**1. Understand the Fundamentals of Micro controller.**

1.1 Define Microcontroller and Microprocessor.
1.2 List the basic building blocks of a Microcontroller.
1.3 Compare Microprocessor and Microcontroller.
1.4 Mention the types of Microcontroller.
1.5 Describe the Harvard and Von-Neumann architecture.
1.6 Compare CISC and RISC.
1.7 Mention the field of Microcontroller applications.
1.8 List the commercial Microcontrollers with salient features
1.9 Mention the criteria for choosing a microcontroller

**2. Understand Features of the Intel 8051 Microcontroller and PIC.**

2.1 Define 8051 family.
2.2 Mention the features of the Intel 8051 microcontroller and PIC.
2.3 Compare the features of different member of the 8051 family.

**3. Understand Architecture of the Intel 8051 Microcontroller**

3.1 Describe the simplified Block diagram of the Intel 8051 microcontroller.
3.2 Explain the programming model of the 8051 microcontroller.
3.3 Describe the memory organization of the 8051 microcontroller mentioning the function of SFR, Register bank, bit addressable & general purpose RAM.
3.4 State the function of each flag of the PSW register.
3.5 Explain the pin and signal of the 8051 microcontroller.
3.6 Describe the Clock and Reset circuits of the 8051.
3.7 Compare Atmel 89C2051 and 89C4051 with 8051.

**4. Understand the Instruction Set of 8051 Microcontroller.**

4.1 Define Instruction and Instruction set.
4.2 Classify the 8051 instruction
4.3 Describe the function of each instruction.
4.4 Discuss short, absolute and long range.
4.5 Explain the Addressing modes of 8051.

**5. Understand the Assembly Language Programming of the Intel 8051 Microcontroller.**

5.1 Define (i) Assembly language (ii) Assembler and (iii) Linker (iv) IDE.
5.2 Mention the fields of an assembly language instruction.
5.3 State the meaning of CPU Instruction and Assembler directives.
5.4 List the commonly used Assembler directives.
5.5 Mention the steps to create and execute assembly language program.
5.6 Identify the files in assembly language program.
5.7 Explain the step-by-step action of the 8051 upon applying power on it.
5.8 Write simple assembly language program for the 8051.

**6. Understand the Subroutine.**

6.1 Define Subroutine.
6.2 Mention the Advantages of Subroutine.
6.3 Describe calling and returning from Subroutine.
6.4 Write Subroutine for creating delay.

**7. Understand Programming 8051 in C.**

7.1 Mention the reasons for writing program in C.
7.2 List C data types and operators for 8051
7.3 Describe creating time delay in C.
7.4 Write program in C for sending data to port, Accessing code ROM, Data serialization and interrupt operation.

**8. Understand I/O port Programming.**

8.1 List the I/O ports of 8051.
8.2 Mention the purpose of Pull-up resistor.
8.3 Write code for configuring port for Input.
8.4 Write program for receiving and sending data through I/O port
8.5 Describe the alternate functions of Port-3.

**9. Understand the 8051 Timer/counter**

9.1 List the function of a timer.
9.2 Discuss the mode of operation of timer.
9.3 Describe the function of each bit of TMOD & TCON Register.
9.4 Write code for setting timer in different mode.
9.5 Explain the procedure of starting, stopping and controlling timer.
9.6 Calculate the initial value of timer for creating a certain delay.
9.7 Write subroutine for creating delay of certain amount of time using Timer.
9.8 Develop program for generating square wave and PWM .
9.9 Describe the Timer as an event counter.

**10. Understand 8051 Serial communication.**

10.1 Define serial and parallel communication.
10.2 Describe the RS232 9-pin D type connector.
10.3 Describe the pins and connections of line driver(Such as MAX232).
10.4 Describe the function of each bit of SCON register.
10.5 Describe the modes of serial communication of the 8051.
10.6 Describe configuring and setting Baud rate of serial port.
10.7 Mention the purpose of SBUF register.
10.8 Write program to transmit and receive data through serial port.

**11. Under Stand the Interrupt of the 8051 Microcontroller.**

11.1 List the source of interrupt of the 8051.
11.2 Define Interrupt service routine (ISR).
11.3 Mention the interrupt priority and vector locations.
11.4 Describe each bit of the interrupt enable (IE) register.
11.5 Describe the procedure of enabling and disabling interrupt.
11.6 Mention the steps in executing an interrupt.
11.7 Describe the register protection during interrupt.
11.8 Describe External Hardware, Timer and Serial communication Interrupt.
11.9 State the common problem with interrupt.

**12. Under Stand LCD and Keyboard Interfacing.**

12.1 Describe the pin diagram of LCD.
12.2 Describe the Instruction register, data register and busy flag.

12.3 List the LCD command codes,
12.4 Write Program for displaying data to LCD.
12.5 Describe the organization of a matrix Keyboard.
12.6 Explain the steps to detect and identify the key pressed.
12.7 Write a keyboard subroutine to send the ASCII code for pressed key to a port.

**13. Understand real world interfacing**

13.1 Describe interfacing the 8051 with a DC motor.
13.2 Describe interfacing the 8051 with a stepper motor.
13.3 Describe interfacing ADC chips to the 8051.
13.4 Describe interfacing the DAC chips to the 8051.
13.5 Describe interfacing temperature sensor to the 8051.
13.6 Develop program for above interfacing circuits.
13.7 Explain how to use off-chip memory with the 8051.
13.8 Describe interfacing the 8051 with a Dot matrix display.

## Practical

**1. Test a program to send a data to port P1 and p2.**

1.1 Draw the flow chart.
1.2 Start a simulator program.
1.3 Type the program.
1.4 Save the program.
1.5 Compile the program
1.6 Run the Simulator and observe the execution of the program.

**2. Test a program to copy 8 bytes of data from RAM location staring at 30H to RAM location starting 50H.**

2.1 Draw the flow chart.
2.2 Start an IDE .
2.3 Type the program.
2.4 Save the program.
2.5 Compile the program
2.6 Run the Simulator and observe the execution of the program.

**3. Test a program to transfer the 16 bytes of data from ROM location staring at 30H to RAM location starting 50H.**

3.1 Draw the flow chart.
3.2 Start an IDE .
3.3 Type the program.
3.4 Save the program.
3.5 Compile the program
3.6 Run the Simulator and observe the execution of the program.

**4. Test a Program to find the sum of the values of RAM locations from 50H to 55H. Register A should contain the low byte and R7 the high byte.**

4.1 Draw the flow chart.
4.2 Start an IDE .
4.3 Type the program.
4.4 Save the program.
4.5 Compile the program
4.6 Run the Simulator and observe the execution of the program.

**5. Test a program to get the vale of x and send $x^2$ to p2 continuously.**
- 5.1 Draw the flow chart.
- 5.2 Start an IDE .
- 5.3 Type the program.
- 5.4 Save the program.
- 5.5 Compile the program
- 5.6 Run the Simulator and observe the execution of the program.

**6. Test a program to get Hex data in the range of 00-FFH from the port.**
- 6.1 Draw the flow chart.
- 6.2 Start an IDE .
- 6.3 Type the program.
- 6.4 Draw the Flow chart.
- 6.5 Type and save the program.
- 6.6 Compile the program.
- 6.7 Run the Simulator and observe the execution of the program.

**7. Develop and test a program for flashing LEDs.**
- 7.1 Draw the Flow chart.
- 7.2 Type and save the program.
- 7.3 Compile the program.
- 7.4 Download the Hex program to 8051 code memory.
- 7.5 Construct the circuit.
- 7.6 Power the circuit and observe the output.

**8. Develop and test a program for displaying 0 to 9 on 7-Segment display.**
- 8.1 Draw the Flow chart.
- 8.2 Type and save the program.
- 8.3 Compile the program.
- 8.4 Download the Hex program to 8051 code memory.
- 8.5 Construct the circuit.
- 8.6 Power the circuit and observe the output.

**9. Develop and test a program for Generating Square wave.**
- 9.1 Draw the Flow chart.
- 9.2 Type and save the program.
- 9.3 Compile the program.
- 9.4 Download the Hex program to 8051 code memory.
- 9.5 Construct the circuit.
- 9.6 Power the circuit and observe the output.

**10. Develop and test a program for Interfacing Keyboard.**
- 10.1 Draw the Flow chart.
- 10.2 Type and save the program.
- 10.3 Compile the program.
- 10.4 Download the Hex program to 8051 code memory.
- 10.5 Construct the circuit.
- 10.6 Power the circuit and observe the output.

**11. Develop and test a program for Interfacing LCD.**
- 11.1 Draw the Flow chart.
- 11.2 Type and save the program.
- 11.3 Compile the program.
- 11.4 Download the Hex program to 8051 code memory.
- 11.5 Construct the circuit.
- 11.6 Power the circuit and observe the output.

**12. Develop and test a program for Interfacing DC Motor.**
- 12.1 Draw the Flow chart.
- 12.2 Type and save the program.
- 12.3 Compile the program.
- 12.4 Download the Hex program to 8051 code memory.
- 12.5 Construct the circuit.
- 12.6 Power the circuit and observe the output.

**13. Develop and test a program for Interfacing Stepper Motor.**
- 13.1 Draw the Flow chart.
- 13.2 Type and save the program.
- 13.3 Compile the program.
- 13.4 Download the Hex program to 8051 code memory.
- 13.5 Construct the circuit.
- 13.6 Power the circuit and observe the output.

**14. Develop and test a program for Interfacing DAC.**
- 14.1 Draw the Flow chart.
- 14.2 Type and save the program.
- 14.3 Compile the program.
- 14.4 Download the Hex program to 8051 code memory.
- 14.5 Construct the circuit.
- 14.6 Power the circuit and observe the output.

**15. Develop and test a program for Interfacing ADC.**
- 15.1 Draw the Flow chart.
- 15.2 Type and save the program.
- 15.3 Compile the program.
- 15.4 Download the Hex program to 8051 code memory.
- 15.5 Construct the circuit.
- 15.6 Power the circuit and observe the output.

**16. Develop and test a program for Interfacing Temperature sensor.**
- 16.1 Draw the Flow chart.
- 16.2 Type and save the program.
- 16.3 Compile the program.
- 16.4 Download the Hex program to 8051 code memory.
- 16.5 Construct the circuit.
- 16.6 Power the circuit and observe the output.

**17. Develop and test a program for dot matrix display.**
- 17.1 Draw the Flow chart.
- 17.2 Type and save the program.
- 17.3 Compile the program.
- 17.4 Download the Hex program to 8051 code memory.
- 17.5 Construct the circuit.
- 17.6 Power the circuit and observe the output.

## REFERENCE BOOKS


1. The 8051 Microcontroller and Embedded system_ Mazidi
2. The 8051 Microcontroller _ I. Scott MacKenzie
3. 8051 Tutorial_ Donal Heffernan
4. 8051 Microcontrollers. An Applications-Based Introduction_David Calcutt\Fred Cowan; Hassan Parchizadeh
5. Microcontrollers Theory and Application - Ajay V Deshmukh

## মানবণ্টন

১। সমাপনী পরীক্ষা (তাত্ত্বিক)

**ক-বিভাগ** — **অতি সংক্ষিপ্ত প্রশ্ন**

(১৫টি প্রশ্ন থাকবে। ১৫টির উত্তর দিতে হবে। প্রতিটি প্রশ্নের মান– ২।) ২ × ১০ = ২০

**খ-বিভাগ** — **সংক্ষিপ্ত প্রশ্ন**

(১০টি প্রশ্ন থাকবে। ১০টির উত্তর দিতে হবে। প্রতিটি প্রশ্নের মান– ৩।) ৩ × ১০ = ৩০

**গ-বিভাগ** — **রচনামূলক প্রশ্ন**

(৬টি প্রশ্ন থাকবে। ৫টির উত্তর দিতে হবে। প্রতিটি প্রশ্নের মান– ৬।) ৬ × ৫ = ৩০

**সর্বমোট : ৮০**


**অধিক কমনের নিশ্চয়তায়**

**টেকনিক্যাল প্রকাশনীর**

**"টেকনিক্যাল গাইড"**

**গুণে ও মানে বাজারের সেরা গাইড।**

**"টেকনিক্যাল গাইড" পড়ুন**

**আপনার সাফল্য নিশ্চিত করুন।**

# সূচিপত্র

## অধ্যায়- ১ মাইক্রোকন্ট্রোলারের মৌলিক ধারণা


| | | |
|---|---|---|
| ১.০ | ভূমিকা | ১৫ |
| ১.১ | মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের সংজ্ঞা | ১৫ |
| ১.১.১ | মাইক্রোকন্ট্রোলার | ১৫ |
| ১.১.২ | মাইক্রোপ্রসেস | ১৬ |
| ১.২ | মাইক্রোকন্ট্রোলারের মৌলিক ব্লক ডায়াগ্রামের তালিকা | ১৭ |
| ১.৩ | মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের মধ্যে পার্থক্য | ১৯ |
| ১.৪ | মাইক্রোকন্ট্রোলারের প্রকারভেদ উল্লেখ করণ | ২০ |
| ১.৫ | হার্ভার্ড ও ভননিউম্যান আর্কিটেকচার বর্ণনা | ২১ |
| ১.৫.১ | Harvard I Von Neumann আর্কিটেকচারের পার্থক্য | ২৪ |
| ১.৬ | সিআইএসসি ও আরআইএসসি এর তুলনা | ২৪ |
| ১.৬.১ | কমপ্লেক্স ইনস্ট্রাকশন সেট কম্পিউটার (সিআইএসসি) | ২৫ |
| ১.৬.২ | রিডিউসড ইনস্ট্রাকশন সেট কম্পিউটার (আরআইএসসি) মাইক্রোকন্ট্রোলার | ২৬ |
| ১.৭ | মাইক্রোকন্ট্রোলারের প্রয়োগক্ষেত্র উল্লেখকরণ | ২৭ |
| ১.৮ | বাণিজ্যিক মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য | ২৮ |
| ১.৯ | মাইক্রোকন্ট্রোলার নির্বাচনে শর্তসমূহ উল্লেখকরণ | ৩১ |


## অধ্যায়- ২ ইন্টেল 8051 ও পিআইসি মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য


| | | |
|---|---|---|
| ২.০ | ভূমিকা | ৫০ |
| ২.১ | MCS-51 পরিবার | ৫০ |
| ২.২ | ইন্টেল 8051 মাইক্রোকন্ট্রোলার ও PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য উল্লেখকরণ | ৫০ |
| ২.২.১ | ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য | ৫০ |
| ২.২.২ | PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য | ৫১ |
| ২.৩ | 8051 পরিবারের বিভিন্ন সদস্যের বৈশিষ্ট্য তুলনা | ৫২ |
| ২.৩.১ | MCS-51 এর প্রকারভেদ | ৫২ |
| ২.৩.২ | MCS-51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি | ৫২ |


## অধ্যায়- ৩ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের গঠন


| | | |
|---|---|---|
| ৩.০ | ভূমিকা | ৫৮ |
| ৩.১ | ইন্টেল 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম বর্ণনা | ৫৯ |
| ৩.২ | 8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল | ৬২ |
| ৩.৩ | 8051 মাইক্রোকন্ট্রোলারের মেমোরির গঠন | ৬৩ |

| | | |
|---|---|---|
| ৩.৩.১ | প্রোগ্রাম মেমোরি গঠন (অভ্যন্তরীণ ও বাহ্যিক) | ৬৩ |
| ৩.৩.২ | ডাটা মেমোরির গঠন (অভ্যন্তরীণ ও বাহ্যিক) | ৬৪ |
| ৩.৩.৩ | অভ্যন্তরীণ ডাটা মেমোরি | ৬৪ |
| ৩.৩.৪ | স্পেশাল ফাংশন রেজিস্টারস (এসএফআর) | ৬৬ |
| ৩.৪ | প্রোগ্রাম স্ট্যাটাস ওয়ার্ড (পিএসডব্লিউ) রেজিস্টার বর্ণনা | ৬৭ |
| ৩.৫ | 8051 মাইক্রোকন্ট্রোলারের পিন ও সিগন্যাল ডায়াগ্রাম ব্যাখ্যা | ৬৮ |
| ৩.৬ | 8051 মাইক্রোকন্ট্রোলারের ক্লক ও রিসেট সার্কিট | ৭০ |
| ৩.৬.১ | ক্লক সার্কিট | ৭০ |
| ৩.৬.২ | রিসেট সার্কিট | ৭১ |


## অধ্যায়- ৪ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের নির্দেশনার সেট


| | | |
|---|---|---|
| ৪.০ | ভূমিকা | ৮১ |
| ৪.১ | নির্দেশনা ও নির্দেশনা সেট | ৮৩ |
| ৪.২ | 8051 এর নির্দেশনার শ্রেণিবিভাগ | ৮৩ |
| ৪.৩ | প্রত্যেকটি নির্দেশনার কার্যাবলি বর্ণনা | ৮৩ |
| ৪.৪ | শর্ট রেঞ্জ, অ্যাবসলিউট ভ্যালু এবং লং রেঞ্জ আলোচনা | ৮৯ |
| ৪.৫ | 8051 এর অ্যাড্রেসিং মুড ব্যাখ্যা | ৯০ |


## অধ্যায়- ৫ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের অ্যাসেম্বলি ভাষা প্রোগ্রামিং


| | | |
|---|---|---|
| ৫.০ | ভূমিকা | ৯৭ |
| ৫.১ | অ্যাসেম্বলি প্রোগ্রামিং সম্পর্কিত সংজ্ঞা | ৯৭ |
| ৫.২ | অ্যাসেম্বলি ভাষার নির্দেশনার ক্ষেত্রসমূহ | ৯৯ |
| ৫.৩ | সিপিইউ নির্দেশনা ও অ্যাসেম্বলার ডিরেকটিভ বর্ণনা | ৯৯ |
| ৫.৪ | সচরাচর ব্যবহৃত অ্যাসেম্বলার ডিরেকটিভ এর তালিকা | ১০০ |
| ৫.৫ ও ৫.৬ | অ্যাসেম্বলি প্রোগ্রাম তৈরি ও সম্পাদনের ধাপ উল্লেখ করণ | ১০০ |
| ৫.৭ | 8051 মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর ধাপে ধাপে এর কাজের ব্যাখ্যা | ১০১ |
| ৫.৮ | একটি সরল অ্যাসেম্বলি ভাষা প্রোগ্রাম | ১০২ |


## অধ্যায়- ৬ সাবরুটিন


| | | |
|---|---|---|
| ৬.০ | ভূমিকা | ১০৭ |
| ৬.১ | সাবরুটিন | ১০৭ |
| ৬.২ | সাবরুটিনের সুবিধা উল্লেখ করণ | ১০৭ |
| ৬.৩ | সাবরুটিন আহ্বান ও প্রত্যাবর্তন বর্ণনা | ১০৮ |
| ৬.৪ | ডিলে উৎপন্ন করার জন্য একটি সরল সাবরুটিন প্রোগ্রাম | ১০৮ |

## অধ্যায়- ৭ সি ভাষায় 8051 মাইক্রোকন্ট্রোলার প্রোগ্রামিং


| | | |
|---|---|---|
| ৭.০ | ভূমিকা | ১১২ |
| ৭.১ | সি ভাষায় প্রোগ্রাম লিখার কারণ উল্লেখ করণ | ১১৪ |
| ৭.২ | 8051 এর জন্য সি ডাটার ধরন ও অপারেটরের তালিকা | ১১৪ |
| ৭.৩ | 8051 এর জন্য সি ভাষায় টাইম ডিলে তৈরির প্রক্রিয়া বর্ণনা | ১১৬ |
| ৭.৪ | ৮০৫১ এর জন্য সি প্রোগ্রাম | ১১৭ |


## অধ্যায়- ৮ ইনপুট/আউটপুট পোর্ট প্রোগ্রামিং


| | | |
|---|---|---|
| ৮.১ | ৮০৫১ এর ইনপুট/আউটপুট পোর্টের তালিকা | ১২৭ |
| ৮.২ | পুল-আপ রেজিস্টরের উদ্দেশ্য উল্লেখ করণ | ১২৭ |
| ৮.৩ | 8051 এর পোর্টকে ইনপুট হিসেবে সাজানোর প্রোগ্রাম | ১২৯ |
| ৮.৪ | ইনপুট/আউটপুট পোর্টের মাধ্যমে ডাটা গ্রহণ ও প্রেরণের প্রোগ্রাম | ১২৯ |
| ৮.৫ | পোর্ট 3 এর বিকল্প কাজ বর্ণনা | ১৩০ |


## অধ্যায়- ৯ 8051 এর টাইমার/কাউন্টার


| | | |
|---|---|---|
| ৯.১ | টাইমারের কাজের তালিকা | ১৩৪ |
| ৯.২ | 8051 এ টাইমার/কাউন্টার অপারেশনের সহযোগী রেজিস্টারসমূহ আলোচনা | ১৩৪ |
| ৯.২.১ | TMOD Register | ১৩৪ |
| ৯.২.২ | এঈঙঘ জবমরংঃবৎ | ১৩৫ |
| ৯.৩ | টাইমার অপারেশনের মুড আলোচনা | ১৩৫ |
| ৯.৪ | টাইমারকে বিভিন্ন মুডে সেটিং এর কোড | ১৩৮ |
| ৯.৫ | টাইমার আরম্ভ, বন্ধ ও নিয়ন্ত্রণ প্রক্রিয়া ব্যাখ্যা | ১৩৮ |
| ৯.৬ | একটি টাইম ডিলে তৈরির জন্য টাইমারের প্রাথমিক মানের হিসাব | ১৩৯ |
| ৯.৭ | টাইমার ব্যবহার করে 8051 প্রোগ্রাম | ১৪০ |
| ৯.৮ | জেনারেটিং স্কয়ার ওয়েভ ও PWM এর জন্য প্রোগ্রাম উন্নয়ন | ১৪০ |
| ৯.৯ | ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার | ১৪২ |


## অধ্যায়- ১০ ৮০৫১ সিরিয়াল কমিউনিকেশন


| | | |
|---|---|---|
| ১০.০ | ভূমিকা | ১৫১ |
| ১০.১ | সিরিয়াল ও প্যারালাল কমিউনিকেশন | ১৫১ |
| ১০.২ | আরএস 232 ডিবি 9 পিন কানেক্টর আলোচনা | ১৫২ |
| ১০.৩ | লাইন ড্রাইভারের পিন ও সংযোগ আলোচনা | ১৫৩ |
| ১০.৪ | ঝঈঙঘ রেজিস্টারের বিটগুলোর কাজ আলোচনা | ১৫৪ |
| ১০.৫ | 8051 এর সিরিয়াল কমিউনিকেশনের মুডসমূহের বর্ণন | ১৫৪ |
| ১০.৬ | সিরিয়াল পোর্টের Baud rate সেটিং ও সজ্জা বর্ণনা | ১৫৭ |
| ১০.৭ | SBUF রেজিস্টারের উদ্দেশ্য উল্লেখ | ১৫৮ |
| ১০.৮ | 8051 এর সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণ ও গ্রহণের প্রোগ্রাম | ১৫৯ |

## অধ্যায়- ১১ 8051 মাইক্রোকন্ট্রোলারের ইন্টারাপ্ট


| | | |
|---|---|---|
| ১১.০ | ভূমিকা | ১৬৭ |
| ১১.১ | 8051 এর ইন্টারাপ্ট এর উৎসের তালিকা | ১৬৭ |
| ১১.২ | ইন্টারাপ্ট সার্ভিস রুটিন | ১৬৮ |
| ১১.৩ | ইন্টারাপ্ট অগ্রগণ্যতা ও ভেক্টর অবস্থান উল্লেখ করণ | ১৬৮ |
| ১১.৪ | ইন্টারাপ্ট এনাবল রেজিস্টার IE বর্ণনা | ১৬৯ |
| ১১.৫ | ইন্টারাপ্ট কার্যকর ও অকার্যকর করার প্রক্রিয়া বর্ণনা | ১৭০ |
| ১১.৬ | একটি ইন্টারাপ্ট সম্পাদনের ধাপসমূহ উল্লেখ করণ | ১৭০ |
| ১১.৭ | ইন্টারাপ্টের সময় রেজিস্টার সুরক্ষাকরণ বর্ণনা | ১৭১ |
| ১১.৮ | বাহ্যিক হার্ডওয়্যার, টাইমার ও সিরিয়াল কমিউনিকেশন ইন্টারাপ্ট বর্ণনা | ১৭১ |
| ১১.৯ | ইন্টারাপ্টের সাধারণ সমস্যা বর্ণনা | ১৭৩ |


## অধ্যায়- ১২ এলসিডি ও কী-বোর্ড ইনটারফেসিং


| | | |
|---|---|---|
| ১২.১ | এলসিডি এর পিন ডায়াগ্রাম বর্ণনা | ১৮২ |
| ১২.২ | ইনস্ট্রাকশন রেজিস্টার, ডাটা রেজিস্টার ও বিজি ফ্ল্যাগ বর্ণনা | ১৮৩ |
| ১২.৩ | এলসিডি কমান্ড তালিকা | ১৮৩ |
| ১২.৪ | LCD- তে ডাটা প্রদর্শনের প্রোগ্রাম | ১৮৪ |
| ১২.৫ | একটি মেট্রিক্স কী-বোর্ডের গঠন বর্ণনা | ১৮৫ |
| ১২.৬ | একটি কী চাপ শনাক্তকরণ ও চিহ্নিতকরণের ধাপ ব্যাখ্যা | ১৮৭ |
| ১২.৭ | কোন পোর্টে কী চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম | ১৮৮ |


## অধ্যায়- ১৩ বাস্তব জগতের সাথে ইন্টারফেসিং


| | | |
|---|---|---|
| ১৩.১ | ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং বর্ণনা | ১৯৭ |
| ১৩.২ | 8051 এর সাথে স্টেপার মোটরের ইন্টারফেসিং | ২০১ |
| ১৩.৩ | 8051 এর এডিসি চিপ্স এর ইন্টারফেসিং বর্ণনা | ২০৩ |
| ১৩.৪ | 8051 এর সাথে ডিএসি চিপ্স এর ইন্টারফেসিং | ২০৪ |
| ১৩.৫ | 8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা | ২০৫ |
| ১৩.৬ | 8051 এর সাথে বাহ্যিক মেমোরির ব্যবহার | ২০৭ |
| ১৩.৭ | ডট ম্যাট্রিক্স ডিসপ্লে এর সাথে 8051 এর ইন্টারফেসিং | ২১০ |
| ১৩.৮ | ডট মেট্রিক ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং বর্ণনা | ২১১ |

ব্যবহারিক..... ২২৩-২৩৪

সাজেশন্স..... ২৩৫-২৪১

বিগত বছরের বোর্ড পরীক্ষার প্রশ্ন ও উত্তর ..... ২৪২-২৪৮

# অধ্যায় ১ মাইক্রোকন্ট্রোলারের মৌলিক ধারণা

## Understand the Fundamentals of Micro Controller

### ১.০ ভূমিকা

**(Introduction)**

বর্তমান যুগ কম্পিউটারের যুগ। এই কম্পিউটার মানুষের জীবনে এনে দিয়েছে এক বৈপ্লবিক পরিবর্তন। এই Computer দ্বারা অনেক কাজ করা যায়। যেমন : Calculation, Data Storage, Networking, ইন্টারনেট ব্রাউজিং, প্রোগ্রামিং, প্রিন্টিং, অ্যানিমেশন তৈরি, গ্রাফিক্স ডিজাইন, রোবট কন্ট্রোল, আবহাওয়ার পূর্বাভাস ইত্যাদি। তবে প্রত্যেকটি কাজের পিছনে রয়েছে বিভিন্ন ধরনের হার্ডওয়্যার ও সফ্টওয়্যার। হার্ডওয়্যারগুলো বিভিন্ন ধরনের প্রসেসর, চিপ্স, কন্ট্রোলিং ইউনিট, ইন্টিগ্রেটেড সার্কিট বা IC ইত্যাদির মাধ্যমে তৈরি। বর্তমানে বিভিন্ন ধরনের ছোট ছোট একক কন্ট্রোল ইউনিট ডিভাইস তৈরি হয়েছে যেমন : রিমোট কন্ট্রোল, যার মাধ্যমে দূর থেকে নির্দিষ্ট কাজকে বা প্রক্রিয়াকে নিয়ন্ত্রণ করা সম্ভব। ছোট ছোট কাজ করার জন্য মাইক্রোপ্রসেসর মাইক্রোকন্ট্রোলার ব্যবহার করা হয়ে থাকে। এদেরকে একটি স্বয়ংসম্পূর্ণ ডিভাইস বলা হয়ে থাকে। আবার কিছু কিছু ডিভাইস আছে যে শুধুমাত্র একটি কাজ করতে পারে, যেমন– প্রিন্টার। যে সিস্টেম শুধুমাত্র নির্দিষ্ট একটি কাজ করতে সক্ষম, তাকে ইমবেডেড সিস্টেম (Embedded System) বলে। এ অধ্যায় হতে শিক্ষার্থীরা নিচের বিষয় সম্পর্কে জানতে পারবে।

(১) ইমবেডেড সিস্টেম এবং মাইক্রোকন্ট্রলারের সম্পর্কে জানতে পারবে।
(২) Von Neumann এবং Harvard আর্কিটেকচারের সম্পর্কে জানতে পারবে।
(৩) মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসর সম্পর্কে ধারণা লাভ করবে।
(৪) মাইক্রোকন্ট্রোলারের প্রয়োগক্ষেত্র জানতে পারবে।
(৫) মাইক্রোকন্ট্রোলারের নির্বাচনসমূহের শর্ত সম্পর্কে জানতে পারবে।

### ১.১ মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের সংজ্ঞা

**(Define Microcontroller and Microprocessor)**

#### ১.১.১ মাইক্রোকন্ট্রোলার

**(Microcontroller)**

মাইক্রোকন্ট্রোলার হলো একটি ইন্টিগ্রেটেড সার্কিট (Integrated circuit) বা IC, যার মধ্যে একত্রে প্রসেসর (Processor), RAM (র‍্যাম), রম (ROM), I/O Port এবং টাইমার (Timer) যুক্ত থাকে। এটি নিজে নিজেই একটি স্বয়ংসম্পূর্ণ ডিভাইস। এটি মাইক্রোকম্পিউটারের চেয়ে সহজে ও দ্রুত কাজ করতে পারে। মাইক্রোকন্ট্রোলারকে আবার সিঙ্গেল চিপ মাইক্রোকন্ট্রোলার অথবা ইমবেডেড কন্ট্রোলারও বলা হয়ে থাকে। এটি দামে কম এবং ওজনে হালকা। মাইক্রোকন্ট্রোলারের উদাহরণ হলো– TV Remote Control ট্রাফিক কন্ট্রোল ইউনিট ইত্যাদি।

নিচে মাইক্রোকন্ট্রোলারের সাধারণ একটি Block দেখানো হলো :

| | | |
|---|---|---|
| CPU | RAM | ROM |
| I/O | Timer | Sereal Com Port |

চিত্র-১.১ : মাইক্রোকন্ট্রোলার

**নিচে মাইক্রোকন্ট্রোলার বৈশিষ্ট্য দেয়া হলো :**

(ক) এটি একটি হাইলি ইন্টিগ্রেটেড চিপ।

(খ) মাইক্রোপ্রসেসর, মেমোরি, I/O বিল্ট ইন থাকে।

(খ) আকারে ছোট, ওজন কম ও হালকা হয়ে থাকে।

(ঘ) হার্ডওয়্যার ও সফটওয়্যার ইন্টারফেস বিশিষ্ট।

(ঙ) দ্রুত সিস্টেম উন্নয়ন করা যায়।

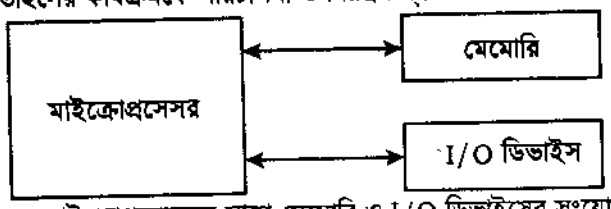
(চ) ডিজিটাল I/O ব্যবহৃত হয়।

(ছ) এটি একটি সিঙ্গেল চিপ বা I/C।

### ১.১.২ মাইক্রোপ্রসেসর

**(Microprocessor)**

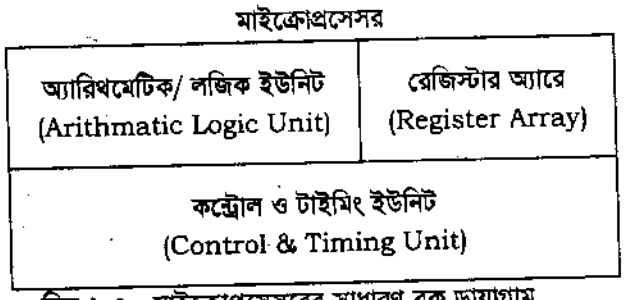
মাইক্রোপ্রসেসর একটি প্রোগ্রামেবল ডিভাইস। মাইক্রোপ্রসেসরে বিভিন্ন ধরনের ইনস্ট্রাকশন প্রদান করা যায় এবং এর ক্ষমতা অনুযায়ী বিভিন্ন ধরনের কাজ করানো যায়। মাইক্রোপ্রসেসর ডিজাইনারগণ বিভিন্ন বাইনারি ইনস্ট্রাকশন বুঝার জন্য এবং ঐ ইনস্ট্রাকশনকে এক্সিকিউট করার জন্য বিভিন্ন ধরনের মাইক্রোপ্রসেসর ডিজাইন করে থাকেন। ডিজাইনকৃত মাইক্রোপ্রসেসরে প্রয়োজনীয় সংখ্যক লজিক সার্কিট বিদ্যমান থাকে আধুনিক কম্পিউটার সিস্টেমে মাইক্রোপ্রসেসর হচ্ছে কম্পিউটারের মাল্টিপারপাস প্রোগ্রামেবল ইন্টিগ্রেটেড ডিভাইস, যা ইনপুটকৃত ডাটা প্রসেসিং বা কম্পিউটিং এবং সিদ্ধান্ত গ্রহণের কাজ করতে পারে। এটা বিভিন্ন ধরনের I/O ডিভাইস এবং মেমোরিকে নিয়ন্ত্রণ ও ডাটাকে সাময়িকভাবে সংরক্ষণের কাজও করে থাকে। এটা মেমোরি নামক স্টোরেজ ডিভাইস হতে বাইনারি নির্দেশ পড়তে পারে, বাইনারি ডাটা গ্রহণ করতে পারে এবং নির্দেশ অনুযায়ী বাইনারি ডাটাকে প্রসেস করে ফলাফল প্রদান করতে পারে। মাইক্রোপ্রসেসরের সাথে প্রধানত I/O ডিভাইস এবং মেমোরির নিবিড় সম্পর্ক থাকে। এটা মেমোরি এবং I/O ডিভাইসের কার্যক্রমকে পরিচালনা ও নিয়ন্ত্রণ করে থাকে।




চিত্র-১.২ : মাইক্রোপ্রসেসরের সাথে মেমোরি ও I/O ডিভাইসের সংযোগ

মাইক্রোপ্রসেসর একটি নির্দিষ্ট সংখ্যক ইনস্ট্রাকশন বুঝতে পারে এবং সে অনুযায়ী কাজ করতে পারে। ইনস্ট্রাকশনসমূহ বাইনারি সংখ্যা 0 এবং 1 এর সমন্বয়। প্রোগ্রামারগণ একটি নির্দিষ্ট কাজের জন্য প্রয়োজনীয় ইনস্ট্রাকশন নির্বাচন করেন এবং প্রোগ্রাম এক্সিকিউশনের ধারাবাহিকতা নির্ধারণ করে। ইনস্ট্রাকশনসমূহ মেমোরি নামক স্টোরেজ ডিভাইসে সংরক্ষিত থাকতে পারে, যা মাইক্রোপ্রসেসর পড়তে পারে এবং সে অনুযায়ী কাজ করতে পারে।

মাইক্রোপ্রসেসরকে মূলত তিনটি ইউনিটে ভাগ করা যায়। ইউনিট তিনটি হচ্ছে– অ্যারিথমেটিক/লজিক ইউনিট (ALU), রেজিস্টার অ্যারে এবং কন্ট্রোল ও টাইমিং ইউনিট।




চিত্র-১.৩ : মাইক্রোপ্রসেসরের সাধারণ ব্লক ডায়াগ্রাম

**মাইক্রোপ্রসেসরের কাজ (Function of Microprocessor) :**

(১) মাইক্রোপ্রসেসর ALU এর মাধ্যমে গাণিতিক ও যৌক্তিক ডাটাকে প্রসেস করে।

(২) এটি কন্ট্রোল ইউনিটের মাধ্যমে মাইক্রোকম্পিউটার সিস্টেমে ব্যবহৃত মেমোরি ও I/O ডিভাইসকে কন্ট্রোল করে এবং টাইমিং সমন্বয় করে।

(৩) এটি রেজিস্টার স্ট্রাকচারের মাধ্যমে ALU এর প্রসেসিং কাজে ব্যবহৃত ডাটা ও ফলাফলকে সাময়িকভাবে ধারণ করে

(৪) এটি নির্দিষ্ট সংখ্যক ইনস্ট্রাকশন সেটকে বুঝতে পারে এবং সে অনুযায়ী কাজ করতে পারে।

(৫) এটি অ্যাড্রেস বাসের মাধ্যমে বিভিন্ন মেমোরি ও I/O ডিভাইসকে সিলেক্ট বা অ্যাড্রেসিং করে এবং ডাটা বাসের মাধ্যমে তাদের সাথে ডাটা লেনদেন করে।

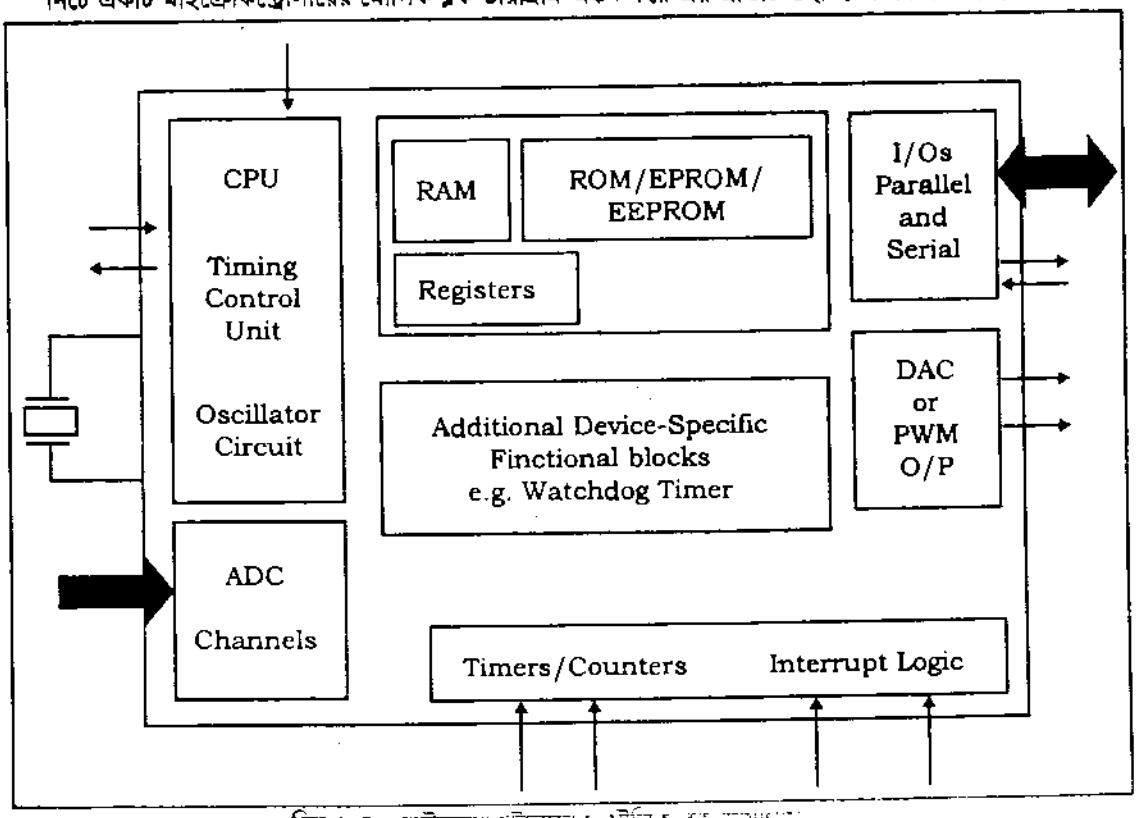
(৬) এটি বিভিন্ন পেরিফেরাল ডিভাইসের রিকোয়েস্ট অনুযায়ী বিভিন্ন কার্য সম্পন্ন করে।

(৭) এটি মেমোরি থেকে ইনস্ট্রাকশন গ্রহণ করে এবং সে অনুযায়ী কাজ করে।

## ১.২ মাইক্রোকন্ট্রোলারের মৌলিক ব্লক ডায়াগ্রামের তালিকা (List the Basic Building Blocks of a Microcontroller)

**মাইক্রোকন্ট্রোলার (Microcontroller) :** মাইক্রোকন্ট্রোলার হলো একটি ইন্টিগ্রেটেড সার্কিট বা IC যার মধ্যে একত্রে প্রসেসর, র‍্যাম (RAM), রম (ROM), I/O Port এবং টাইমার যুক্ত থাকে। এটি একটি স্বয়ংসম্পূর্ণ ডিভাইস।

নিচে একটি মাইক্রোকন্ট্রোলারের মৌলিক ব্লক ডায়াগ্রাম অঙ্কন করে এর প্রতিটি উইনিটের বর্ণনা করা হলো :




চিত্র-১.৪ : মাইক্রোকন্ট্রোলারের মৌলিক ব্লক ডায়াগ্রাম

উপরোক্ত চিত্র হতে দেখা যায়, একটি মাইক্রোকন্ট্রোলার নিচের ইউনিটগুলো নিয়ে গঠিত। যথা :

(1) Central Processing Unit (CPU);

(2) মেমোরি (Memory);

(3) ইনপুট/আউটপুট পোর্ট বা I/O Port;

(4) সিরিয়াল পোর্ট (Serial Port);
(5) টাইমিং এবং কন্ট্রোল ইউনিট (Timing and Control Unit);
(6) ADC এবং DAC;
(7) ইন্টারাপ্ট লজিক (Interrupt Logic);
(8) টাইমার এবং কাউন্টার (Timer and counter);
(9) অসিলেটর সার্কিট (Oscillator circuit);
(10) এডিশনাল সার্কিট (Additional circuit).

নিচে গুরুত্বপূর্ণ ইউনিটগুলোর বর্ণনা পয়েন্ট আকারে দেওয়া হলো :

**(১) সেন্ট্রাল প্রসেসিং ইউনিট বা সিপিইউ (Central Processing Unit) :**

(1) সকল কার্যাদি এই ইউনিটের মাধ্যমে সম্পাদিত হয়।
(2) সকল তথ্য বা ডাটা ইনপুট ডিভাইসের মাধ্যমে গ্রহণ করে, তা CPU তে প্রক্রিয়াকরণের জন্য প্রেরণ করা হয়।
(3) CPU ডাটা প্রক্রিয়াকরণ শেষে প্রয়োজনীয় আউটপুট তৈরি করে।
(4) এতে গাণিতিক, যৌক্তিক কাজ এবং কন্ট্রোল প্রক্রিয়া সম্পূর্ণ হয়ে থাকে।

**(২) মেমোরি (Memory) :**

(1) মাইক্রোকন্ট্রোলারে তিন ধরনের মেমোরি ব্যবহৃত হয়। যেমন : Read Only Memory (ROM), Random Access Memory (RAM), Electrically Programmable Read Only Memory (EPPROM)।
(2) সব ধরনের ডাটা বা তথ্য Memory তে সংরক্ষিত থাকে।
(3) এখানে ROM এর মধ্যে নির্দিষ্ট কাজের Program স্থায়ীভাবে Install করা থাকে, যা শুধুমাত্র Read করা যায়।

**(৩) I/O পোর্ট (I/O Port) :**

(1) মাইক্রোকন্ট্রোলারে 2 ধরনের পোর্ট থাকে, একটি হলো ইনপুট পোর্ট এবং অপরটি হলো আউটপুট পোর্ট।
(2) Input পোর্টের মাধ্যমে প্রয়োজনীয় তথ্য ইনপুট ডিভাইসে প্রদান করা হয় এবং Output পোর্টের মাধ্যমে প্রক্রিয়াকরণের পর প্রাপ্ত ফলাফলকে আউটপুট ডিভাইসে প্রদান করা হয়।

**(৪) টাইমিং এবং কন্ট্রোল ইউনিট (Timing and Control Unit) :**

(1) এই ইউনিটের কাজ হলো মাইক্রোকন্ট্রোলারের অপারেশনের সাথে ক্লকের সমন্বয় সাধন করা।
(2) এই ইউনিট CPU এবং বিভিন্ন পেরিফেরাল ডিভাইসের সাথে এবং মেমোরির সাথে যোগাযোগের জন্য প্রয়োজনীয় কন্ট্রোল Signal তৈরি করে।

**(৫) ইন্টারাপ্ট লজিক ইউনিট (Interrupt Logic Unit) :**

(1) মাইক্রোকন্ট্রোলারে বিভিন্ন ধরনের Interrupt থাকে।
(2) কোন ইনস্ট্রাকশন আগে কাজ করবে বা কোন ইনস্ট্রাকশন পরে কাজ করবে তা এই ইউনিটের মাধ্যমে নির্ধারণ হয়।
(3) 8051 মাইক্রোকন্ট্রোলারে 6 টি ইন্টারাপ্ট লজিক রয়েছে।

**(৬) ADC এবং DAC :**

(1) ADC মানে হলো Analog to Digital Converter এবং DAC মানে হলো Digital to Analog Converter.
(2) এই ইউনিটগুলোর মাধ্যমে প্রয়োজনে Analog Signal কে Digital Signal-এ এবং Digital Signal কে Analog Signal এ রূপান্তর করা হয়।

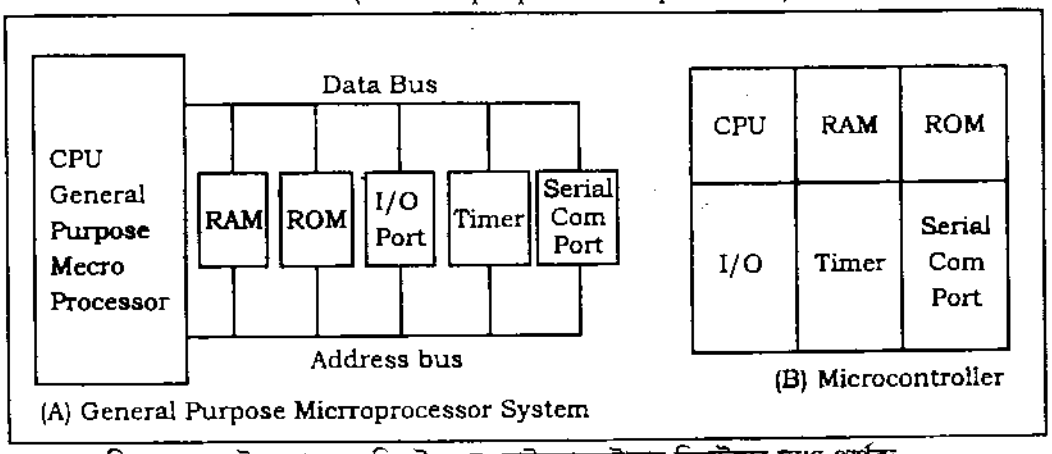
**(৭) অসিলেটর সার্কিট (Oscillator Circuit) :**

(1) Oscillation frequency তৈরি করতে এটি ব্যবহৃত হয়।
(2) মাইক্রোকন্ট্রোলারে অসিলেটর হিসেবে সাধারণত কোয়ার্টজ ক্রিস্টাল অসিলেটর ব্যবহৃত হয়।

## ১.৩ মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের মধ্যে পার্থক্য (Compare Microprocessor and Microcontroller)

প্রকৃতপক্ষে মাইক্রোপ্রসেসর বলতে জেনারেল পারপাস মাইক্রোপ্রসেসরকে (General purpose microprocessor) বুঝায়, অর্থাৎ 8086, 80286, 80386, 80486 এবং পেন্টিয়াম অথবা মটোরোলা ফ্যামিলি (Motorola's family)।

এ মাইক্রোপ্রসেসরগুলোর মধ্যে কোন RAM, ROM এবং I/O Port যুক্ত থাকে না। আর এ জন্য এটিকে জেনারেল পারপাস মাইক্রোপ্রসেসর (General purpose microprocessor) বলে।



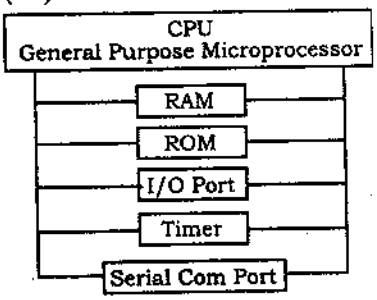
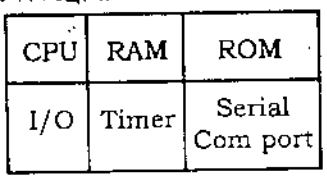

চিত্র-১.৫ : মাইক্রোপ্রসেসর সিস্টেম এবং মাইক্রোকন্ট্রোলার সিস্টেমের মধ্যে পার্থক্য

একজন সিস্টেম ডিজাইনার যখন একটি মাইক্রোপ্রসেসরকে নিয়ে বাস্তবে কাজ করতে চায়, তখন এ মাইক্রোপ্রসেসরের সাথে অবশ্যই RAM, ROM, I/O Port, Timer ইত্যাদি Unit-গুলো যুক্ত করতে হবে, যেমন– একটি কম্পিউটার। একটি কম্পিউটারের মধ্যে ভিন্নভাবে একটি মাইক্রোপ্রসেসর, একটি ROM, RAM, I/O Ports এবং Timer যুক্ত থাকে। তাই একে ইমবেডেড (Embeded) কন্ট্রোলারও বলা হয়। যেহেতু মাইক্রোকন্ট্রোলারের মধ্যে সবগুলো Unit embeded অবস্থায় একটি সিঙ্গেল চিপ হিসেবে থাকে; তাই এটি দামে সস্তা, হালকা এবং প্রয়োজনমত ব্যবহার করা যায়।

একটি ট্রাফিক Control-এর উদাহরণ দেয়া হলো। একটি ট্রাফিক control-এ কোনো গণনা কার্যের প্রয়োজন নেই। এটি শুধু তার নির্দিষ্ট কাজ সম্পন্ন করে। ফলে, এটি এক ধরনের Microcontroller দিয়ে তৈরি।

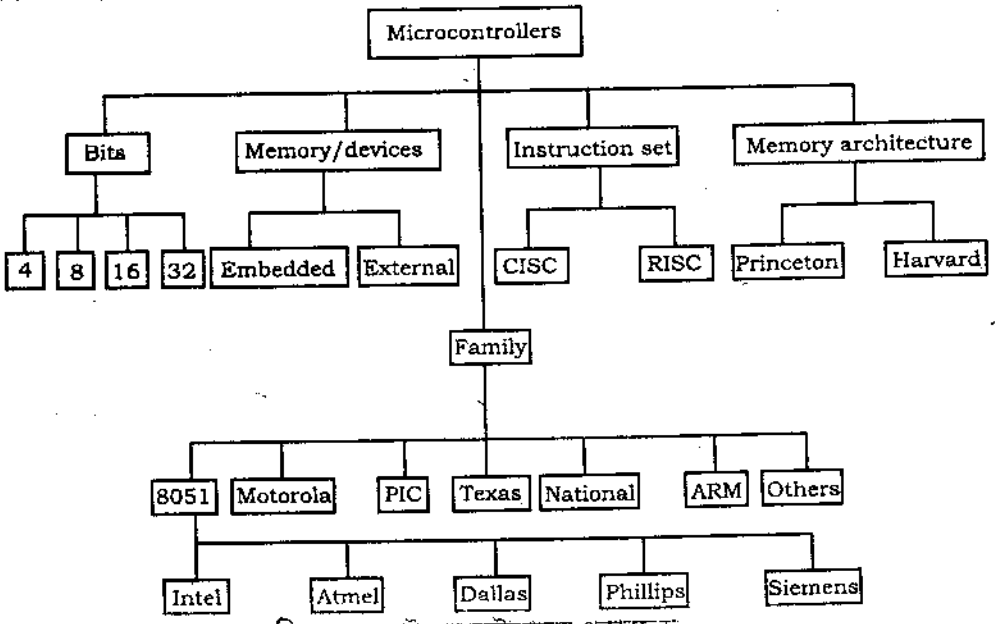
মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের মধ্যে তুলনামূলক পার্থক্য নিচে তুলে ধরা হলো :

| মাইক্রোপ্রসেসর | মাইক্রোকন্ট্রোলার |
|---|---|
| (১) মাইক্রোপ্রসেসরে বাহ্যিকভাবে র‍্যাম, রম, আই/ও, টাইমার সংযোগ দিতে হয়। | (১) মাইক্রোকন্ট্রোলারে র‍্যাম, রম, আই/ও টাইমার বিল্ট ইন অবস্থায় থাকে। |
| (২) মাইক্রোকম্পিউটারের সকল অংশের জন্য কন্ট্রোল সিগনাল এবং টাইমিং পালস এর ব্যবস্থা করে। | (২) মাইক্রোকন্ট্রোলার হলো মাইক্রোপ্রসেসরের সাথে সংযুক্ত অতিরিক্ত ডিভাইস যাতে র‍্যাম, রম টাইমার বিল্ট ইন থাকে। |
| (৩) মাইক্রোপ্রসেসরে অধিক পরিমাণে অপকোড এবং অল্প পরিমাণ বিট হ্যান্ডলিং ইনস্ট্রাকশন থাকে। | (৩) মাইক্রোকন্ট্রোলারে অল্প পরিমাণ অপকোড এবং অধিক পরিমাণ বিট হ্যান্ডলিং ইনস্ট্রাকশন থাকে। |
| (৪) মাইক্রোকম্পিউটারের প্রাণশক্তি হলো মাইক্রোপ্রসেসর। | (৪) মাইক্রোকম্পিউটার পদ্ধতিতে কন্ট্রোল কাজে ব্যবহৃত মাইক্রোপ্রসেসরের সাথে অতিরিক্ত চিপই হলো মাইক্রোকন্ট্রোলার। |

| | |
|---|---|
| (৫) মাইক্রোপ্রসেসর হলো স্বয়ংসম্পূর্ণ ডিভাইস। | (৫) মাইক্রোকন্ট্রোলার স্বয়ংসম্পূর্ণ ডিভাইস নয়। |
| (৬) এর সাথে সিস্টেমের অন্যান্য পেরিফেরাল যুক্ত হয়। | (৬) এটি সিস্টেমের সাথে বহিঃস্থভাবে সরাসরি যুক্ত থাকে। |
| (৭) সাধারণত হায়ার ইন্ড ইন্ডাসট্রিয়াল প্রজেক্টে মাইক্রোপ্রসেসর ব্যবহৃত হয়। | (৭) লোয়ার ইন্ড ইন্ডাসট্রিয়াল প্রজেক্টে মাইক্রোকন্ট্রোলার ব্যবহৃত হয়। |
| (৮) একে General Purpose mieroprocessor ও বলে। | (৮) একে Embedded controller ও বলে। |
| (৯) মাইক্রোপ্রসেসর উদাহরণ : 8085, 8086, 80380, Pentium, core i, dual core ইত্যাদি। | (৯) মাইক্রোকন্ট্রোলারের উদাহরণ : TV Remote control, Traffic control ইত্যাদি। |
| (১০) মাইক্রোপ্রসেসর সিস্টেমের block চিত্র :   | মাইক্রোকন্ট্রোলার সিস্টেমের block চিত্র :   |
| (১১) এর দাম বেশি ও ওজনে ভারি। | (১১) এর দাম সস্তা ও ওজনে হালকা। |

## ১.৪ মাইক্রোকন্ট্রোলারের প্রকারভেদ উল্লেখ করণ (Mention the Types of Microcontroller)

মেমোরি গঠন, প্রসেসিং ওয়ার্ডের আকার, মেমোরি ডিভাইস ও নির্দেশনা সেটের উপর ভিত্তি করে মাইক্রোকন্ট্রোলারকে নিচের ছক মোতাবেক বিভক্ত করা যায় :




চিত্র-১.৬ : মাইক্রোকন্ট্রোলারের প্রকারভেদ

(ক) প্রসেসিং ওয়ার্ডের আকারের উপর ভিত্তি করে–
(১) 4-বিট মাইক্রোকন্ট্রোলার,
(২) 8-বিট মাইক্রোকন্ট্রোলার,
(৩) 16-বিট মাইক্রোকন্ট্রোলার,
(৪) 32-বিট মাইক্রোকন্ট্রোলার,

(খ) মেমোরি ডিভাইসের উপর ভিত্তি করে–
(১) সন্নিহিত মেমোরি মাইক্রোকন্ট্রোলার (Embedded memory microcontroller)
(২) বাহ্যিক মেমোরি মাইক্রোকন্ট্রোলার (External memory microcontroller)

(গ) ইনস্ট্রাকশন সেটের উপর ভিত্তি করে–
(১) Complex Instruction Set Computer (CISC)
(২) Reduced Instruction Set Computer (RISC)

(ঘ) মেমোরি গঠনের উপর ভিত্তি করে–
(১) Von-Neumann (Princeton) architecture microcontroller
(২) Harvard architecture microcontroller.

## ১.৫ হার্ভার্ড ও ভননিউম্যান আর্কিটেকচার বর্ণনা

## (Describe the Harvard and Von-Neumann Architecture)

**মাইক্রোকন্ট্রোলার (Microcontroller) :**

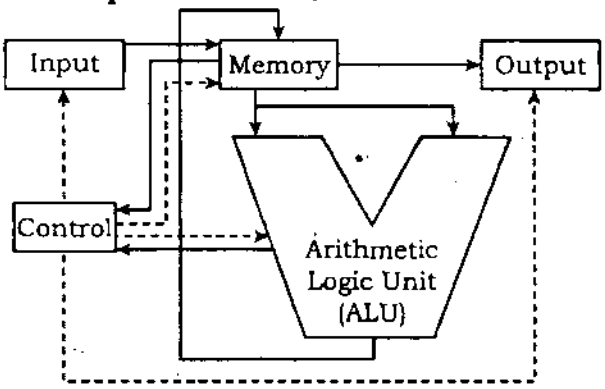
মাইক্রোকন্ট্রোলার হচ্ছে একটি সিঙ্গেল চিপ বা IC, যার মধ্যে একত্রে Processor, RAM, ROM, I/O Port এবং Timer যুক্ত থাকে। এটি নিজেই একটি স্বয়ংসম্পূর্ণ ডিভাইস। এটি মাইক্রোকম্পিউটারের চেয়ে সহজে ও দ্রুত কাজ করতে পারে। একে আবার Embedded কন্ট্রোলারও বলা হয়। প্রোগ্রাম মেমোরি ও ডাটা মেমোরি একই সাথে থাকা বা ভিন্ন ভিন্নভাবে থাকার উপর নির্ভর করে 2টি আর্কিটেকচার আছে। যথা–

১। Von Neumann architecture

২। Harvard architecture.

**Von Newnan architecture :** প্রোগ্রাম মেমোরি ও ডাটা মেমোরি একই মেমোরি ব্যবহার করলে সেই আর্কিটেকচারকে, Von Neuman আর্কিটেকচার বলে।

নিচে **Von Neumann Computer** সিস্টেম এর ব্লক চিত্র অঙ্কন করে বর্ণনা করা হলো :




-------→ Control
———→ Data/Instruciton

চিত্র-১.৭ : Von Neumann Architecture

**মেমোরি (Memory) :** যে ডিভাইস Data ও Program-সমূহ নির্দিষ্ট Binary Information হিসেবে ধারণ করে, তাকে Memory বলে। এটা CPU-এর নির্দেশনা অনুযায়ী Data সমূহ CPU তে পাঠায় এবং প্রসেস শেষে পুনরায় ফলাফল ধারণ করে। এর দুইটি অংশ, যথা : (i) MAR, (ii) MDR

**প্রসেসিং ইউনিট (Processing Unit) :** CPU এর বিভিন্ন ইন্টার্নাল ও এক্সটার্নাল ডিভাইসসমূহ Data Processing কার্য সম্পন্ন করে। এছাড়া এটা প্রোগ্রাম নির্বাহ করার সময় সাময়িকের জন্য ডাটা এবং এড্রেস ধারণ করতে পারে। এর দুটি অংশ। যথা: ALU, Register.

**ইনপুট (Input) :** Input ডিভাইসের মাধ্যমে ব্যবহারকারী প্রসেসিং-এর জন্য Data ও Program-সমূহ প্রয়োগ করে। Input Device হচ্ছে : Keyboard, Mouse, Scanner, Disk ইত্যাদি।

**আউটপুট (Output) :** Output ডিভাইসের মাধ্যমে ব্যবহারকারী Data ও Program-সমূহ প্রদর্শন করে, যেমন– Monitor, Printer, LED, Disk.

**কন্ট্রোল ইউনিট (Control Unit) :** এটা বিভিন্ন Timing ও Control signal-এর মাধ্যমে Internal ও External ডিভাইসসমূহ Control করে, তথা Computer-এর সকল কার্যক্রম (Operation) নিয়ন্ত্রণ করে। এছাড়া এটা CPU এবং Memory অথবা Peripherals ডিভাইসমূহের মধ্যে Data আদান-প্রদান নিয়ন্ত্রণ করে।

হাঙ্গেরিয়ান গণিতবিদ John Von Neumann প্রথম Stored Program Computer EDVAC আবিষ্কার করেন। নিচে এর প্রধান বৈশিষ্ট্যগুলো দেয়া হলো :

(১) এতে Program ও Data– সমূহ আলাদা আলাদা উচ্চ- গতিসম্পন্ন মেমোরিতে Store করা যেত।

(২) এতে যে কোন Program এর নির্দেশনাগুলোকে সহজেই পরিবর্তন করা যেত।

(৩) এতে ডাটা ও প্রোগ্রামসমূহ বাইনারি ফরমে সংরক্ষণ করা ও প্রসেস করা সম্ভব ছিল।

(৪) এতে একটি দ্রুতগতির Main Memory, যার Capacity 1024 byte বা 1KB এবং অন্যটি কম গতির Secondary memory, যার Capacity 20K Words. এ দুই ধরনের মেমোরি ব্যবহৃত হত।

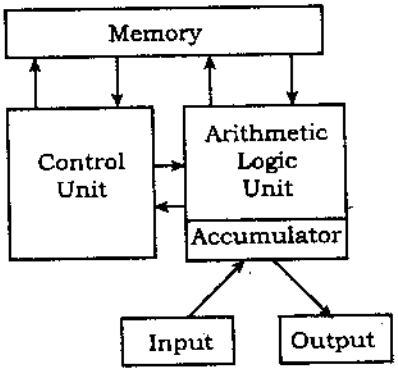
**(১) Von-Neumann আর্কিটেকচারের অসুবিধা :**

(১) প্রধান অসুবিধা হলো– এটা Bottleneck সমস্যা।

(২) ইনস্ট্রাকশন সিরিয়ালি সম্পাদিত হয়।

(৩) প্রত্যেক ইনস্ট্রাকশন প্রক্রিয়া হওয়ার সময় বেশি লাগে কারণ একই মেমোরিতে জমাকৃত ডাটা ও নির্দেশনাকে একটিমাত্র বাস ব্যবহার করে retrieve করতে হয়। মেমোরি অ্যাক্সেস দুই ভাবে হয়– প্রথমে ডাটা অ্যাক্সেস ও পরে নির্দেশনা অথবা প্রথমে নির্দেশনা এবং পরে ডাটা অ্যাক্সেস করা হয়। এতে সিস্টেম ক্যাশ হওয়ার সম্ভাবনা থাকে।

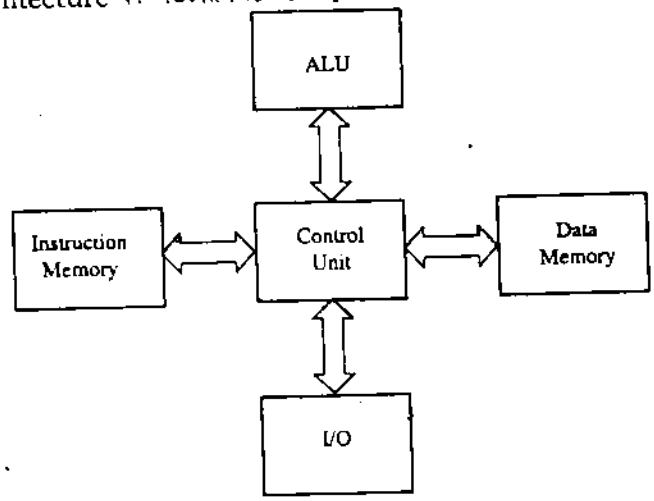
(৪) Fetch-decode-execute-reset সাইকেলে কিছু কিছু রেজিস্টার অলস (idle) থাকে।




চিত্র-১.৮ : Von-Neumann Architecture

(২) **Harvard Architecture কম্পিউটার সিস্টেম এর ব্লক চিত্রসহ বর্ণনা :**

নিচে Harvard Architecture কম্পিউটার সিস্টেমের ব্লক চিত্র অঙ্কন করে বর্ণনা করা হলো :




**চিত্র-১.৯ :** Harvard Architecture

হার্ভার্ড আর্কিটেকচার কম্পিউটার সিস্টেম নিম্নলিখিত অংশ নিয়ে গঠিত।

(i) ALU (ii) Instruction Memory (iii) I/O (Iv) Data Memory (v) Control Unit

নিম্নে এ অংশগুলোর বর্ণনা দেয়া হলো :

**(i) ALU :** এ ALU এর পূর্ণরূপ Arithmetic Logic Unit কম্পিউটারের এ Unit এর মাধ্যমে কম্পিউটার সকল গাণিতিক ও যৌক্তিক কার্য সম্পাদন করে।

**(ii) Instruction Memory :** কম্পিউটার সিস্টেমে Instuction Register বাফার রেজিস্টার হিসেবে কাজ করে। এটি কম্পিউটারের প্রদানকৃত Instruction সমূহকে ধারণ করে।

**(iii) I/O :** কম্পিউটার সিস্টেমে তথ্য ডাটা প্রক্রিয়াকরণের জন্য ডাটাসমূহকে যে ডিভাইসের মাধ্যমে দেয়া হয় তাদের Input Device এবং প্রক্রিয়াকৃত ডাটাকে যে ডিভাইসের মাধ্যমে কম্পিউটার বাইরে প্রদান করে তাদের Output Divice বলে ।

**(iv) Data Memory :** কম্পিউটার সিস্টেমে ডাটা সংরক্ষণের যে ডিভাইস বা স্থান ব্যবহার করা হয় তাকে মেমোরি বা ডাটা মেমোরি বলে। কম্পিউটারে ডাটা প্রক্রিয়াকরণের পর এ ডাটা মেমোরিতে ডাটা সংরক্ষণ করে রাখা হয়।

**(v) Control Unit :** কম্পিউটার সিস্টেমের সকল কার্য এ Control Unit-এর মাধ্যমে নিয়ন্ত্রিত হয়।

**Harvard Architecture এর প্রধান বৈশিষ্ট্যগুলো নিম্নরূপ :**

(১) এই কম্পিউটারটির মাধ্যমে যোগ, বিয়োগ, গুণ, ভাগ এবং ত্রিকোণমিতিক ফাংশন ছাড়াও অনেক জটিল গাণিতিক কাজ করা যেত।

(২) এটি সাধারণত ভালভ বা ভ্যাকুয়াম টিউব দ্বারা তৈরি;

(৩) এর স্মৃতি, চুম্বকীয় ড্রামের ন্যায়;

(৪) এতে কোড ব্যবহার করে প্রোগ্রাম চালানোর ব্যবস্থা ছিল;

(৫) এতে Binary digit 0 এবং 1 এর সমন্বয়ে Program লিখা সম্ভব ছিল।

**হার্ভার্ড আর্কিটেকচারের সুবিধা :**

(১) দক্ষ পাইপলাইনিং– অপারেন্ড ও ইনস্ট্রাকশন যুগপৎ সংঘটিত হতে পারে।

(২) ডাটা বাস ও নির্দেশনা বাস ভিন্ন সমান্তরালে অপারেশন হয় এবং প্রসেসরে গতি বৃদ্ধি পায়।

(৩) ইনস্ট্রাকশন বেশি, হার্ডওয়্যার কম।

**হার্ভার্ড আর্কিটেকচারের অসুবিধা :**

(১) এর ব্যবহার কম।

(২) একে কার্যে পরিণত করা অধিক জটিল।

(৩) এর পিন সংখ্যা বেশি।

### ১.৫.১ Harvard ও Von Neumann আর্কিটেকচারের পার্থক্য
### (Difference Between Harvard and Von Neumann Architecutre)

নিচে Harvard ও Von Neumann আর্কিটেকচারের পার্থক্য তুলে ধরা হলো :

| Von Neumann architecutre | Harvard architecture |
|---|---|
| (ক) এই আর্কিটেকচারে প্রোগ্রাম মেমোরি ও ডাটা মেমোরি একই মেমোরি ব্যবহার করে। | (ক) এই আর্কিটেকচারে প্রোগ্রাম মেমোরি ও ডাটা মেমোরি আলাদা আলাদা মেমোরি ব্যবহার করে। |
| (খ) এটি মাইক্রোকন্ট্রোলারের চিপ ডিজাইনকে Simplify করে। | (খ) এটি মাইক্রোকন্ট্রোলারের চিপ ডিজাইনকে Complex করে। |
| (গ) একই বাস ব্যবহারের জন্য প্রোগ্রাম ও ডাটা একই সাথে কাজ করতে পারে না। | (গ) আলাদা বাস ব্যবহারের জন্য প্রোগ্রাম ও ডাটা একই সাথে কাজ করতে পারে। |
| (ঘ) কাজের গতি তুলনামূলকভাবে অনেক কম। | (ঘ) কাজের গতি তুলনামূলকভাবে অনেক বেশি। |
| (ঙ) পাইপ লাইন প্রসেসিং ব্যবহার করা যায় না। | (ঙ) পাইপ লাইন প্রসেসিং ব্যবহার করা যায়। |
| (চ) Clock Speed কম। | (চ) Clock Speed বেশি। |
| (ছ) কম সংখ্যক Control Signal এর প্রয়োজন হয়। | (ছ) বেশি সংখ্যক Control Signal-এর প্রয়োজন হয়। |
| (জ) এটি একই বাস ব্যবহার করে। | (জ) এটি একাধিক বাস ব্যবহার করে |

### ১.৬ সিআইএসসি ও আরআইএসসি এর তুলনা
### (Compare CISC and RISC)

CISC ও RISC প্রসেসরের পার্থক্য নিচে উল্লেখ করা হলো :

| CISC | RISC |
|---|---|
| ১। অনেক অ্যাড্রেসিং মুড। | ১। স্বল্পসংখ্যক অ্যাড্রেসিং মুড। |
| ২। মেমোরি কাজে শুধু Load/Store নির্দেশনা ব্যবহৃত হয়। | ২। মেমোরি কাজে অনেক নির্দেশনা ব্যবহৃত হয়। |
| ৩। এতে হার্ডওয়্যারে জোর দেয়া হয়। | ৩। সফটওয়্যারে জোর দেওয়া হয়। |
| ৪। জটিলতা মাইক্রো-কোডে। | ৪। জটিলতা কম্পাইলারে। |
| ৫। নির্দেশনার দৈর্ঘ্য পরিবর্তনশীল। | ৫। একই দৈর্ঘ্যের নির্দেশনা। |
| ৬। কোডের আকার ছোট, সাইকেল/সেকেন্ড বেশি। | ৬। কোডের আকার বড়, সাইকেল/সেকেন্ড কম। |
| ৭। মেমোরি– মেমোরি অপারেশন। | ৭। রেজিস্টর- রেজিস্টর অপারেশন। |
| ৮। এর নির্দেশনা জটিল এবং সংখ্যায় অনেক (~ 200-300 instructions)। | ৮। এতে স্বল্পসংখ্যক সরল নির্দেশনা (~ 50) আছে। |
| ৯। নির্দেশনা সংরক্ষণের জন্য কমসংখ্যক ট্রানজিস্টর প্রয়োজন। | ৯। মেমোরি রেজিস্টারের জন্য অধিক ট্রানজিস্টর প্রয়োজন। |
| ১০। একক ক্লক নির্দেশনা। | ১০। বহু ক্লক নির্দেশনা। |
| ১১। প্রোগ্রাম রচনা নমনীয়। | ১১। প্রোগ্রাম রচনা কঠিন। |
| ১২। উৎপাদন খরচ বেশি। | ১২। উৎপাদন খরচ কম। |
| ১৩। ছোট মেমোরি। | ১৩। বড় মেমোরি। |
| ১৪। প্রতি প্রোগ্রামে নির্দেশনা হ্রাস করে। | ১৪। প্রতি ইনস্ট্রাকশনে সাইকেল সংখ্যা হ্রাস করে। |
| ১৫। কাজের গতি মন্থর। | ১৫। কাজের গতি অনেক দ্রুত। |
| ১৬। Pipeline-নেই বা খুব নগণ্য। | ১৬। Pipeline আছে। |

**কম্পিউটারের কর্মসম্পাদন সমীকরণের সাহায্যে পার্থক্য নিরূপণ :**

কম্পিউটারের কর্মসম্পাদন ক্ষমতা প্রকাশ করার জন্য বর্তমানে নিচের সমীকরণটি বহুল ব্যবহৃত হয় :

$$\frac{\text{time}}{\text{program}} = \frac{\text{time}}{\text{cycle}} \times \frac{\text{cycles}}{\text{instruction}} \times \frac{\text{instructions}}{\text{pgrogram}}$$

CISC প্রতি নির্দেশনায় সাইকেল সংখ্যার বদলে প্রতি প্রোগ্রামে ইনস্ট্রাকশন সংখ্যা হ্রাস করে, পক্ষান্তরে RISC তার বিপরীত করে অর্থাৎ এটি প্রতি প্রোগ্রামে নির্দেশনা সংখ্যা কমানোর পরিবর্তে প্রতি ইনস্ট্রাকশনে সাইকেল সংখ্যা হ্রাস করে।

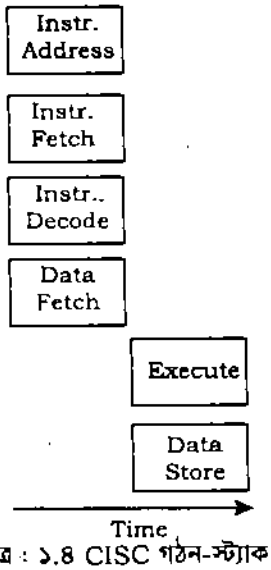
### ১.৬.১ কমপ্লেক্স ইনস্ট্রাকশন সেট কম্পিউটার (সিআইএসসি)

**(Complex Instruction Set Computer (CISC) Architecture Microcontroller)**

যখন কোন মাইক্রোকন্ট্রোলার ইউনিট গাণিতিক ও যৌক্তিক নির্দেশনা সম্পাদন এবং মেমোরি কার্যসম্পাদন-কালীন ALU অপারেশন ও ডাটা ট্রান্সফার অপারেশন কাজে ব্যবহৃত নির্দেশনা অনেক অ্যাড্রেসিং মুড সমর্থন করে তবে ঐ মাইক্রোকন্ট্রোলারকে CISC architecture microcontroller বলে। উদাহরণ : Intel 8096, 8086 ISA.

CISC ডাটা ট্রান্সফার, গাণিতিক, যৌক্তিক ও অন্যান্য অপারেশন বিভিন্নভাবে সম্পাদন করার জন্য খুবই নমনীয়। ফলে প্রোগ্রাম রচনা খুব সহজ হয়। এই প্রসেসরের প্রাথমিক উদ্দেশ্য ছিল মেমোরির পরিমাণ কমানোর জন্য প্রতি প্রোগ্রামের নির্দেশনা সংখ্যা হ্রাস করা এবং একটি একক নির্দেশনায় একাধিক কাজ (multiple operation) করে নির্দেশনার সংখ্যা কমানো হয়। কিন্তু একক নির্দেশনায় বহুবিদ কাজের অসুবিধা- এতে মেমোরি অ্যাক্সেস করতে পারে এমন বিভিন্ন ধরনের অনেক নির্দেশনা প্রয়োজন।

এর নির্দেশনা পরিবর্তনশীল সংখ্যক ক্লক সাইকেলে (multi-clock) সম্পাদিত হয়, এতে নির্দেশনার দৈর্ঘ্যও পরিবর্তনশীল হয়। ফলে নির্দেশনা fetch, decode ও সম্পাদনকালে অনিশ্চিত হয়, এতে নির্দেশনা অধিক জটিল হয় এবং হার্ডওয়্যারকে অধিক জটিলতা মোকাবেলা করতে হয়। সেজন্যেই CISC প্রসেসর সিস্টেমে হার্ডওয়্যারকে বেশি গুরুত্ব (emphasis) দেওয়া হয়।




চিত্র-১.১০ : চিত্র : ১.৪ CISC গঠন-স্ট্যাক ডিজাইন

CISC প্রসেসরের ইনস্ট্রাকশন সেট জটিল ও পরিবর্তনশীল হওয়ায় অনুবাদের জন্য অধিক সময় প্রয়োজন এবং সাথে সাথে অধিক হার্ডওয়্যার প্রয়োজন। সাধারণত পরিবর্তনশীল দৈর্ঘ্যের নির্দেশনাকে মোকাবেলা করার জন্য মাইক্রো-প্রোগ্রাম নিয়ন্ত্রণ বাস্তবায়ন করা হয়।

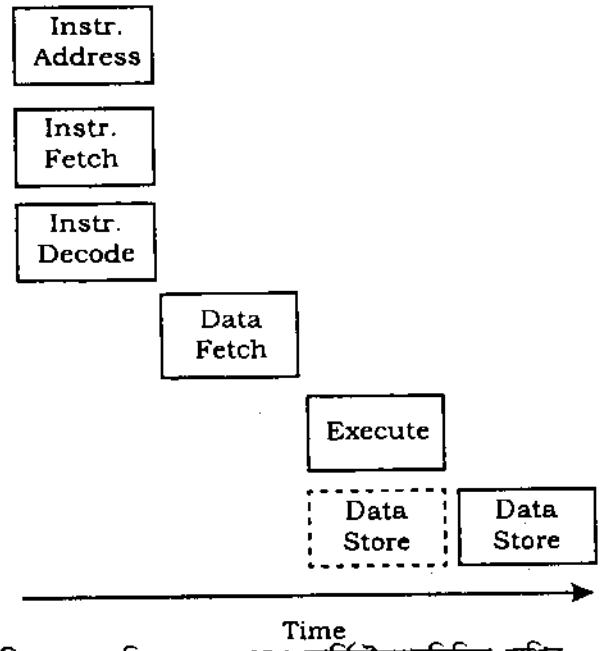
CISC এর অপর একটি অন্যতম উদ্দেশ্য হলো কোন জটিল কাজকে যতটুকু সম্ভব কয়েকটি অ্যাসেম্বলি লাইনে সম্পূর্ণ করা। এটি সম্ভব যদি এমন একটি প্রসেসর হার্ডওয়্যার তৈরি করা যায় যা কাজের একটি ধারাকে বুঝতে ও সম্পাদন করতে পারে। এই বিশেষ কাজের জন্য একটি বিশেষ নির্দেশনা "MULT" নিয়ে এই প্রসেসর তৈরি হয়। যখন এই নির্দেশনা সম্পাদিত হয় তখন এই দুটি ভিন্ন রেজিস্টারে দুটি মান (value) জমা করে, এক্সিকিউশন অংশে অপারেন্ডকে গুণ করে (Multiply) এবং গুণফল নির্দিষ্ট রেজিস্টারে জমা করে। এখানে CISC দুটি সংখ্যার গুণের কাজকে একটিমাত্র নির্দেশনায় শেষ করে। "MULT" একটি জটিল নির্দেশনা ("Complex instruction")। এটি সরাসরি কম্পিউটার মেমোরি ব্যাংকে কাজ করে এবং যে-কোনরূপ loading ও Storing ফাংশনের জন্য সুস্পষ্টভাবে প্রোগ্রামারকে প্রয়োজন হয় না। এটি High level language এর সাথে খুবই সদৃশ। CISC প্রসেসরে RISC প্রসেসরে RISC প্রসেসরে তুলনায় অধিক সংখ্যক অভ্যন্তরীণ নির্দেশনা (~ 200-300 instructions) থাকে, ফলে কাজের বৈচিত্র্য বেশি কিন্তু জটিল নির্দেশনার কারণে এর গতি খুব মন্থর।

### ১.৬.২ রিডিউসড ইনস্ট্রাকশন সেট কম্পিউটার (আরআইএসসি) মাইক্রোকন্ট্রোলার

**(Reduced instruction set computer (RISC) microcontroller)**

যখন কোন মাইক্রোকন্ট্রোলার ইউনিট গাণিতিক ও যৌক্তিক নির্দেশনা সম্পাদন কাজে এবং মেমোরি অ্যাক্সেস-কালীন ALU অপারেশন ও ডাটা ট্রান্সফার অপারেশন নির্দেশনায় (load, store, push ও pop) খুবই স্বল্পসংখ্যক অনেক অ্যাড্রেসিং মুড সমর্থন করে তবে ঐ মাইক্রোকন্ট্রোলারকে RISC architecture microcontroller বলে। অ্যাড্রেসিং মুড কম বিধায় RISC ডাটা ট্রান্সফার, গাণিতিক, যৌক্তিক ও অন্যান্য অপারেশন বিভিন্নভাবে সম্পাদন করার জন্য নমনীয় নয়। উদাহরণ : ARM প্রসেসর পরিবারভিত্তিক MCU, LC3, MIPS, PowerPC (IBM), SPARC (Sun)।

RISC এর মূল ধারণা হলো সমস্ত প্রয়োজনীয় কাজ সম্পাদনে সক্ষম এমন একটি ছোট ইনস্ট্রাকশন সেট (~ 50) হ্রাস প্রতিষ্ঠা করা এবং কয়েকটি সরল ও একই দৈর্ঘ্যের নির্দেশনার মাধ্যমে নির্দেশনার জটিলতা হ্রাস করা হয়। সুস্পষ্টভাবে শুধুমাত্র LOAD ও STORE নির্দেশনার মাধ্যমে মেমোরি কাজ করা হয়। নির্দেশনা থেকে জটিলতা দূর হলেও এই জটিলতা যুক্ত হয় কম্পাইলার ইন্টাপ্রিটারে। CISC প্রসেসরের তুলনায় RISC প্রসেসরের সফটওয়্যারকে অধিক কাজ মোকাবেলা করতে হয়। তাই RISC প্রসেসর সিস্টেমে সফটওয়্যারকে বেশি জোর দেওয়া হয়।



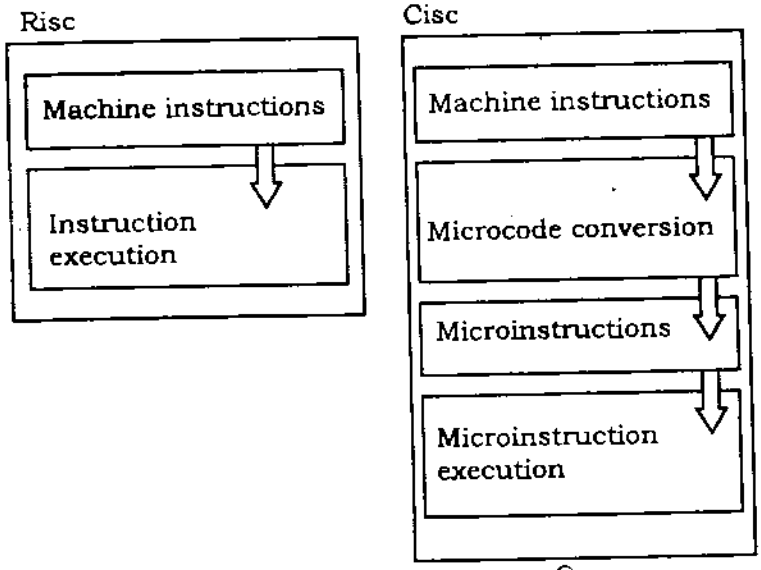

চিত্র-১.১১ : চিত্র : ১.৫ RISC আর্কিটেকচারভিত্তিক মেশিন

এর প্রতিটি নির্দেশনার দৈর্ঘ্য একই এবং প্রতিটি নির্দেশনা একক ক্লক সাইকেলে একটি স্বতন্ত্র হার্ডওয়্যার কন্ট্রোল ব্যবহার করে সম্পাদিত হয়। এতে প্রতিটি কাজকে ছোট ছোট কোডে বিভক্ত করে রচনা করা হয়। এতে কার্যসম্পাদন প্রক্রিয়া সরল ও সহজ হয় বটে কিন্তু প্রোগ্রাম অনেক বড় হয় এবং মেমোরি অনেক বড় প্রয়োজন হয়। ফলে খরচ বৃদ্ধি পায়। কিন্তু যেহেতু প্রতিটি নির্দেশনা মাত্র এক ক্লক সাইকেলই সম্পাদিত হয় তাই এর প্রোগ্রাম সম্পাদনের সময় প্রায় CISC দ্বারা সম্পাদন সময়ের কাছাকাছি। পরন্তু বর্তমানে RAM এর দাম ক্রম হ্রাস এবং কম্পাইলার প্রযুক্তি উন্নত হওয়ার কারণে RISC ও RAM এর ব্যবহার ও সফটওয়্যারের উপর জোর দেওয়া ধীরে ধীরে আদর্শে রূপান্তর হচ্ছে। এর Hardwared কন্ট্রোল ইউনিট কাজের সর্বোচ্চ গতির নিশ্চয়তা দেয়, পরন্তু কার্যসম্পাদন ক্ষমতা বৃদ্ধিকণের জন্য এতে pipelining (এটি এমন একটি ব্যবস্থা যাতে 4/5টি কাজ একত্রে সম্পাদন করা যায়) এর ব্যবস্থা আছে।

এর ALU একটি ছোট ইনস্ট্রাকশন সেট ব্যবহার করে এবং কার্যসম্পাদনের গতি দ্রুত করার জন্য নির্দেশনাগুলোকে প্রোগ্রামে বার বার ব্যবহার করা হয়। নির্দেশনা সম্পাদন দ্রুতগতির হওয়ায় এই ডিজাইন, পরীক্ষণ ও তৈরি কম খরচে করা যায়। এতে কম সংখ্যক সার্কিটের প্রয়োজন। ফলে এর শক্তি অপচয় কম।

এতে অধিক সংখ্যক রেজিস্টার বিদ্যমান। অধিকাংশ কাজই এই রেজিস্টার ব্যবহার করেই সংঘটিত হয়। এতে বহিঃস্থ মেমোরি থেকে ইনস্ট্রাকশন আনয়ন (fetch) খুব কমে গেছে। মেমোরি অপারেশনে LOAD ও STORE নির্দেশনা ব্যবহৃত হয়। এর কার্য সম্পাদন হয় অপারেন্ডসমূহকে রেজিস্টারে বোঝাই (Load) করে এবং কাজের ফলাফল রেজিস্টারে জমা (Store) করে। পরবর্তীতে এটি রেজিস্টার থেকে মেমোরিতে জমা করা হয়।

বাহ্যিক fectch কম থাকায় এবং হার্ডওয়্যার সম্পাদিত নির্দেশনার কারণে CISC এর তুলনায় RISC এর ক্ষমতা অনেক বেশি। আজকাল অধিকাংশ মাইক্রোপ্রসেসর ও মাইক্রোকন্ট্রোলার RISC core এর উপর ভিত্তি করেই তৈরি হয়। এর নির্দেশনা সংখ্যা কম বিধায় সংরক্ষণের জন্য কম ট্রানজিস্টর লাগে ফলে উৎপাদন খরচ অনেক কম।




চিত্র-১.১২ : CISC ও RISC-এর সম্পাদন প্রক্রিয়া

## ১.৭ মাইক্রোকন্ট্রোলারের প্রয়োগক্ষেত্র উল্লেখকরণ (Mention the Field of Microcontroller Applications)

নিচে মাইক্রোকন্ট্রোলারের প্রয়োগক্ষেত্রগুলো উল্লেখ করা হলো :

**বাসাবাড়ি (Home) :**

(১) বিভিন্ন ধরনের অ্যাপ্লায়েন্স
(২) টেলিফোন
(৩) ইন্টারকম
(৪) সিকিউরিটি সিস্টেম
(৫) গ্যারেজ ডোর ওপেনারস
(৬) আনসারিং মেশিন (Answering machines)
(৭) ক্যামেরা
(৮) খেলনা
(৯) টিভি
(১০) ক্যাবল টিভি টিউনার
(১১) ভিসিআর
(১২) রিমোট কন্ট্রোল
(১৩) ভিডিও গেমস
(১৪) হোম কম্পিউটার
(১৫) মিউজিক্যাল ইনস্ট্রুমেন্ট
(১৬) সেলাই মেশিন
(১৭) লাইটিং কন্ট্রোল
(১৮) ফ্যাক্স মেশিন
(১৯) সেলুলার ফোন
(২০) ব্যায়ামের যন্ত্র।

**অফিস (Office) :**

(১) সিকিউরিটি সিস্টেম
(২) পেজিং
(৩) কপিয়ার
(৪) ফ্যাক্স মেশিন
(৫) মাইক্রোওয়েভ
(৬) টেলিফোন
(৭) প্রিন্টার
(৮) কম্পিউটার।

**অটো (Auto) :**

(১) ট্রিপ কম্পিউটার
(২) ইঞ্জিন কন্ট্রোল
(৩) ইনস্ট্রুমেনটেশন
(৪) ট্র্যান্সমিশন কন্ট্রোল
(৫) ইন্টারটেইনমেন্ট
(৬) আবহাওয়া কন্ট্রোল।

## ১.৮ বাণিজ্যিক মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য

## (List the Commercial Microcontrollers With Salient Features)

নিচে বিভিন্ন পরিবারের মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য উল্লেখ করা হলো :

**(১) ইন্টেল 8051 (Intel 8051) মাইক্রোকন্ট্রোলার :** ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 4k byte এর ROM থাকে।
(2) এটি 40 পিনের DIP প্যাকেজ।
(3) CPU একই সময়ে 8 বিট ডাটা নিয়ে কাজ করে।
(4) এটিতে On chip oscillator ও clock সার্কিট থাকে।
(5) এতে 8টি বিটের CPU থাকে।
(6) এতে 6 টি Interrupt সুবিধা থাকে।
(7) 8 বিটের 4 (চার)টি I/O পোর্ট থাকে।
(8) এতে 128 byte এর RAM বিদ্যমান।
(9) এতে সিরিয়াল পোর্ট বিদ্যমান।
(10) এতে 2টি 16 বিটের টাইমার থাকে।

**(২) ইন্টেল 8048 (Intel 8048) মাইক্রোকন্ট্রোলার :** ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 2টি maskable (মাস্কেবল) ইন্টারাপ্ট রয়েছে।
(2) এটি 8 বিটের মাইক্রোকন্ট্রোলার।
(3) এতে 64 byte-এর RAM আছে।
(4) এতে 1k byte এর ROM বিদ্যমান।
(5) এটি 40 পিনের DIP প্যাকেজ।
(6) এতে Timer রয়েছে।
(7) এটি 8 বিটের 3 টি I/O পোর্ট বিদ্যমান।

**(৩) ইন্টেল 8096 (Intel 8096) Microcontroller :** ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 10 bit ADC (Analog to digital converter) বিদ্যামান।
(2) এটি 16 bit-এর মাইক্রোকন্ট্রোলার।
(3) এই মাইক্রোকন্ট্রোলারে 16 বিট বিশিষ্ট্য 2 টি টাইমার বিদ্যমান।
(4) এটি full-duplex Serial Port বিশিষ্ট্য।
(5) এতে 8 বিটের 5 টি ইনপুট/আউটপুট পোর্ট (I/O Port) রয়েছে।
(6) এতে 16 বিট এর CPU রয়েছে।

**(৪) Intel 2920 Microcontroller :** ইন্টেল 2920 মাইক্রোকন্ট্রোলারের কিছু বৈশিষ্ট্য নিচে তুলে ধরা হলো :

(1) এতে Analog ডিমাল্টিপ্লেক্সার বিদ্যমান।
(2) এটি 32 bit এর মাইক্রোকন্ট্রোলার।
(3) এটিতে D/A converter ও বিদ্যমান।
(4) এটি একটি Programmable IC।
(5) এতে A/D converter ও রয়েছে।
(6) এতে Analog মাল্টিপ্লেক্সার বিদ্যমান।
(7) এতে 192 Ward এর EPROM বিদ্যমান।

**(৫) PIC16 F84A মাইক্রোকন্ট্রোলার :** PIC16F84 A মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এটি RISE Architecture বিশিষ্ট।
(2) এটি মাইক্রোচিপ কোম্পানির তৈরি 8 বিটের মাইক্রোকন্ট্রোলার।
(3) এতে 2টি I/O পোর্ট বিদ্যমান।
(4) এতে 35টি ইনস্ট্রাকশন বিদ্যমান।
(5) এর 13টি I/O পিন রয়েছে।
(6) এর Clock frequency 20 MHz পর্যন্ত হয়।
(7) এর প্রায় সকল ইনস্ট্রাকশন 1 মেশিন সাইকেল।
(8) এতে 1 কিলোওয়ার্ড প্রোগ্রাম মেমোরি বিদ্যমান।

**(৬) MSC-51মাইক্রোকন্ট্রোলার :** MSC-51 মাইক্রোকন্ট্রোলারের কিছু বৈশিষ্ট্য উল্লেখ করা হলো :

(1) CHMOS (Complementary high speed metal oxide semiconductor) version এরও মাইক্রোকন্ট্রোলার রয়েছে।
(2) এটি 8 bit এর Microcontroller।
(3) CHMOS version এর μc-এ Idle mode থাকে।
(4) এতে MCS-48 microcontroller এর সাথে Replace করা যায়।
(5) এটি মূলত HMOS (High speed metal oxide semiconductor) Tachnology তে তৈরি।
(6) CHMOS version এর মাইক্রোকন্ট্রোলার হলো Lower Power consmption, higher noise immunity এবং high speed বিশিষ্ট।

ইনটেল কোম্পানি (Intel Company) বা Corporation 1981 সালে প্রথম 8 bit মাইক্রোকন্ট্রোলার তৈরি করে যাকে 8051 নামে অভিহিত করা হয়। 8051 মাইক্রোকন্ট্রোলারের প্রচন্ড জনপ্রিয় (Popular) হবার পর Intel Company অন্যান্য Manufacturers কে 8051 মাইক্রোকন্ট্রোলার তৈরির অনুমতি প্রদান করে। বিভিন্ন ধরনের ম্যানুফ্যাকচারিং (Manufacturer) কোম্পানি মাইক্রোকন্ট্রোলার তৈরি করে থাকে। কোম্পানিগুলো হলো :

(i) ফিলিপস/সিগনিলাইস (Philips/Signelies),
(ii) ইন্টেল (Intel),
(iii) ইনফাইনিওন (Infineon),
(iv) অ্যাটমেল (Atmel),
(v) ডালাস সেমি/ম্যাক্সিম (Dallas semi/maxim) ইত্যাদি।

**ইনটেল কোম্পানি (Intel Company) :** 1981 সালে Intel Corporation 8-bit microcontroller (যা 8051 নামে পরিচিত) বাজারজাত করে। এটি নিচের বৈশিষ্ট্যগুলো ধারণ করে :

(i) 128 bytes RAM
(ii) 4k bytes ROM
(iii) 32টি I/O Pins
(iv) 2টি Timers
(v) 1টি সিরিয়াল পোর্ট
(vi) 6টি Interrupt Source রয়েছে।

8051 মাইক্রোকন্ট্রোলারের প্রচুর পরিমাণ চাহিদার কারণে Intel company অন্যান্য Manufacturer-কে 8051 Microcontroller তৈরিতে উৎসাহ প্রদান করে। ফলে, বিভিন্ন Company বিভিন্ন Flavours-এর Microntroller বাজারে ছাড়ে। এর মধ্যে RAM, ROM, Speed, Packaging দাম ইত্যাদির পার্থক্য আছে।

**ফিলিপস কোম্পানি (Philips Company) :** Philips কোম্পানি 8051 Family-এর Microcontroller তৈরি করে থাকে। এ Microcontroller বেশি কিছু ভিন্ন বৈশিষ্ট্য ধারণ করে, যেমন– (1) A to D converter, (2) D to A converter, (3) extended I/O, (4) OTP and flash.

**ডালাস কোম্পানি (Dallas Company) :** Dallas Semiconductor (Maxim), 8051 Microcontroller বাজারে ছেড়েছে। এখানে Flash ROM ব্যবহৃত হয়েছে।

Flash ROM ব্যবহারের কারণে উক্ত কোম্পানির 8051 microcontroller টি বেশ জনপ্রিয়তা আছে।

**অ্যাটমেল কোম্পানি (Atmel Company) :** অ্যাটমেল কোম্পানি দুই ধরনের মাইক্রোকন্ট্রোলার তৈরি করে থাকে। এক ধরনের মাইক্রোকন্ট্রোলার হলো ROM flash জাতীয়, অন্য ধরনের মাইক্রোকন্ট্রোলার হলো স্পিড ডিপেন্ডেন্ট।

নিচের টেবিলের সাহায্যে বিভিন্ন Manufacturer কর্তৃক তৈরি Microcontroller-এর মধ্যে তুলনা দেখানো হয়েছে :

**টেবিল-1.1 :** Versions of 8051 From Atmel (All ROM Flash)

| Part Number | ROM | RAM | I/O Pins | Timer | Interrupt | $V_{cc}$ | Packaging |
|---|---|---|---|---|---|---|---|
| AT89C51 | 4K | 128 | 32 | 2 | 6 | 5V | 40 |
| AT89LV51 | 4K | 128 | 32 | 2 | 6 | 3V | 40 |
| AT89C1051 | 1K | 64 | 15 | 1 | 3 | 3V | 20 |
| AT89C2051 | 2K | 128 | 15 | 2 | 6 | 3V | 20 |
| AT89C52 | 8K | 128 | 32 | 3 | 8 | 5V | 40 |
| AT89LV52 | 8K | 128 | 32 | 3 | 8 | 3V | 40 |

**Note :** "C" in the part number indicates CMOS :

**টেবিল-1.2 :** Versions of 8051/52 Microcontroller From Dallas Semiconductor (Maxim)

| Part Number | ROM | RAM | I/O pins | Timers | Interrupts | $V_{cc}$ |
|---|---|---|---|---|---|---|
| DS89C420/30 | 16K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS89C440 | 32K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS89C450 | 64K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS5000 | 8K (NVRAM) | 128 | 32 | 2 | 6 | 5V |
| DS80C320 | 0K | 256 | 32 | 3 | 6 | 5V |
| DS87520 | 16K (UVROM) | 256 | 32 | 3 | 6 | 5V |

**টেবিল- 1.3 :** Various Speeds of 8051 From Atmel

| Part Number | Speed | Pins | Packaging | Use |
|---|---|---|---|---|
| AT89C51-12PC | 12 MHz | 40 | DIP plastic | Commercial |
| AT89C51-16PC | 16 MHz | 40 | DIP plastic | Commercial |
| AT89C51-20PC | 20 MHz | 40 | DIP plastic | Commercial |

## ১.৯ মাইক্রোকন্ট্রোলার নির্বাচনে শর্তসমূহ উল্লেখকরণ

**(Mention the Criteria for Choosing a Microcontroller)**

মাইক্রোকন্ট্রোলার এমন একটি ইন্টিগ্রেটেড ডিজাইন, যার মধ্যে প্রসেসর, RAM, ROM, Timer, I/O পোর্ট ইত্যাদি একত্রে থাকে। যার ফলে এর মাধ্যমে নির্দিষ্ট একটি কাজ করা সম্ভব। প্রকৃতপক্ষে, চার প্রকারের 8-বিট মাইক্রোকন্ট্রোলার রয়েছে, যেমন– 6811, 8051, Zilog'sZ8, PIC16X. প্রত্যেক মাইক্রোকন্ট্রোলারের আলাদা ইনস্ট্রাকশন সেট এবং আলাদা রেজিস্টার সেট রয়েছে এবং প্রত্যেকে ভিন্ন ভিন্ন কাজের জন্য তৈরি। একটি প্রোগ্রামের জন্য তৈরি মাইক্রোকন্ট্রোলার দিয়ে অন্য কাজ করা সম্ভব নয়। অর্থাৎ যে কাজের জন্য Microcontroller তৈরি, শুধুমাত্র সেই কাজের জন্য উক্ত Microcontroller ব্যবহার করা যাবে, অন্য কোথাও নয়।

মাইক্রোকন্ট্রোলার নির্বাচনের ক্ষেত্রে গুরুত্বপূর্ণ তিনটি শর্ত রয়েছে। যথা–

(১) যে কাজের জন্য Microcontroller তৈরি হবে, সেই কাজ দক্ষতার সাথে অল্প খরচে হচ্ছে কিনা, তা লক্ষ্য রাখতে হবে।

(২) যে সব কাজের জন্য Microcontroller তৈরি, সে সব কাজের জন্য উপযুক্ত Software, Compilers, Assemblers এবং Debugger আছে কিনা, তা লক্ষ্য রাখতে হবে।

(৩) Microcontroller-এর প্রাচুর্যতা এবং সঠিক উৎস থাকতে হবে।

**নিচে বিস্তারিতভাবে আলোচনা করা হলো :**

(১) Microcontroller নির্বাচনের প্রথম ও প্রধান শর্ত হলো, যে কাজের জন্য Microcontroller তৈরি হবে, সেই কাজ দক্ষতার সাথে হচ্ছে কিনা, তা লক্ষ্য রাখতে হবে। Microcontroller নির্বাচনের ক্ষেত্রে তা কত bit অর্থাৎ 8-bit, 16-bit না 32-bit, তা অবশ্যই খেয়াল রাখতে হবে। তাছাড়াও নিচের বিষয়গুলো খেয়াল রাখতে হবে।

(ক) **গতি (Speed) :** Microcontroller, speed support করছে কিনা, তা লক্ষ্য রাখা।

(খ) **প্যাকেজিং (Packaging) :** প্যাকেজ ফরম্যাট কেমন অর্থাৎ 40-Pin DIP (Deal inline package) নাকি অন্য ফরম্যাট।

(গ) **পাওয়ার কনজাম্পশন (Power Consumption) :** ব্যাটারি পাওয়ার প্রোডাক্টের Power consumption একটি গুরুত্বপূর্ণ বিষয়।

(ঘ) র‍্যাম RAM ও রম ROM-এর জায়গার পরিমাণ।

(ঙ) Input/Output Pin এবং Timer-এর সংখ্যা।

(চ) মাইক্রোকন্ট্রোলারের কয়টি টাইমার Timer আছে ও কী কী কাজ করে তাও জানা অত্যন্ত জরুরি বিষয়।

(ছ) **Time প্রতি Unit খরচ :** প্রতি Unit তৈরির খরচ কত, সেদিকে লক্ষ্য রাখতে হবে।

(২) মাইক্রোকন্ট্রোলার নির্বাচনের দ্বিতীয় শর্ত হলো কত সহজে Microcontroller দিয়ে প্রোডাক্ট প্রস্তুত করা যায়। এছাড়া যে কাজের জন্য Microcontroller তৈরি, সেই কাজের উপযুক্ত Software, Compiler, Assembler এবং Debugger থাকে কিনা, তা লক্ষ্য রাখতে হবে।

(৩) তৃতীয় ও সর্বশেষ মাইক্রোকন্ট্রোলার নির্বাচনের শর্ত হলো বর্তমান ও ভবিষ্যতে উক্ত মাইক্রোকন্ট্রোলারের সুবিধা ও সঠিক উৎস আছে কিনা, তা লক্ষ্য রাখা। অর্থাৎ কোনো Product তৈরির জন্য এমন ধরনের মাইক্রোকন্ট্রোলার নির্বাচন করা হলো, যা ভবিষ্যতে আর পাওয়া যাবে না, তখন উক্ত মাইক্রোকন্ট্রোলার নির্বাচন করা সঠিক হবে না। যেমন, 8-bit মাইক্রোকন্ট্রোলার দিয়ে তৈরি Product হয়তো ভবিষ্যতে আর ব্যবহার হবে না। তাই, তৃতীয় ক্রাইটেরিয়া অবশ্যই গ্রহণযোগ্য এবং লক্ষ্য রাখা।

পলিটেকনিকের সকল বই ডাওনলোড করতে
ভিজিটঃ
www.BDeBooks.Com/polytechnic

# অনুশীলনী-০১

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **মাইক্রোকন্ট্রোলার কাকে বলে?** *[বাকাশিবো : '১০, '১১, '১২, '১২, '১৪, '১৫]*

**উত্তর :** মাইক্রোকন্ট্রোলার একটি সিঙ্গেল চিপ বা IC, যার মধ্যে একত্রে Processor, RAM, ROM, I/O Port এবং Timer যুক্ত থাকে; যথা– TV remote Control, ট্রাফিক কন্ট্রোল ইত্যাদি।

২। **ইমবেডেড সিস্টেম বলতে কী বুঝায়?** *[বাকাশিবো : '১১, '১৪, '১৫]*

**উত্তর :** ইমবেডেড সিস্টেম হলো এমন এক ধরনের সিস্টেম বা মেকানিক্যাল ডিভাইস যা শুধুমাত্র একটি নির্দিষ্ট কাজ করতে সক্ষম।

৩। **Embedded controller কাকে বলে?** *[বাকাশিবো : '১৪R]*

**উত্তর :** কোন বিশেষ কাজের জন্য যে কম্পিউটার বা কন্ট্রোলার কোন সিস্টেমের অভ্যন্তরে সন্নিহিত থাকে, তাকে Embedded controller বলে।

৪। **Microcontroller শব্দের অর্থ কী?** *[বাকাশিবো : '১৪]*

**উত্তর :** ইংরেজি শব্দ "Micro" অর্থ ছোট এবং "Controller" অর্থ কোন অবজেক্ট (object) বা প্রক্রিয়া বা ইভেন্টের (event) নিয়ন্ত্রণকারী। মাইক্রোকন্ট্রোলার কোন প্রসেস বা ডিভাইসকে নিয়ন্ত্রণকারী ছোট আকারের একটি কম্পিউটার।

৫। **কয়েকটি জনপ্রিয় মাইক্রোকন্ট্রোলারের নাম লিখ।** *[বাকাশিবো : '১৪R]*

**উত্তর :** বহুল জনপ্রিয় কিছু মাইক্রোকন্ট্রোলার : Intel 8051, PIC 16F84, Atmega 88.

৬। **ইমবেডেড মাইক্রোকন্ট্রোলার কাকে বলে?** *[বাকাশিবো : '১০, '১১, '১২, '১২R]*

**অথবা, ইমবেডেড মাইক্রোকন্ট্রোলার কী?**

**উত্তর :** এ ইমবেডেড কন্ট্রোলার এমন এক ধরনের কন্ট্রোলার IC, যেখানে কোন সার্কিট বা সিস্টেমকে নিয়ন্ত্রণ করার জন্য প্রয়োজনীয় বিভিন্ন ডিভাইস, কমপোনেন্ট, IC ইত্যাদি একই IC-তে ইন্টিগ্রেটেড অবস্থায় থাকে। অর্থাৎ কোন সার্কিট বা সিস্টেমকে পরিচালনা করার যাবতীয় ডিভাইস, কম্পোনেন্ট, IC ইত্যাদি একটি IC-তে ইমবেডেড অবস্থায় থাকে।

৭। **মাইক্রোকন্ট্রোলারের গতি (Speed) বলতে কী বুঝ?** *[বাকাশিবো : '১২R]*

**উত্তর :** মাইক্রোকন্ট্রোলারের গতি (Speed) বলতে এর প্রোগ্রাম নির্বাহ করা সময়কে বুঝায়। মাইক্রোকন্ট্রোলার নির্বাচনের ক্ষেত্রে গতি (Speed) একটি গুরুত্বপূর্ণ Criteria। সর্বোচ্চ Speed কত এবং এটি মাইক্রোকন্ট্রোলারের সাথে Support করে কিনা তা অবশ্যই লক্ষ্য রাখতে হবে। বিভিন্ন Speed-এর Microconroller বাজারে পাওয়া যায়, যেমন– 12 MHz, 16 MHz, 20 MHz ইত্যাদি।

৮। **মাইক্রোকন্ট্রোলারের প্যাকেজিং (Packaging) বলতে কী বুঝ?** *[বাকাশিবো : '১২R]*

**উত্তর :** বিভিন্ন কোম্পানির বিভিন্ন প্যাকেজিং Formate-এ Microcontroller বাজারে পাওয়া যায়, যেমন– 40 pin DIP (Dual inline package), QFP (Quad flat package) এবং অন্যান্য Package ইত্যাদি। মাইক্রোকন্ট্রোলারের পরিধি বা জায়গা; অ্যাসেমব্লিকং, প্রোটোটাইপিং ইত্যাদি Packaging format-এর উপর নির্ভর করে।

৯। **মাইক্রোকন্ট্রোলারের মেমোরি বলতে কী বুঝ?** *[বাকাশিবো : '১২]*

**উত্তর :** মাইক্রোকন্ট্রোলারের কার্যপরিধি তার মেমোরি ROM ও RAM-এর উপর নির্ভর করে। তাই Microcontroller নির্বাচনের ক্ষেত্রে মেমোরি সম্পর্কে জ্ঞান হওয়া একান্ত প্রয়োজন। 8051 Microcontroller-এ ROM হিসেবে 4K bytes এবং RAM হিসেবে 128 bytes ব্যবহৃত হয়। ঠিক তেমনি 8052 সিরিজে ROM-8K bytes এবং RAM-256 bytes ব্যবহৃত হয়।

১০। **Microcontroller এর ব্যবহার লেখ।** *[বাকাশিবো : '১১]*

**উত্তর :** বিভিন্ন ক্ষেত্রে মাইক্রোকন্ট্রোলার ব্যবহার করা হয়। যেমন : ট্রাফিক কন্ট্রোল লাইট, টিভি রিমোট কন্ট্রোল, মাইক্রোওয়েভ, প্রিন্টার, খেলনা ইত্যাদি।

**১১। একটি মৌলিক মাইক্রোকন্ট্রোলারের অভ্যন্তরে ন্যূনতম কী কী উপাদান সন্নিহিত থাকে?**

**উত্তর :** একটি মাইক্রোকন্ট্রোলারের অভ্যন্তরে ন্যূনতম নিম্নবর্ণিত উপাদানগুলো সন্নিহিত থাকে :

(ক) প্রসেসর

(খ) মেমোরি

(গ) ইনপুট/আউটপুট পোর্ট।

**১২। একটি আধুনিক মাইক্রোকন্ট্রোলারের অভ্যন্তরে কী কী অংশ সন্নিহিত থাকে?**

**উত্তর :** একটি আধুনিক মাইক্রোকন্ট্রোলারে মাইক্রোকম্পিউটারের ন্যূনতম উপাদান যেমন– প্রসেসর, মেমোরি ও ইনপুট/আউটপুট পোর্ট ছাড়াও একটি পূর্ণাঙ্গ কম্পিউটারের প্রায় সব অংশই যেমন ADC, DAC, টাইমার, মনিটর প্রোগ্রাম (Monitor program), ইন্টারাপ্ট কন্ট্রোল (Interrupt control), সিরিয়াল ইনপুট/আউটপুট (serial input/output) ইত্যাদি সন্নিহিত থাকে।

**১৩। মাইক্রোপ্রসেসর কী? উদাহরণ দাও।**

**উত্তর :** মাইক্রোপ্রসেসর একটি জটিল সিকোয়েন্সিয়াল ডিজিটাল প্রসেসর যা সাধারণ কাজে ব্যবহৃত, প্রোগ্রামযোগ্য, ক্লক পরিচালিত, LSI বা VLSI প্রযুক্তিতে তৈরি একটি সেমিকন্ডাক্টর ডিভাইস এবং যা বাহ্যিক মেমোরি থেকে বাইনারি নির্দেশনা ও ইনপুট গ্রহণ করে এবং নির্দেশনা মোতাবেক প্রক্রিয়া শেষে ফলাফল উৎপন্ন করে। মাইক্রোপ্রসেসরের উদাহরণ : Intel 8085, Intel 8086, MC 6800, Z80 ইত্যাদি।

**১৪। মাইক্রোকন্ট্রোলারকে Embedded controller বলা হয় কেন?**

**উত্তর :** যে সমস্ত কম্পিউটার অন্য ডিভাইস বা মেশিনের সাথে এমনভাবে সন্নিহিত থাকে যে এটা ডিভাইসের বৈশিষ্ট্য বা কাজকে নিয়ন্ত্রণ করে, তাদেরকে Embedded controller বলা হয়। মাইক্রোকন্ট্রোলারকে Embedded controller বলার কারণ, এর প্রসেসর ও সাপোর্ট সার্কিটসমূহ একই ডিভাইসে সন্নিহিত থাকে এবং এটা কোন একটি বিশেষ কার্য নিয়ন্ত্রণ করে, যেমন– মাইক্রোওয়েভ ওভেন, কার ব্রেকিং বা ক্রুজ মিসাইল ইত্যাদি।

**১৫। Embedded প্রসেসর হিসেবে মাইক্রোপ্রসেসর কম ব্যবহার হয় কেন?**

**উত্তর :** Embedded প্রসেসর হিসেবে মাইক্রোপ্রসেসর ও মাইক্রোকন্ট্রোলার উভয়ই ব্যবহৃত হয়। কিন্তু মাইক্রোপ্রসেসরে পাওয়ার খরচ বেশি হয় বিধায় এটা ইমবেডেড প্রসেসর হিসেবে কম ব্যবহৃত হয়।

**১৬। NMOS এর তুলনায় CMOS microcontroller এর সুবিধা কী?**

**উত্তর :** NMOS (N channel MOS FET) মাইক্রোকন্ট্রোলারের তুলনায় CMOS (Complementary MOS FET) মাইক্রোকন্ট্রোলারের আকার ছোট ও পাওয়ার অপচয় অনেক কম, ক্লকের গতিও অনেক বেশি।

**১৭। ৪-বিট মাইক্রোকন্ট্রোলার বলতে কী বুঝায়?**

**উত্তর :** যদি কোন মাইক্রোকন্ট্রোলারের ALU এক নির্দেশনায় এক বাইট (৪ বিট) এ গাণিতিক ও যৌক্তিক কাজ সম্পাদন করে তবে ঐ মাইক্রোকন্ট্রোলারকে ৪-বিট মাইক্রোকন্ট্রোলার বলে। ৪-বিট মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ বাসের আকার ৪-বিট। এটা ৪-বিটের ডাটাকে এক ক্লক সাইকেলে প্রক্রিয়া করতে পারে, অর্থাৎ এর ওয়ার্ড সাইজ ৪-বিট। উদাহরণ : Intel 8031/8051, Motorola MC68H11 ও PIC1X পরিবার।

**১৮। উদাহরণসহ CISC মাইক্রোকন্ট্রোলারের সংজ্ঞা দাও।**

**উত্তর :** যখন কোন মাইক্রোকন্ট্রোলার ইউনিট গাণিতিক ও যৌক্তিক নির্দেশনা সম্পাদন এবং মেমোরি কার্য সম্পাদন-কালীন ALU অপারেশন ও ডাটা ট্রান্সফার অপারেশন কাজে ব্যবহৃত নির্দেশনা অনেক অ্যাড্রেসিং মুড (addressing mode) সমর্থন করে তবে ঐ মাইক্রোকন্ট্রোলারকে CISC architecture microcontroller বলে। উদাহরণ : Intel 8096, 8086 ISA.

**১৯। উদাহরণসহ RISC মাইক্রোকন্ট্রোলারের সংজ্ঞা দাও।**

**উত্তর :** যখন কোন মাইক্রোকন্ট্রোলার ইউনিট গাণিতিক ও যৌক্তিক নির্দেশনা সম্পাদন কাজে এবং মেমোরি অ্যাক্সেসকালীন ALU অপারেশন ও ডাটা ট্রান্সফার অপারেশন নির্দেশনায় (load, store, push ও pop) খুবই স্বল্পসংখ্যক অনেক অ্যাড্রেসিং মুড (addressing mode) সমর্থন করে তবে ঐ মাইক্রোকন্ট্রোলারকে RISC architecture microcontroller বলে। অ্যাড্রেসিং মুড কম বিধায় RISC ডাটা ট্রান্সফার, গাণিতিক, যৌক্তিক ও অন্যান্য অপারেশন বিভিন্নভাবে সম্পাদন করার জন্য নমনীয় নয়। উদাহরণ : ARM প্রসেসর পরিবারভিত্তিক MCU, LC3, MIPS, PowerPC (IBM), SPARC (Sun)।

**২০। কম্পিউটারের কর্মসম্পাদন ক্ষমতা প্রকাশ করার সমীকরণটি লিখ।**

**উত্তর :** কম্পিউটারের কর্মসম্পাদন ক্ষমতা প্রকাশ করার জন্য বর্তমানে নিচের সমীকরণটি বহুল ব্যবহৃত হয় :

$$\frac{\text{time}}{\text{program}} = \frac{\text{time}}{\text{cycle}} \times \frac{\text{cycles}}{\text{instruction}} \times \frac{\text{instructiions}}{\text{program}}$$

**২১। ক্যাশ মেমোরি কী?**

**উত্তর :** ঘন ঘন ব্যবহৃত নির্দেশনা ও ডাটাকে সাময়িক জমা রাখার জন্য প্রসেসর ও মেমোরির মাঝখানে একটি ছোট ও উচ্চ গতির মেমোরি স্থাপন করা হয় যা ডাটা ও নির্দেশনা উভয়কে জমা রাখে এবং একে সহজে অ্যাক্সেস করা যায়। একে ক্যাশ (cache) মেমোরি বলে

**২২। মাইক্রোপ্রসেসর কাকে বলে?**

**উত্তর :** মাইক্রোপ্রসেসর বলতে সাধারণত General purpose microprocessor-কে বুঝায়। অর্থাৎ এর সাথে কোনো RAM, ROM, I/O port, Timer যুক্ত থাকে না। এর উদাহরণ হলো– 8086, 80386, 80486 পেন্টিয়াম ইত্যাদি।

**২৩। মাইক্রোকন্ট্রোলারের প্রতি Unit খরচ বলতে কী বুঝ?**

**উত্তর :** যে প্রোডাক্ট মাইক্রোকন্ট্রোলার ব্যবহৃত হয়েছে, তার মোট খরচ কত, তা জানা অত্যন্ত জরুরি একটি বিষয়। প্রতি Unit মাইক্রোকন্ট্রোলারের কস্ট যত কম হবে, Product-এর দাম ততই কমে আসবে। তাই স্বল্প খরচে মাইক্রোকন্ট্রোলার নির্বাচন একান্ত জরুরি।

**২৪। ইমবেডেড সিস্টেমের বৈশিষ্ট্য কী?**

**উত্তর :** ইমবেডেড সিস্টেমের কিছু বৈশিষ্ট্য নিচে তুলে ধরা হলো।

(ক) এটি নির্দিষ্ট কাজ করতে সক্ষম ।

(খ) এটি এক ধরনের মেকানিক্যাল ডিভাইস।

(গ) এতে মাইক্রোপ্রসেসর ও মাইক্রোকন্ট্রোলার ব্যবহার করা হয়।

**২৫। মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লেখ।**

**উত্তর :** মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য নিম্নরূপ :

(ক) এটি Highly Integrated Circuit বা সিঙ্গেল চিপ।

(খ) এর মধ্যে মাইক্রোপ্রসেসর মেমোরি (ROM, RAM), I/O, Timer Built in থাকে।

(গ) এটি ছোট ও ওজনে হালকা।

(ঘ) এতে ডিজিটাল I/O ব্যবহৃত হয়।

(ঙ) এটি হার্ডওয়্যার ও সফ্টওয়্যার ইন্টারফেস বিশিষ্ট।

**২৬। 8051 মাইক্রোকন্ট্রোলারের পরিবারের 2 টি মাইক্রোকন্ট্রোলারের নাম কী?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের পরিবারের 2 টি সদস্য হলো – (ক) 8052, (খ) 8031।

**২৭। 8051 মাইক্রোকন্ট্রোলার কত বিটের?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলার 8-bit।

**২৮। মাইক্রোকন্ট্রোলার তৈরিকৃত কোম্পানিগুলোর নাম কী কী?**

**উত্তর :** মাইক্রোকন্ট্রোলার তৈরিকৃত কোম্পানিগুলোর নাম নিম্নরূপ :

(ক) ইন্টেল (Intel), (খ) অ্যাটমেল (Atmel), (গ) ফিলিপস/সিগনিলাইস (Philips/Signelies), (ঘ) ইনফাইনিওন (Infineon), (ঙ) ডালাস সেমি/ম্যাক্সিম (Dallas Semi/Maxim) ইত্যাদি।

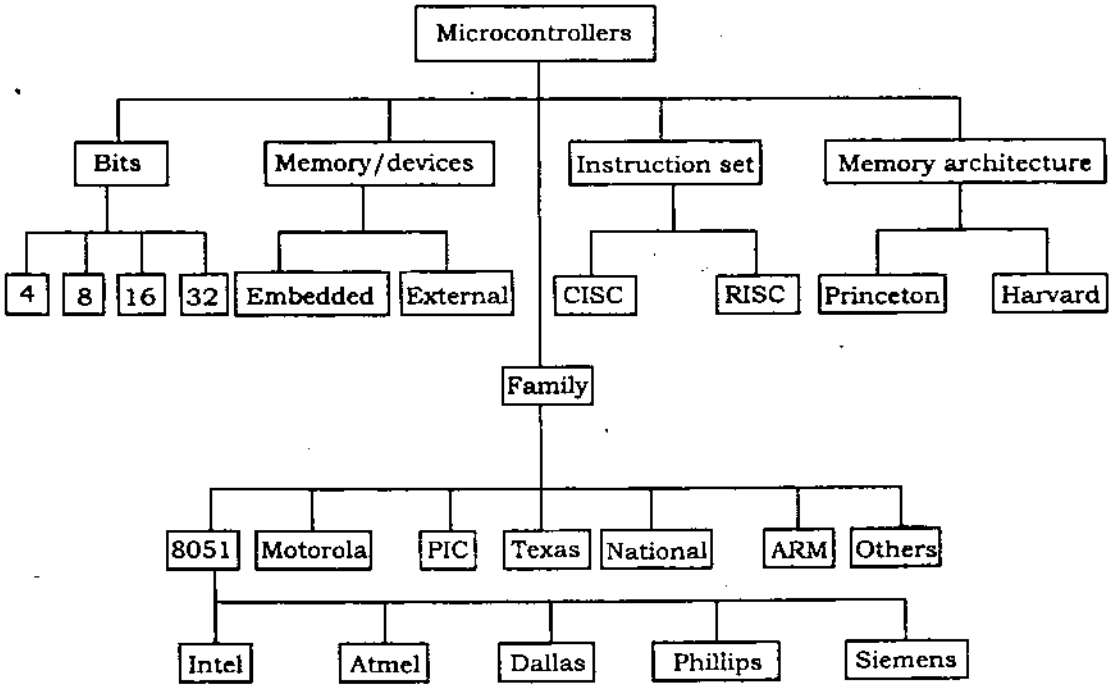
**২৯। Harvard আর্কিটেকচারের সুবিধা কী?**

**উত্তর :** প্রোগ্রাম ও ডাটা মেমোরি আলাদা আলাদা মেমোরি ব্যবহার করার জন্য তুলনামূলকভাবে গতি বেশি।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **মাইক্রোকন্ট্রোলারের শ্রেণিবিভাগ উল্লেখ কর।** *[বাকাশিবো : '১৫]*

**উত্তর :** মেমোরি গঠন, প্রসেসিং ওয়ার্ডের আকার, মেমোরি ডিভাইস ও নির্দেশনা সেটের উপর ভিত্তি করে মাইক্রোকন্ট্রোলারকে নিচের ছক মোতাবেক বিভক্ত করা যায় :




**চিত্র :** মাইক্রোকন্ট্রোলারের প্রকারভেদ

(ক) প্রসেসিং ওয়ার্ডের আকারের উপর ভিত্তি করে–

(১) 4-বিট মাইক্রোকন্ট্রোলার,

(২) 8-বিট মাইক্রোকন্ট্রোলার,

(৩) 16-বিট মাইক্রোকন্ট্রোলার ও

(৪) 32-বিট মাইক্রোকন্ট্রোলার।

(খ) মেমোরি ডিভাইসের উপর ভিত্তি করে–

(১) সন্নিহিত মেমোরি মাইক্রোকন্ট্রোলার (Embedded memory microcontroller)

(২) বাহ্যিক মেমোরি মাইক্রোকন্ট্রোলার (External memory microcontroller)

(গ) ইনস্ট্রাকশন সেটের উপর ভিত্তি করে–

(১) Complex Instruction Set Computer (CISC)

(২) Reduced Instruction Set Computer (RISC)

(ঘ) মেমোরি গঠনের উপর ভিত্তি করে–

(১) Von-Neumann (Princeton) architecture microcontroller

(২) Harvard architecture microcontroller.

২। **মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের পার্থক্য লিখ।** *[বাকাশিবো : '১৪, '১৪R]*

**উত্তর :** মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের পার্থক্য নিচে উল্লেখ করা হলো :

| মাইক্রোপ্রসেসর | মাইক্রোকন্ট্রোলার |
|---|---|
| ১। এটা সাধারণ ব্যবহারের কম্পিউটার এর একটি সিপিইউ। | ১। এটা নিজেই একটি মাইক্রোকম্পিউটার। |
| ২। এটা কম্পিউটার সিস্টেমের প্রাণ। | ২। এটা ইমবেডেড সিস্টেমের প্রাণ। |
| ৩। এটি শুধু একটি প্রসেসর, কম্পিউটার সিস্টেমে মেমোরি ও ইনপুট/ আউটপুট বাহ্যিকভাবে সংযুক্ত। | ৩। মাইক্রোকন্ট্রোলারে প্রসেসর, মেমোরি ও ইনপুট/আউটপুট পোর্ট একই চিপে সন্নিহিত। |
| ৪। যেহেতু মেমোরি, টাইমার, ইন্টারাপ্ট, ইনপুট/ আউটপুট ইত্যাদি বাহ্যিকভাবে সংযুক্ত, সুতরাং সিস্টেম বড় হয়। | ৪। যেহেতু মেমোরি, টাইমার, ইন্টারাপ্ট, ইনপুট/ আউটপুট পোর্ট অভ্যন্তরীণভাবে সংযুক্ত, সুতরাং সিস্টেম অনেক ছোট। |
| ৫। সম্পূর্ণ সিস্টেমের খরচ বেশি। | ৫। সম্পূর্ণ সিস্টেমের খরচ কম। |
| ৬। শক্তি অপচয় বেশি। | ৬। শক্তি অপচয় অনেক কম। |
| ৭। অধিকাংশ মাইক্রোপ্রসেসরের পাওয়ার সেভিং বৈশিষ্ট্য নেই। | ৭। প্রায় সব মাইক্রোকন্ট্রোলারে বিভিন্নভাবে পাওয়ার সেভিং বৈশিষ্ট্য আছে। |
| ৮। মাইক্রোপ্রসেসরে রেজিস্টারের সংখ্যা কম, অধিকাংশ অপারেশন মেমোরি-ভিত্তিক। | ৮। এতে অধিক সংখ্যক রেজিস্টার থাকে, অপারেশন রেজিস্টারভিত্তিক। |
| ৯। মাইক্রোপ্রসেসর Von Neumann আর্কিটেকচারের অর্থাৎ এর প্রোগ্রাম ও ডাটা একই মেমোরিতে জমা থাকে। | ৯। মাইক্রোকন্ট্রোলার Harvard আর্কিটেকচারের অর্থাৎ এর প্রোগ্রাম ও ডাটা আলাদা। |
| ১০। মাইক্রোপ্রসেসরকে প্রয়োজন অনুসারে পুনঃপ্রোগ্রাম করা হয়। | ১০। মাইক্রোকন্ট্রোলারকে একটি কাজের জন্য স্থায়ীভাবে প্রোগ্রাম করা হয়। |
| ১১। এর RAM-to-ROM অনুপাত বড়। | ১১। এর RAM-to-ROM অনুপাত ছোট। |
| ১২। মেমোরি অনেক বড় এবং প্রয়োজনে বৃদ্ধি করা যায়। | ১২। এর মেমোরির আকার ছোট ও সীমাবদ্ধ (limited)। |
| ১৩। এর ক্লকের গতি অনেক বেশি। | ১৩। এর ক্লকের গতি সীমিত। |
| ১৪। এটা বুলিয়ান ফাংশন সম্পূর্ণ করতে পারে না। | ১৪। এটা বুলিয়ান ফাংশন সম্পূর্ণ করতে পারে। |
| ১৫। এর খুব কম সংখ্যক পিনকে প্রোগ্রামযোগ্য করা যায়। | ১৫। এর অধিকাংশ পিনকে প্রোগ্রামযোগ্য করা যায়। |
| ১৬। এর খুব কম সংখ্যক bit handling instruction আছে। | ১৬। এর অনেক bit handling instruction আছে। |
| ১৭। এর মেমোরি অ্যাক্সেস টাইম বেশি। | ১৭। এর মেমোরি অ্যাক্সেস টাইম কম। |
| ১৮। পারসোনাল কম্পিউটার ও ল্যাপটপে এটা ব্যাপক ব্যবহৃত হয়। | ১৮। ছোট কন্ট্রোল সিস্টেমে এটা ব্যবহৃত হয়। |
| ১৯। উদাহরণ : Intel 8086, Motorola MC 6800, Z80 ইত্যাদি। | ১৯। Intel 8051, PIC 16F877 ইত্যাদি। |

৩। **হার্ভার্ড আর্কিটেকচারের সুবিধা/অসুবিধা লিখ।** *[বাকাশিবো : '১৪R]*

**উত্তর : হার্ভার্ড আর্কিটেকচারের সুবিধা :**

(১) দক্ষ পাইপলাইনিং- অপারন্ড ও ইনস্ট্রাকশন যুগপৎ সংঘটিত হতে পারে।

(২) ডাটা বাস ও নির্দেশনা বাস ভিন্ন-সমান্তরালে অপারেশন হয় এবং প্রসেসরে গতি বৃদ্ধি পায়।

(৩) ইনস্ট্রাকশন বেশি, হার্ডওয়্যার কম।

**হার্ভার্ড আর্কিটেকচারের অসুবিধা :**

(১) এর ব্যবহার কম।

(২) একে কার্যে পরিণত করা অধিক জটিল।

(৩) এর পিন সংখ্যা বেশি।

৪। **একটি মাইক্রোকন্ট্রোলার নির্বাচনের শর্তসমূহ আলোচনা কর।** *[বাকাশিবো : '১৪]*

**উত্তর :** প্রকৃতপক্ষে, চার প্রকারের 8-বিট মাইক্রোকন্ট্রোলার রয়েছে, যেমন– 6811, 8051, Zilog'sZ8, PIC 16X। প্রত্যেক মাইক্রোকন্ট্রোলারের আলাদা ইনস্ট্রাকশন সেট এবং আলাদা রেজিস্টার সেট রয়েছে এবং প্রত্যেকে আলাদাভাবে কাজের জন্য তৈরি। একটি প্রোগ্রামের জন্য তৈরি মাইক্রোকন্ট্রোলার দিয়ে অন্য কাজ করা সম্ভব নয়। অর্থাৎ যে কাজের জন্য Microcontroller তৈরি, শুধুমাত্র সেই কাজের জন্য এটিকে Microcontroller ব্যবহার করা যাবে, অন্য কোথাও নয়।

মাইক্রোকন্ট্রোলার নির্বাচনের ক্ষেত্রে গুরুত্বপূর্ণ তিনটি শর্ত রয়েছে। যথা :

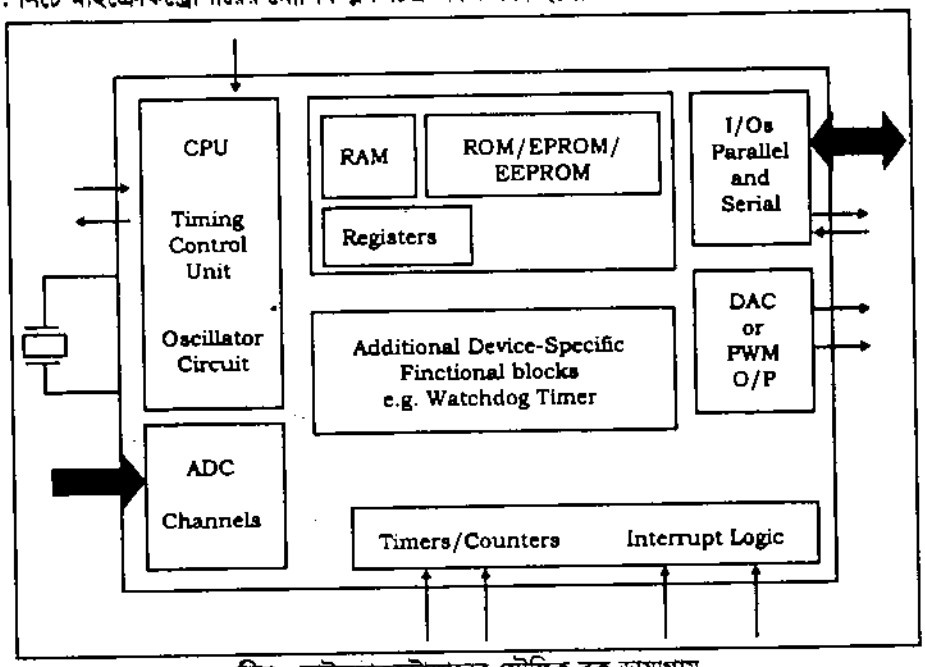
১। যে কাজের জন্য Microcontroller তৈরি হবে, সেই কাজ দক্ষতার সাথে স্বল্প খরচে হচ্ছে কিনা, তা লক্ষ রাখতে হবে।

২। যে কাজের জন্য Microcontroller তৈরি, সেই কাজের উপযুক্ত Software, Compiler, Assembler এবং Debuggers আছে কিনা, তা লক্ষ রাখতে হবে।

৩। Microcontroller-এর প্রাচুর্যতা এবং সঠিক উৎস থাকতে হবে।

৫। **মাইক্রোকন্ট্রোলারের মৌলিক ব্লক চিত্র অঙ্কন কর।** *[বাকাশিবো : '১২]*

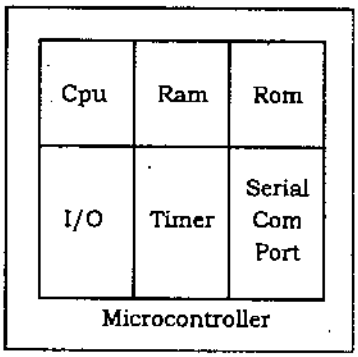
**উত্তর :** নিচে মাইক্রোকন্ট্রোলারের মৌলিক ব্লক চিত্র অঙ্কন করা হলো :




**চিত্র :** মাইক্রোকন্ট্রোলারের মৌলিক ব্লক ডায়াগ্রাম

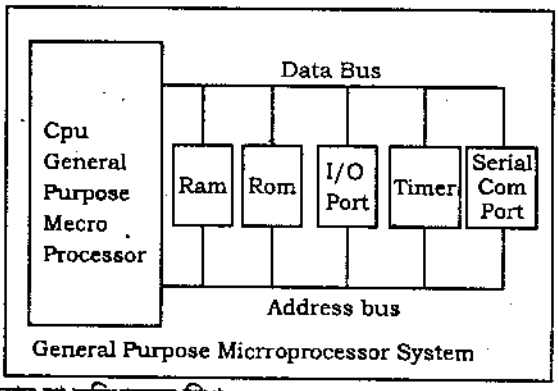
৬। **মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন কর।** *[বাকাশিবো : '১২]*

**উত্তর :** নিচে মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন করা হলো :




৭। **মাইক্রোপ্রসেসরের ব্লক ডায়াগ্রাম অঙ্কন কর।** *[বাকাশিবো : '১২]*

**উত্তর :** নিচে মাইক্রোপ্রসেসরের ব্লক ডায়াগ্রাম দেওয়া হলো :




৮। **মাইক্রোকন্ট্রোলারের ব্যবহার বা সুবিধাসমূহ লিখ।** *[বাকাশিবো : '১১]*

**উত্তর :** নিচে মাইক্রোকন্ট্রোলারের সুবিধা ও ব্যবহার উল্লেখ করা হলো :

**বাসাবাড়ি (Home) :**

(১) বিভিন্ন ধরনের অ্যাপ্লায়েন্স
(২) টেলিফোন
(৩) ইন্টারকম
(৪) সিকিউরিটি সিস্টেম
(৫) গ্যারেজ ডোর ওপেনারস
(৬) আনসারিং মেশিন (Answering machines)
(৭) ক্যামেরা
(৮) খেলনা
(৯) টিভি
(১০) ক্যাবল টিভি টিউনার
(১১) ভিসিআর
(১২) রিমোট কন্ট্রোল
(১৩) ভিডিও গেমস
(১৪) হোম কম্পিউটার
(১৫) মিউজিক্যাল ইনস্ট্রুমেন্ট
(১৬) সেলাই মেশিন
(১৭) লাইটিং কন্ট্রোল
(১৮) ফ্যাক্স মেশিন
(১৯) সেলুলার ফোন
(২০) ব্যায়ামের যন্ত্র

**অফিস (Office) :**

(১) সিকিউরিটি সিস্টেম
(২) পেজিং
(৩) কপিয়ার
(৪) ফ্যাক্স মেশিন
(৫) মাইক্রোওয়েভ
(৬) টেলিফোন
(৭) প্রিন্টার
(৮) কম্পিউটার

**অটো (Auto) :**

(১) ট্রিপ কম্পিউটার (৪) ট্র্যান্সমিশন কন্ট্রোল
(২) ইঞ্জিন কন্ট্রোল (৫) ইন্টারটেইনমেন্ট
(৩) ইন্সট্রুমেনটেশন (৬) আবহাওয়া কন্ট্রোল।

**৯। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?** *[বাকাশিবো : '১০]*

**উত্তর :** ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(১) এটি 40 পিনের DIP প্যাকেজ।
(২) এটিতে On Chip Oscillator ও Clock সার্কিট থাকে।
(৩) এতে 8 টি বিটের CPU থাকে।
(৪) CPU একই সময়ে 8 বিট ডাটা নিয়ে কাজ করে।
(৫) 8 বিটের 4 (চার) টি I/O পোর্ট থাকে।
(৬) এতে সিরিয়াল পোর্ট বিদ্যমান।
(৭) এতে 6 টি Interrupt সুবিধা থাকে।
(৮) এতে 2 টি 16 বিটের টাইমার থাকে।
(৯) এতে 4k Byte এর ROM থাকে।
(১০) এতে 128 byte এর RAM বিদ্যমান।

**১০। 8051-এর মূল বৈশিষ্ট্য লিখ।** *[বাকাশিবো : '১০]*

**উত্তর :** 8051 family-এর প্রধান ও মূল সদস্য হলো 8051 ইন্টেল কোম্পানি একে mcs-51 নামেও ব্যবহার করে। এর মূল বৈশিষ্ট্যগুলো হলো :

| | |
|---|---|
| ROM– | 4K bytes |
| RAM– | 128 bytes |
| Tims– | 2 NOS. |
| I/O Ports– | 32 NOS. |
| Serial Port– | 1 |
| Interrupt sources– | 6 |

এ ছাড়াও 8052 এবং 8031 নামে দুটি সদস্য (Members) আছে।

**১১। মাইক্রোকন্ট্রোলার বলতে কী বুঝায়?**

**উত্তর :** ইংরেজি শব্দ "Micro" অর্থ ছোট এবং "Controller" অর্থ কোন অবজেক্ট (object) বা প্রক্রিয়া বা ইভেন্টের (event) নিয়ন্ত্রণকারী। মাইক্রোকন্ট্রোলার কোন প্রক্রিয়া বা ডিভাইসকে নিয়ন্ত্রণকারী ছোট আকারের একটি কম্পিউটার। এটি VLSI প্রযুক্তিতে একক চিপে (single chip) জটিল সিকুয়েন্সিয়াল লজিক সার্কিটে তৈরি একটি ছোট, প্রোগ্রামযোগ্য (programmable) ও সর্বজনীন কম্পিউটার। এটি মূলত একটি ইন্টিগ্রেটেড সার্কিট (Integrated circuit, IC) যাতে ডেস্কটপ কম্পিউটারের প্রায় সব অংশই বিদ্যমান। মাইক্রোকন্ট্রোলারের অভ্যন্তরে ন্যূনতম নিম্নলিখিত উপাদানসমূহ থাকে :

(ক) একটি কম্পিউটার প্রসেসর, যা প্রোগ্রাম সম্পাদন করে;
(খ) একটি ছোট RAM, যা প্রোগ্রাম চলাকালীন চলকের উপাত্ত সংরক্ষণ করে;
(গ) একটি ছোট ROM, যা প্রোগ্রাম সংরক্ষণ করে;
(ঘ) ইনপুট/আউটপুট ডিভাইস পোর্ট, যা বহির্জগতের সাথে মাইক্রোকন্ট্রোলারের যোগাযোগ প্রতিষ্ঠা করে।

প্রযুক্তির উন্নতির কল্যাণে বিশেষ করে মাইক্রোইলেকট্রনিক্স প্রযুক্তিতে ডিভাইসে সার্কিট সন্নিহিতের হার দিন দিন বৃদ্ধি পাওয়ার ফলে আধুনিক মাইক্রোকন্ট্রোলারে একটি পূর্ণাঙ্গ কম্পিউটারের প্রায় সব অংশই যেমন : ADC, DAC, টাইমার, মনিটর প্রোগ্রাম (Monitor program), ইন্টারাপ্ট কন্ট্রোল (Interrup control), সিরিয়াল ইনপুট/আউটপুট (serial input/output) ইত্যাদি সন্নিহিত থাকে। উদাহরণ : বহুল জনপ্রিয় কিছু মাইক্রোকন্ট্রোলার– Intel 8051, PIC 16F84, Atmega 88.

**১২। কী কী বৈশিষ্ট্যের জন্য কোন কম্পিউটারকে মাইক্রোকন্ট্রোলার ক্যাটাগরিতে বিবেচনা করা হয়?**

**উত্তর :** কম্পিউটারের কিছু কিছু সাধারণ বৈশিষ্ট্য আছে যার জন্য মাইক্রোকন্ট্রোলারকে বিভিন্ন নামে সংজ্ঞায়িত করা হয়। যে সমস্ত কম্পিউটার ঐ বৈশিষ্ট্যের অধিকাংশের সাথে মিলে যায় তাদেরকে মাইক্রোকন্ট্রোলারের ক্যাটাগরিতে বিবেচনা করা হয়। যেমন, নিচের কম্পিউটারসমূহকে তাদের বিশেষ বৈশিষ্ট্যের কারণে মাইক্রোকন্ট্রোলার হিসেবে বিবেচনা করা হয় :

(১) যে সমস্ত কম্পিউটার অন্য ডিভাইস বা মেশিনের সাথে এমনভাবে সন্নিহিত থাকে যে, এটি ডিভাইসের বৈশিষ্ট্য বা কাজকে নিয়ন্ত্রণ করে, তাদেরকে Embedded controller-ও বলা হয়। একে Embedded controller বলার কারণ, এর প্রসেসর ও সাপোর্ট সার্কিটসমূহ একই ডিভাইসে সন্নিহিত থাকে এবং এটা কোন একটি বিশেষ কার্য নিয়ন্ত্রণ করে, যেমন– মাইক্রোওয়েভ ওভেন, কার ব্রেকিং বা ক্রুজ মিসাইল ইত্যাদি।

(২) যে সমস্ত কম্পিউটার শুধুমাত্র একটি কাজ বা অত্যন্ত আন্তঃসম্পর্কিত কাজের সেট সম্পাদন করে এবং তজ্জন্য একটিমাত্র প্রোগ্রাম সম্পাদন করে। প্রোগ্রামটি ROM এ জমা থাকে এবং যা সাধারণত পরিবর্তন করা হয় না।

(৩) ব্যাটারিচালিত কম্পিউটার যা 50 মিলিওয়াটেরও কম পাওয়ার অপচয় করে।

**১৩। CISC ও RISC এর প্রসেসরের পার্থক্য লিখ।**

**উত্তর :** CISC ও RISC প্রসেসরের পার্থক্য নিচে উল্লেখ করা হলো :

| CISC | RISC |
|---|---|
| ১। অনেক অ্যাড্রেসিং মুড। | ১। স্বল্পসংখ্যক অ্যাড্রেসিং মুড। |
| ২। মেমোরি কাজে শুধু LOAD/STORE নির্দেশনা ব্যবহৃত হয়। | ২। মেমোরি কাজে অনেক নির্দেশনা ব্যবহৃত হয়। |
| ৩। এতে হার্ডওয়্যারে জোর দেয়া হয়। | ৩। সফটওয়্যারে জোর দেওয়া হয়। |
| ৪। জটিলতা মাইক্রো-কোডে। | ৪। জটিলতা কম্পাইলারে। |
| ৫। নির্দেশনার দৈর্ঘ্য পরিবর্তনশীল। | ৫। একই দৈর্ঘ্যের নির্দেশনা। |
| ৬। কোডের আকার ছোট, সাইকেল/সেকেন্ড বেশি। | ৬। কোডের আকার বড়, সাইকেল/সেকেন্ড কম। |
| ৭। মেমোরি– মেমোরি অপারেশন। | ৭। রেজিস্টর- রেজিস্টর অপারেশন। |
| ৮। এর নির্দেশনা জটিল এবং সংখ্যায় অনেক (~ 200-300 instructions)। | ৮। এতে স্বল্পসংখ্যক সরল নির্দেশনা (~ 50) আছে। |
| ৯। নির্দেশনা সংরক্ষণের জন্য কমসংখ্যক ট্রানজিস্টর প্রয়োজন। | ৯। মেমোরি রেজিস্টারের জন্য অধিক ট্রানজিস্টর প্রয়োজন। |
| ১০। একক ক্লক নির্দেশনা। | ১০। বহু ক্লক নির্দেশনা। |
| ১১। প্রোগ্রাম রচনা নমনীয়। | ১১। প্রোগ্রাম রচনা কঠিন। |
| ১২। উৎপাদন খরচ বেশি। | ১২। উৎপাদন খরচ কম। |
| ১৩। ছোট মেমোরি। | ১৩। বড় মেমোরি। |
| ১৪। প্রতি প্রোগ্রামে নির্দেশনা হ্রাস করে। | ১৪। প্রতি ইনস্ট্রাকশনে সাইকেল সংখ্যা হ্রাস করে। |
| ১৫। কাজের গতি মন্থর। | ১৫। কাজের গতি অনেক দ্রুত। |
| ১৬। Pipeline নেই বা খুব নগণ্য। | ১৬। Pipeline আছে। |

**১৪। Von-Neumann এর bottleneck সমস্যা কী? বা Von-Neumann আর্কিটেকচারের সীমাবদ্ধতা লিখ।**

**উত্তর :** Von-Neumann আর্কিটেকচারের সীমাবদ্ধতা হলো "Von-Neumann bottleneck"। এর প্রকৃত সত্য হচ্ছে– এই আর্কিটেকচারে সমস্ত নির্দেশনা এবং ডাটা একটি সাধারণ মাল্টিপ্লেক্সড বাসকে ভাগাভাগি করে প্রসেসর হতে মেমোরি বা মেমোরি হতে প্রসেসরে আসে। যেহেতু অধিকাংশ কম্পিউটারে প্রসেসরে কার্যকরী ক্ষমতা মেমোরির তুলনায় অনেক বেশি, তাই প্রসেসর ও মেমোরির মধ্যকার গড় ডাটা সম্প্রচার হার (throughput) অনেক কম। এতে একটি প্রক্রিয়ার জন্য প্রয়োজনীয় ডাটা ও নির্দেশনা মেমোরি থেকে access করার পর পরবর্তী

প্রক্রিয়ার নির্দেশনা ও ডাটার জন্য অপেক্ষা করতে হয় অর্থাৎ মাইক্রোকন্ট্রোলার কিছুই করে না ("do nothing") বা অলস থাকে। এতে প্রসেসরে কার্যকরী প্রক্রিয়ার গতিতে ভয়ানক সীমাবদ্ধতা নিয়ে আসে। মেমোরি ও প্রসেসরে গতির পার্থক্য দিন দিন যত বাড়বে throughput-ও সমহারে বৃদ্ধি পাবে। বিশেষ করে প্রসেসর যখন কোন বড় আকারের ডাটার উপর সংক্ষিপ্ত কাজ করে তখন এই সমস্যা ভয়ানক হয়।

১৫। **Von-Neumann এর bottleneck সমস্যা সমাধানের উপায় উল্লেখ কর।**

**উত্তর :** Von-Neumann আর্কিটেকচার এর bottleneck সমস্যা সমাধানের অনেক প্রচেষ্টা করা হলেও এর সম্পূর্ণ সমাধান হয়নি। নিচে সমাধানের কিছু প্রচেষ্টা উল্লেখ করা হলো–

(১) **Caching :** ঘন ঘন ব্যবহৃত নির্দেশনা ও ডাটাকে সাময়িক জমা রাখার জন্য প্রসেসর ও মেমোরির মাঝখানে একটি ছোট ও উচ্চ গতির ক্যাশ (cache) মেমোরি স্থাপন করা হয় যা ডাটা ও নির্দেশনা উভয়কে জমা রাখে এবং একে সহজে অ্যাক্সেস করা যায়। ক্যাশের প্রতিটি অংশ পরিবর্তনশীল। তবে ডাটা ও নির্দেশনার জন্য স্বতন্ত্র ও উচ্চ গতির ভিন্ন ক্যাশ ও বাসের ব্যবস্থা করা যায় যেখানে নির্দেশনা ও ডাটাকে পৃথকভাবে জমা রাখা যায়। ফলে নির্দেশনাকে ডাটার জন্য এবং ডাটাকে নির্দেশনার জন্য অপেক্ষা করতে হয় না। তারপরেও মেইন বাস ব্যবহারে এই সমস্যা কিছুটা থেকে যায়।

(২) **Prefetching :** কোন request এর জন্য দ্রুত অ্যাক্সেসের পূর্বেই কিছু ডাটাকে cache এ জমা রাখা।

(৩) **Multithreading :** ভিন্ন ভিন্ন thread এ একাধিক request কে একই সময়ে ব্যবস্থা করা।

(৪) **নতুন ধরনের RAM :** নতুন ধরনের RAM যেমন– DDR, SDRAM ইত্যাদির ব্যবহার যারা সিস্টেম ক্লকের rising ও falling উভয় প্রান্তে আউটপুট উৎপন্ন করতে পারে।

(৫) **RAMBUS :** এটা একটি মেমোরি সাব-সিস্টেম যা RAM, RAM CONTROLLER ও BUS (RAM TO PROCESSOR) নিয়ে গঠিত।

(৬) **PIM (processing in memory) :** এটা প্রসেসর ও মেমোরিকে একক ডিভাইসে সন্নিহিত করে।

১৬। **Von-Neumann এর সুবিধা/অসুবিধা লিখ।**

**উত্তর : Von-Neumann আর্কিটেকচারের সুবিধা :**

(১) অধিকাংশ ডিস্ক অপারেটিং সিস্টেম von-Neumann আর্কিটেকচারের উপর ভিত্তি করে যা কম্পিউটারকে পরিচালনায় অনেক সহজ করে।

(২) যেহেতু ডাটা ও নির্দেশনার জন্য ভিন্ন ডাটা বাসের প্রয়োজন হয় না, তাই এতে হার্ডওয়্যারের পরিমাণ অনেক কম লাগে।

(৩) একটি একক পেরিফেরাল ব্যবহার করেই অত্যন্ত সহজভাবেই প্রোগ্রাম পরিবর্তন করা যায় যা সিস্টেমের সামর্থ্য অনেক বৃদ্ধি করে।

**Von-Neumann আর্কিটেকচারের অসুবিধা :**

(১) প্রধান অসুবিধা হলো- এটা Bottleneck সমস্যা।

(২) ইনস্ট্রাকশন সিরিয়ালি সম্পাদিত হয়।

(৩) প্রত্যেক ইনস্ট্রাকশন প্রক্রিয়া হওয়ার সময় বেশি লাগে কারণ একই মেমোরিতে জমাকৃত ডাটা ও নির্দেশনাকে একটিমাত্র বাস ব্যবহার করে retrieve করতে হয়। মেমোরি অ্যাক্সেস দুভাবে হয়– প্রথমে ডাটা অ্যাক্সেস ও পরে নির্দেশনা অথবা প্রথমে নির্দেশনা এবং পরে ডাটা অ্যাক্সেস করা হয়। এতে সিস্টেম ক্যাশ হওয়ার সম্ভাবনা থাকে।

(৪) Fetch-decode-execute-reset সাইকেলে কিছু কিছু রেজিস্টার অলস (idle) থাকে।

১৭। **পরিবর্তিত হার্ভার্ড আর্কিটেকচার কী?**

**উত্তর :** পরিবর্তিত হার্ভার্ড আর্কিটেকচার অনেকটা বিশুদ্ধ হার্ভার্ড আর্কিটেকচারের মতো কিন্তু ইহা নির্দেশনা ও ডাটার মধ্যকার কঠিন বিচ্ছিন্নতা শিথিল করে, যদিও বা CPU দুই বা ততোধিক মেমোরি বাসকে একই সময়ে (concurrently) অ্যাক্সেস করে। এর প্রধান পরিবর্তন হল– একটি সাধারণ অ্যাড্রেস ক্ষেত্র দ্বারা সমর্থিত ভিন্ন ইনস্ট্রাকশন ক্যাশ ও ডাটা ক্যাশ মেমোরি। CPU যখন ক্যাশ থেকে প্রোগ্রাম নির্বাহ করে তখন ইহা বিশুদ্ধ হার্ভার্ড

আর্কিটেকচারের মত ক্রিয়া করে। আবার যখন সমর্থিত মেমোরি (backed memory) অ্যাক্সেস করে তখন ইহা ভন-নিউম্যান আর্কিটেকচারের মতো ক্রিয়া করে। আধুনিক প্রসেসরে ইহা বহুল ব্যবহৃত হয়। উদাহরণ : ARM architecture ও X86 প্রসেসর। এদের সরলভাবে হার্ভার্ড আর্কিটেকচার বলা হলেও প্রকৃতপক্ষে এরা পরিবর্তিত হার্ভার্ড আর্কিটেকচার।

**১৮। ছোট প্রয়োগক্ষেত্রে মাইক্রোকন্ট্রোলার নির্বাচনে বিবেচ্য বিষয়গুলো উল্লেখ কর।**

**উত্তর :** ছোট প্রয়োগের ক্ষেত্রে মোটামুটিভাবে মাইক্রোকন্ট্রোলার নির্বাচনের (Roughly microcontroller selection) জন্য নিম্নবর্ণিত বিষয়গুলো অবশ্যই বিবেচনায় আনতে হয় :

(ক) ব্যয় সাশ্রয়িতা ও কাজের দক্ষতার জন্য প্রয়োজনীয় কম্পিউটিং ক্ষমতা বিবেচনায়–

(১) কাজের গতি
(২) পাওয়ার খরচ
(৩) On-chip RAM/ROM এর পরিমাণ
(৪) ইনপুট/আউটপুট পিনের সংখ্যা
(৫) On-chip টাইমার
(৬) প্যাকেজিং
(৭) খরচ

(খ) হার্ডওয়্যার ও সফটওয়্যার উন্নয়ন যন্ত্রপাতি (যথা– কম্পাইলার, অ্যাসেম্বলার, ডিবাগার, প্রোগ্রামার ইত্যাদি) এর প্রাচুর্যতা।

**১৯। Von Neumann ও Haruard আর্কিটেকচারের পার্থক্য লেখ।**

**উত্তর :** নিচে Harvard ও Von Neumann আর্কিটেকচারের পার্থক্য তুলে ধরা হলো :

| Von Neumann architecutre | Harvard architecture |
|---|---|
| (ক) এই আর্কিটেকচারে প্রোগ্রাম মেমোরি ও ডাটা মেমোরি একই মেমোরি ব্যবহার করে। | (ক) এই আর্কিটেকচারে প্রোগ্রাম মেমোরি ও ডাটা মেমোরি আলাদা আলাদা মেমোরি ব্যবহার করে। |
| (খ) এটি মাইক্রোকন্ট্রোলারের চিপ ডিজাইনকে Simplify করে। | (খ) এটি মাইক্রোকন্ট্রোলারের চিপ ডিজাইনকে Complex করে। |
| (গ) একই বাস ব্যবহারের জন্য প্রোগ্রাম ও ডাটা একই সাথে কাজ করতে পারে না। | (গ) আলাদা বাস ব্যবহারের জন্য প্রোগ্রাম ও ডাটা একই সাথে কাজ করতে পারে। |
| (ঘ) কাজের গতি তুলনামূলকভাবে অনেক কম। | (ঘ) কাজের গতি তুলনামূলকভাবে অনেক বেশি। |
| (ঙ) পাইপ লাইন প্রসেসিং ব্যবহার করা যায় না। | (ঙ) পাইপ লাইন প্রসেসিং ব্যবহার করা যায়। |
| (চ) Clock Speed কম। | (চ) Clock Speed বেশি। |
| (ছ) কম সংখ্যক Control Signal এর প্রয়োজন হয়। | (ছ) বেশি সংখ্যক Control Signal-এর প্রয়োজন হয়। |
| (জ) এটি একই বাস ব্যবহার করে। | (জ) এটি একাধিক বাস ব্যবহার করে। |

**২০। ইন্টেল 8051 (Intel 8051) মাইক্রোকন্ট্রোলার এর বৈশিষ্ট্য লিখ।**

**উত্তর :** ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 4k byte এর ROM থাকে।
(2) এটি 40 পিনের DIP প্যাকেজ।
(3) CPU একই সময়ে 8 বিট ডাটা নিয়ে কাজ করে।
(4) এটিতে On chip oscillator ও clock সার্কিট থাকে।
(5) এতে 8টি বিটের CPU থাকে।
(6) এতে 6 টি Interrupt সুবিধা থাকে।
(7) 8 বিটের 4 (চার)টি I/O পোর্ট থাকে।
(8) এতে 128 byte এর RAM বিদ্যমান।
(9) এতে সিরিয়াল পোর্ট বিদ্যমান।
(10) এতে 2টি 16 বিটের টাইমার থাকে।

**২১। ইন্টেল 8048 (Intel 8048) মাইক্রোকন্ট্রোলার এর বৈশিষ্ট্য লিখ।**

**উত্তর :** ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 2টি maskable (মাস্কেবল) ইন্টারাপ্ট রয়েছে।
(2) এটি 8 বিটের মাইক্রোকন্ট্রোলার।
(3) এতে 64 byte-এর RAM আছে।
(4) এতে 1k byte এর ROM বিদ্যমান।
(5) এটি 40 পিনের DIP প্যাকেজ।
(6) এতে Timer রয়েছে।
(7) এটি 8 বিটের 3 টি I/O পোর্ট বিদ্যমান।

**২২। ইন্টেল 8096 (Intel 8096) Microcontroller এর বৈশিষ্ট্য লিখ।**

**উত্তর :** ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এতে 10 bit ADC (Analog to digital converter) বিদ্যামান।
(2) এটি 16 bit-এর মাইক্রোকন্ট্রোলার।
(3) এই মাইক্রোকন্ট্রোলারে 16 বিট বিশিষ্ট্য 2 টি টাইমার বিদ্যমান।
(4) এটি full-duplex Serial Port বিশিষ্ট্য।
(5) এতে 8 বিটের 5 টি ইনপুট/আউটপুট পোর্ট (I/O Port) রয়েছে।
(6) এতে 16 বিট এর CPU রয়েছে।

**২৩। Intel 2920 Microcontroller এর বৈশিষ্ট্য লিখ।**

**উত্তর :** ইন্টেল 2920 মাইক্রোকন্ট্রোলারের কিছু বৈশিষ্ট্য নিচে তুলে ধরা হলো :

(1) এতে Analog ডিমাল্টিপ্লেক্সার বিদ্যমান।
(2) এটি 32 bit এর মাইক্রোকন্ট্রোলার।
(3) এটিতে D/A converter ও বিদ্যমান।
(4) এটি একটি Programmable IC।
(5) এতে A/D converter ও রয়েছে।
(6) এতে Analog মাল্টিপ্লেক্সার বিদ্যমান।
(7) এতে 192 Ward এর EPROM বিদ্যমান।

**২৪। PIC16 F84A মাইক্রোকন্ট্রোলার এর বৈশিষ্ট্য লিখ।**

**উত্তর :** PIC16F84 A মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(1) এটি RISE Architecture বিশিষ্ট।
(2) এটি মাইক্রোচিপ কোম্পানির তৈরি 8 বিটের মাইক্রোকন্ট্রোলার।
(3) এতে 2টি I/O পোর্ট বিদ্যমান।
(4) এতে 35টি ইনস্ট্রাকশন বিদ্যমান।
(5) এর 13টি I/O পিন রয়েছে।
(6) এর Clock frequency 20 MHz পর্যন্ত হয়।
(7) এর প্রায় সকল ইনস্ট্রাকশন 1 মেশিন সাইকেল।
(8) এতে 1 কিলোওয়ার্ড প্রোগ্রাম মেমোরি বিদ্যমান।

**২৫। MSC-51মাইক্রোকন্ট্রোলার এর বৈশিষ্ট্য লিখ।**

**উত্তর :** MSC-51 মাইক্রোকন্ট্রোলারের কিছু বৈশিষ্ট্য উল্লেখ করা হলো :

(1) CHMOS (Complementary high speed metal oxide semiconductor) version এরও মাইক্রোকন্ট্রোলার রয়েছে।
(2) এটি 8 bit এর Microcontroller।
(3) CHMOS version এর μc-এ Idle mode থাকে।

(4) এতে MCS-48 microcontroller এর সাথে Replace করা যায়।

(5) এটি মূলত HMOS (High speed metal oxide semiconductor) Tachnology তে তৈরি।

(6) CHMOS version এর মাইক্রোকন্ট্রোলার হলো Lower Power consmption, higher noise immunity এবং high speed বিশিষ্ট।

**২৬। মাইক্রোকন্ট্রোলার যে সব মৌলিক ইউনিট দ্বারা গঠিত তাদের বর্ণনা কর।**

**উত্তর :** নিচের ইউনিটগুলো নিয়ে মাইক্রোকন্ট্রোলার গঠিত। যেমন–

(১) Central Processing Unit (CPU),

(২) মেমোরি (Memory),

(৩) ইনপুট/আউটপুট পোর্ট বা I/O Port,

(৪) সিলিয়াল পোর্ট (Serial Port),

(৫) টাইমিং এবং কন্ট্রোল ইউনিট (Timing and Control Unit),

(৬) ADC ও DAC,

(৭) ইন্টারাপ্ট লজিক (Interrupt Logic),

(৮) টাইমার এবং কাউন্টার (Timer and Counter),

(৯) অসিলেটর সার্কিট (Oscillator Circuit),

(১০) এডিশনাল সার্কিট (Additional Circuit).

নিচে গুরুত্বপূর্ণ ইউনিটগুলোর বর্ণনা পয়েন্ট আকারে দেওয়া হলো :

**(১) সেন্ট্রাল প্রসেসিং ইউনিট বা সিপিইউ সেন্ট্রাল প্রসেসিং ইউনিট বা সিপিইউ (Central Processing Unit) :**

(ক) সকল কার্যাদি এই ইউনিটের মাধ্যমে সম্পাদিত হয়।

(খ) সকল তথ্য বা ডাটা ইনপুট ডিভাইসের মাধ্যমে গ্রহণ করে, তাহা CPU তে প্রক্রিয়াকরণের জন্য প্রেরণ করা হয়।

(গ) CPU ডাটা প্রক্রিয়াকরণ শেষে প্রয়োজনীয় আউটপুট তৈরি করে।

(ঘ) এতে গাণিতিক, যৌক্তিক কাজ এবং কন্ট্রোল প্রক্রিয়া সম্পূর্ণ হয়ে থাকে।

**(২) মেমোরি (Memory) :**

(ক) মাইক্রোকন্ট্রোলারের তিন ধরনের মেমোরি ব্যবহৃত হয়। যেমন : Read Only Memory (ROM), Random Access Memory (RAM), Electrically Programmable Read Only Memory (EPROM)।

(খ) সব ধরনের ডাটা বা তথ্য Memory তে সংরক্ষিত থাকে।

(গ) এখানে ROM এর মধ্যে নির্দিষ্ট কাজের Program স্থায়ীভাবে Install করা থাকে, যাহা শুধুমাত্র Read করা যায়।

**(৩) I/O পোর্ট (I/O Port) :**

(ক) মাইক্রোকন্ট্রোলারে 2 ধরনের port থাকে, একটি হলো input পোর্ট এবং অপরটি হলো output পোর্ট।

(খ) Input পোর্টের মাধ্যমে প্রয়োজনীয় তথ্য ইনপুট ডিভাইসে প্রদান করা হয় এবং Output পোর্টের মাধ্যমে প্রক্রিয়াকরণের পর প্রাপ্ত ফলাফলকে আউটপুট ডিভাইসে প্রদান করা হয়।

**(৪) টাইমিং এবং কন্ট্রোল ইউনিট (Timing and Control Unit) :**

(ক) এই ইউনিটের কাজ হলো মাইক্রোকন্ট্রোলারের অপারেশনের সাথে ক্লকের সমন্বয় সাধন করা।

(খ) এই ইউনিট CPU এবং বিভিন্ন পেরিফেরাল ডিভাইসের সাথে এবং মেমোরির সাথে যোগাযোগের জন্য প্রয়োজনীয় কন্ট্রোল Signal তৈরি করে।

(৫) **ইন্টারাপ্ট লজিক ইউনিট (Interrupt Logic Unit) :**

(ক) মাইক্রোকন্ট্রোলারের বিভিন্ন ধরনের Interrupt থাকে।

(খ) কোন ইন্সট্রাকশন আগে কাজ করবে বা কোন ইন্সট্রাকশন পরে কাজ করবে তা এই ইউনিটের মাধ্যমে নির্ধারণ হয়।

(গ) 8051 মাইক্রোকন্ট্রোলারে 6 টি ইন্টারাপ্ট লজিক রয়েছে।

(৬) **ADC এবং DAC :**

(ক) ADC মানে হলো Analog to Digital Converter এবং DAC মানে হলো Digital to Analog Converter.

(খ) এই ইউনিটগুলোর মাধ্যমে প্রয়োজনে Analog Signal কে Digital- এ এবং Digital Signal কে Analog Signal এ রূপান্তর করা হয়।

(৭) **অসিলেটর সার্কিট (Oscillator Circuit) :**

(ক) Oscillation Frequency তৈরি করতে এটি ব্যবহৃত হয়।

(খ) মাইক্রোকন্ট্রোলারে অসিলেটর হিসেবে সাধারণত কোয়ার্টজ ক্রিস্টাল অসিলেটর ব্যবহৃত হয়।

**২৭। মাইক্রোকন্ট্রোলারের মেমোরির বর্ণনা দাও।**

**উত্তর : মেমোরি (Memory) :**

(ক) মাইক্রোকন্ট্রোলারে তিন ধরনের মেমোরি ব্যবহৃত হয়। যেমন : Read Only Memory (ROM), Random Access Memory (RAM), Electrically Programmable Read Only Memory (EPROM)।

(খ) সবধরনের ডাটা বা তথ্য Memory তে সংরক্ষিত থাকে।

(গ) এখানে ROM এর মধ্যে নির্দিষ্ট কাজের Program স্থায়ীভাবে Install করা থাকে, যাহা শুধুমাত্র Read করা যায়।

**২৮। 8051 মাইক্রোকন্ট্রোলারের বিভিন্ন Family-এর মধ্যে পার্থক্য দেখাও।**

**উত্তর :** নিচের Table-এর মাধ্যমে 8051 family-এর মধ্যে পার্থক্য দেখানো হলো :

| Feature | 8051 | 8052 | 8031 |
|---|---|---|---|
| ROM (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 |

**২৯। বিভিন্ন ম্যানুফাকচারের তৈরি 8051 মাইক্রোকন্ট্রোলারের মধ্যে তুলনা কর।**

**উত্তর :** বিভিন্ন ধরনের ম্যানুফ্যাকচারার (Manufacturer) Company microcontroller তৈরি করে থাকে। কোম্পানিগুলো হলো–

Intel, Atmel, philips/Signaling, Infineon, Dallas semi/maxim ইত্যাদি।

1981 সালে Intel corporation 8-bit microcontrollers (যা 80051 নামে পরিচিত) বাজারজাত করে। এতে 4k bytes ROM, 128 bytes RAM, 2টি Timers, 32টি I/O Pins, 1 টি Serial port ও 6 টি Interrupt sources রয়েছে। 8051 মাইক্রোকন্ট্রোলার প্রচুর পরিমাণ চাহিদার কারণে Intel company অন্যান্য Manufacturer কে 8051 Microcontroller তৈরিতে উৎসাহ প্রদান করে। ফলে, বিভিন্ন Company বিভিন্ন Flavours এর Microcontroller বাজারে ছাড়ে। এর মধ্যে RAM, ROM, Speed, Packaging দাম ইত্যাদি পার্থক্য রয়েছে।

Philips কোম্পানি 8051 Family এর Microcontroller প্রস্তুত করে থাকে। এ Microcontroller বেশ কিছু ভিন্ন বৈশিষ্ট্য ধারণ করে; যেমন– A to D converter, D to A converter, extended I/O এবং OTP and flash.

Dallas Semicanclutor (Maxim). 8051 Microcontroller বাজারে ছেড়েছে। এখানে Flash Rom ব্যবহৃত হয়েছে।

Flash Rom ব্যবহারের কারণে উক্ত কোম্পানির 8051 microcontroller টি বেশ জনপ্রিয়তা পেয়েছে।

**৩০। ইমবেডেড সিস্টেম কী? উদাহরণের মাধ্যমে বর্ণনা কর।**

**উত্তর :** ইমবেডেড সিস্টেম (Embedded System) :

"ইমবেডেড সিস্টেম হলো এমন একধরনের সিস্টেম বা মেকানিক্যাল ডিভাইস যা শুধুমাত্র একটি নির্দিষ্ট কাজ করতে সক্ষম।"

**বৈশিষ্ট্য :** ইমবেডেড সিস্টেমের কিছু বৈশিষ্ট্য নিচে তুলে ধরা হলো :

(ক) এটি নির্দিষ্ট একটি কাজ করতে সক্ষম।

(খ) এটি এক ধরনের মেকানিক্যাল ডিভাইস।

(গ) এতে মাইক্রোপ্রসেসর ও মাইক্রোকন্ট্রোলার ব্যবহার করা হয়।

**উদাহরণ :** ইমবেডেড সিস্টেমের উদাহরণ নিচে উল্লেখ করা হলো :

**(i) Printer (প্রিন্টার) :** যা শুধুমাত্র Printer করতে পারে।

**(ii) ক্যালকুলেটর :** যা শুধুমাত্র গণনার কাজ করতে পারে।

**(iii) Automated Teller Machine :** যা থেকে শুধুমাত্র টাকা উত্তোলন করা যায়।

**(iv) Photocopier (ফটোকপিয়ার) :** যা শুধুমাত্র ফটোকপি করতে পারে।

**(v) এলার্ম সিস্টেম :** যা শুধুমাত্র এলার্ম সেট করার জন্য কাজ করে থাকে।

**৩১। মাইক্রোকন্ট্রোলারের চিত্রসহ বৈশিষ্ট্যগুলো উল্লেখ কর।**

**উত্তর :** মাইক্রোকন্ট্রোলার হলো একটি ইন্টিগ্রেটেড সার্কিট (Integrated Circuit) বা IC, যার মধ্যে একত্রে প্রসেসর (Processor), RAM (র‍্যাম), রম (ROM), I/O Port এবং টাইমার (Timer) যুক্ত থাকে। এটি নিজে নিজেই একটি স্বয়ংসম্পূর্ণ ডিভাইস। এটি মাইক্রোকম্পিউটারের চেয়ে সহজে ও দ্রুত কাজ করতে পারে। মাইক্রোকন্ট্রোলারকে আবার সিঙ্গল চিপ মাইক্রোকন্ট্রোলার অথবা ইমবেডেড কন্ট্রোলারও বলা হয়ে থাকে।

**মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য :** নিচে মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য উল্লেখ করা হলো :

(ক) এটি Highly Integrated Circuit বা চিপ।

(খ) এর মধ্যে Microprocessor, মেমোরি (ROM & RAM) I/O, Timer built-in থাকে।

(গ) এটি ছোট ও ওজনে হালকা।

(ঘ) এতে ডিজিটাল I/O ব্যবহৃত হয়।

(ঙ) এটি হার্ডওয়্যার ও সফ্টওয়্যার ইন্টারফেস বিশিষ্ট।

নিচে মাইক্রোকন্ট্রোলারের সাধারণ একটি Block চিত্র দেখানো হলো : যা CPU, RAM, ROM, I/O, Timer এবং Serial Com Port নিয়ে গঠিত।

| | | |
|---|---|---|
| CPU | RAM | ROM |
| I/O | Timer | Serial Comport |

**৩২। মাইক্রোকন্ট্রোলারের CPU এর বর্ণনা দাও।**

**উত্তর :** সেন্ট্রাল প্রসেসিং ইউনিট বা সিপিইউ (Central Processing Unit) :

(ক) সকল কার্যাদি এই ইউনিটের মাধ্যমে সম্পাদিত হয়।

(খ) সকল তথ্য বা ডাটা ইনপুট ডিভাইসের মাধ্যমে গ্রহণ করে, তা CPU তে প্রক্রিয়াকরণের জন্য প্রেরণ করা হয়।

(গ) CPU ডাটা প্রক্রিয়াকরণ শেষে প্রয়োজনীয় আউটপুট তৈরি করে।

(ঘ) এতে গাণিতিক, যৌক্তিক কাজ এবং কন্ট্রোল প্রক্রিয়া সম্পূর্ণ হয়ে থাকে।

**৩৩। ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**

**উত্তর :** ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(ক) এটি 8 বিটের মাইক্রোকন্ট্রোলার।

(খ) এটি 40 পিনের DIP প্যাকেজ।

(গ) এটি 8 বিটের 3 টি I/O পোর্ট বিদ্যমান।

(ঘ) এতে 1k byte এর ROM বিদ্যমান।

(ঙ) এতে 64 byte-এর RAM আছে।

(চ) এতে 2 টি Maskable (মাস্কেবল) ইন্টারাপ্ট রয়েছে।

(ছ) এতে Timer রয়েছে।

**৩৪। ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**

**উত্তর :** ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(ক) এটি 16 bit- এর মাইক্রোকন্ট্রোলার।

(খ) এতে 16 বিট এর CPU রয়েছে।

(গ) এতে 10 bit ADC (Anaiog to Digital Converter) বিদ্যমান।

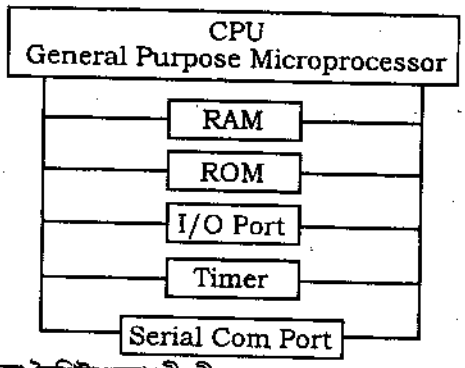
(ঘ) এই মাইক্রোকন্ট্রোলারে 16 বিট বিশিষ্ট 2 টি টাইমার বিদ্যমান।

(ঙ) এতে 8 বিটের 5 টি ইনপুট/আউটপুট পোর্ট (I/O Port) রয়েছে।

(চ) এটি Full-duplex Serial Port বিশিষ্ট।

**৩৫। মাইক্রোপ্রসেসরের Block ডায়াগ্রাম অঙ্কন কর।**

**উত্তর :** মাইক্রোপ্রসেসর এর Block ডায়াগ্রাম নিম্নরূপ :




**৩৬। MSC-51 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**

**উত্তর :** নিচে MSC-51 মাইক্রোকন্ট্রোলারের কিছু বৈশিষ্ট্য উল্লেখ করা হলো :

(ক) এটি 8 bit এর Microcontroller.

(খ) ইহাকে MCS-48 Microcontroller এর সাথে Replace করা যায়।

(গ) এটি মূলত HMOS (High Speed Metal Oxide Semiconductor) Technology তে তৈরি।

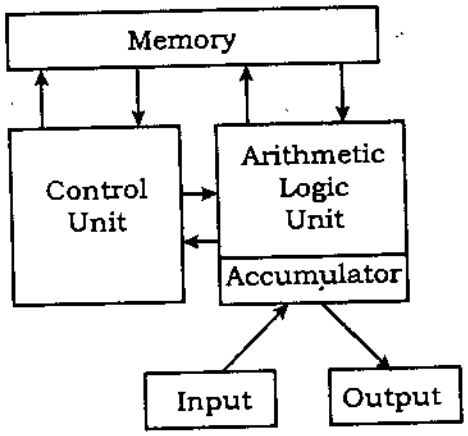
(ঘ) CHMOS (Complementary high Speed Metal Oxide Semiconductor) Version এরও মাইক্রোকন্ট্রোলার রয়েছে।

(ঙ) CHMOS Version এর মাইক্রোকন্ট্রোলার হলো Lower Power Consurption, Higher Noise Immunity এবং High Speed বিশিষ্ট।

(চ) CHMOS Version এর μc-এ Idle Mode থাকে।

**৩৭। Von-Neumann আর্কিটেকচার কাকে বলে? উদাহরণ দাও।**

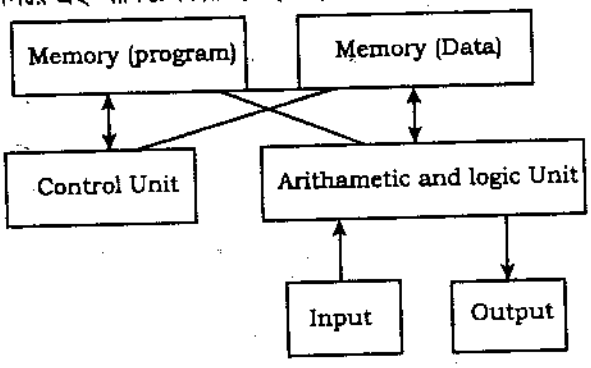
**উত্তর :** যে কম্পিউটার আর্কিটেকচারে প্রোগ্রাম ও ডাটা উভয়কেই সংরক্ষণের জন্য এতে একটি সাধারণ প্রধান মেমোরি ব্যবহৃত হয় তাকে Von-Neumann আর্কিটেকচার বলে। উদাহরণ : Motorola 68HC11, MCS-51 পরিবার ও PIC পরিবার এই আর্কিটেকচার ব্যবহার করে। চিত্রে Von-Neumann architecture দেখানো হলো।




চিত্র : Von-Neumann Architecture

**৩৮। হার্ভার্ড আর্কিটেকচার কাকে বলে? উদাহরণ দাও।**

**উত্তর :** যে কম্পিউটার আর্কিটেকচারে প্রোগ্রাম ও ডাটার জন্য দুটি ভিন্ন মেমোরি থাকে এবং এর জন্য দুটি ভিন্ন ডাটা বাস ও ইনস্ট্রাকশন বাস থাকে তাকে Harvard Architecture বলে। এতে সম্পাদন ক্রিয়া সমান্তরালে সংঘটিত হয়, অর্থাৎ নির্দেশনা পঠন ও ডাটা মেমোরি লিখন/পঠন একই সময়ে ঘটে, ফলে কম্পিউটার আর্কিটেকচার অনেক দ্রুতগতির হয়। Intel MCS-51 মাইক্রোকন্ট্রোলার পরিবার ও PIC মাইক্রোকন্ট্রোলার পরিবার ও AVR মাইক্রোকন্ট্রোলার পরিবার Harvard Architecture ব্যবহার করে। তাছাড়া অডিও/ভিডিও প্রসেসিং এ ডিজিটাল সিগন্যাল প্রসেসরে এই আর্কিটেকচার ব্যবহৃত হয়।




চিত্র : Harvard Architecture

## রচনামূলক প্রশ্নোত্তর

১। **মাইক্রোকন্ট্রোলার নির্বাচনে বিবেচ্য বিষয়গুলো উল্লেখ কর।** *[বাকাশিবো : '১৪, '১৪R]*

**উত্তর :** ১.৯ নং অনুচ্ছেদ দ্রষ্টব্য।

২। **একটি সাধারণ মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রামসহ এর বিভিন্ন ব্লকের বর্ণনা দাও।** *[বাকাশিবো : '১৪R]*
**অথবা, বেসিক মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন অংশের বর্ণনা দাও।** *[বাকাশিবো : '১৪]*

**উত্তর :** ১.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৩। **CISC ও RISC আর্কিটেকচার বর্ণনা কর।**

**উত্তর :** ১.৬ নং অনুচ্ছেদ দ্রষ্টব্য।

৪। **চিত্রসহ Von-Neumann আর্কিটেকচার বর্ণনা কর।**

**উত্তর :** ১.৫ নং অনুচ্ছেদ দ্রষ্টব্য।

৫। **চিত্রসহ Harvard আর্কিটেকচার বর্ণনা কর।**

**উত্তর :** ১.৫ নং অনুচ্ছেদ দ্রষ্টব্য।

৬। **মাইক্রোকন্ট্রোলার বৈশিষ্ট্য উল্লেখ কর।**

**উত্তর :** ১.৮ নং অনুচ্ছেদ দ্রষ্টব্য।

৭। **মাইক্রোকন্ট্রোলার নির্বাচনে শর্তসমূহ লিখ।**

**উত্তর :** ১.৯ নং অনুচ্ছেদ দ্রষ্টব্য।

৮। **মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের মধ্যে পার্থক্য লিখ।**

**উত্তর :** ১.৩ নং অনুচ্ছেদ দ্রষ্টব্য।

৯। **মাইক্রোকন্ট্রোলারের প্রকারভেদ উল্লেখ কর।**

**উত্তর :** ১.৪ নং অনুচ্ছেদ দ্রষ্টব্য।

১০। **কমপ্লেক্স ইনস্ট্রাকশন সেট কম্পিউটার (সিআইএসসি) এর বর্ণনা লিখ।**

**উত্তর :** ১.৬.১ নং অনুচ্ছেদ দ্রষ্টব্য।

১১। **রিডিউসড ইনস্ট্রাকশন সেট কম্পিউটার (আরআইএসসি) মাইক্রোকন্ট্রোলার বর্ণনা লিখ।**

**উত্তর :** ১.৬.২ নং অনুচ্ছেদ দ্রষ্টব্য।

১২। **মাইক্রোকন্ট্রোলারের প্রয়োগক্ষেত্র উল্লেখ কর।**

**উত্তর :** ১.৭ নং অনুচ্ছেদ দ্রষ্টব্য।

# অধ্যায় ২ ইন্টেল 8051 ও পিআইসি মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য

## Understand Features of the Intel 8051 Microcontroller and PIC

## ২.০ ভূমিকা

**(Introduction)**

1980 সালে (প্রকৃতপক্ষে 1997 সালের শেষের দিকে) MCS-51 পরিবারের প্রথম সদস্য হিসেবে 8051 মাইক্রোকন্ট্রোলারের প্রবর্তন হয় এবং প্রাথমিকভাবে এটি Embedded system এর জন্য নির্মিত হয়। এটি CISC instruction set ও Harvard আর্কিটেকচারের একটি একক চিপ কম্পিউটার। যদিও বা অপেক্ষাকৃত পুরানো তথাপি 8051 এখনো একটি জনপ্রিয় মাইক্রোকন্ট্রোলার এবং বিশেষ করে Embedded system এ এটি একটি বিরাট অংশ জুড়ে আছে। সমস্ত মাইক্রোকন্ট্রোলার বাজারের এটি প্রায় 40% দখল করে আছে। অনেক ডেরিভেটিভ (Derivative) মাইক্রোকন্ট্রোলার 8051 এর উপর ভিত্তি করে তৈরি। ইন্টেলের প্রথম MCS-51 পরিবার NMOS টেকনোলজিতে তৈরি। কিন্তু পরবর্তীতে CMOS টেকনোলজিতে (80C51) তৈরি হওয়ার এর পাওয়ার অপচয় অনেকাংশে কমে যায়। উদাহরণ : ইন্টেল 8031, ইন্টেল 8032, ইন্টেল 8051, ইন্টেল 8052, ইন্টেল 8054 ইত্যাদি।

এ অধ্যায়ে হতে শিক্ষার্থীরা নিচের বিষয় সম্পর্কে জানতে পারবে।

(১) MCS-51 পরিবার সম্পর্কে ধারণ লাভ।

(২) ইন্টেল 8051 মাইক্রোকন্ট্রোলার ও PIC মাইক্রোকন্ট্রোলরের বৈশিষ্ট্য জানতে পারবে।

(৩) 8051 পরিবারের বিভিন্ন সদস্যের বৈশিষ্ট্য জানতে পারবে।

## ২.১ MCS-51 পরিবার

**(Define 8051 Family)**

ইন্টেল কর্পোরেশন কর্তৃক MCS-51 মাইক্রোকন্ট্রোলার IC পরিবার উৎপাদন, উন্নয়ন ও বাজারজাত করা হয়। অন্যান্য IC নির্মাতা প্রতিষ্ঠান যথা– সিমেন্স, ফুজিৎসু, ফিলিপস, এটমেল ইত্যাদি MCS-51 পরিবারের ডিভাইসের দ্বিতীয় সরবরাহকারী উৎস হিসেবে লাইসেন্সপ্রাপ্ত।

## ২.২ ইন্টেল 8051 মাইক্রোকন্ট্রোলার ও PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য উল্লেখকরণ

**(Mention the Features of the Intel 8051 Microcontroller and PIC)**

### ২.২.১ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য

**(Features of the Intel 8051 microcontrollers)**

একটি আধুনিক ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য নিম্নরূপ :

(১) এতে বুলিয়ান প্রসেসর আছে এবং Bit-wise operation সমর্থন করে।

(২) এতে 16 বিটের 2টি টাইমার আছে।

(৩) এর বাহ্যিক প্রোগ্রাম মেমোরি 64 কিলোবাইট এবং ডাটা মেমোরি 64 কিলোবাইট (আধুনিক)।

(৪) এতে power saving mode আছে।

(৫) এর on-chip প্রোগ্রাম মেমোরি 4 কিলোবাইট ও ডাটা মেমোরি 128 বাইট।

(৬) এর 6টি ইন্টারপ্ট উৎস আছে।

(৭) এটি RISC ধরনের মাইক্রোকন্ট্রোলার।

(৮) এর সিরিয়াল পোর্ট (Full duplex UART) 1টি।

(৯) এটি Harvard Architecture এর অর্থাৎ প্রোগ্রাম মেমোরি ও ডাটা মেমোরি পৃথক।

(১০) এর অ্যাড্রেস বাসের আকার 16 বিট এবং Low order address Port 0 তে ডাটা বাসের সাথে multiplexed অবস্থায় থাকে।

(১১) এটি একটি 8 বিট মাইক্রোকন্ট্রোলার।

(১২) এতে 8 বিটের 4টি পোর্ট বা 32টি উভয়মুখী ও স্বতন্ত্রভাবে অ্যাড্রেসেবল ইনপুট/আউটপুট পিন আছে।

(১৩) এটি NMOS Technology তে তৈরি, তবে CMOS সংস্করণও পাওয়া যায়।

(১৪) এর ডাটা বাসের আকার 8 বিট (port 0)।

(১৫) এতে on-chip অসিলেটর আছে।

(১৬) এতে গুণন ও ভাগের নির্দেশনা আছে।

(১৭) এর সর্বোচ্চ অপারেটিং ফ্রিকোয়েন্সি (0-12) মেগাহার্টস।

### ২.২.২ PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য

**(Features of the PIC microcontrollers)**

Peripharel Interface Controller এর সংক্ষিপ্ত রূপ হলো PIC। এটি Microchip Technology এর তৈরি Modified Harvard architecture এর একটি মাইক্রোকন্ট্রোলার। General Instrument এর মাইক্রোইলেকট্রনিক্স বিভাগ প্রথম এটি তৈরি করে। কম খরচ, প্রাচুর্যতা, free development tools এবং Serial programming এর জন্য বর্তমানে এটি খুবই জনপ্রিয়। নিচে এর সাধারণ বৈশিষ্ট্য উল্লেখ করা হল :

(১) নির্দেশনা সংখ্যা খুব কম (16 C5XX মাইক্রোকন্ট্রোলারে মাত্র 33টি নির্দেশনা আছে)।

(২) এর আউটপুট পরিচালনা ক্ষমতা বেশি। এটি সরাসরি LED, TRIAC ইত্যাদিকে পরিচালনা করতে পারে।

(৩) সমস্ত নির্দেশনা 12 বা 14 বিট প্রোগ্রাম মেমোরিতে লাগসই।

(৪) এটি in-circuity serial program করতে পারে।

(৫) এটি Harvard আর্কিটেকচারের মাইক্রোকন্ট্রোলার।

(৬) PIC এর ক্লকের গতি তুলনামূলকভাবে অনেক বেশি। PIC16F877A এর ক্লক 20 মেগা হার্টস এবং এটি 30 হার্টসেও কাজ করতে পারে। PIC 18FX এর ক্লক 40 মেগা হার্টস।

(৭) এটি সম্পূর্ণভাবে একটি static মাইক্রোকন্ট্রোলার।

নিচে টেবিল আকারে PIC মাইক্রোকন্ট্রোলারের 4টি ক্যাটাগরির তুলনামূলক বৈশিষ্ট্য টেবিল আকারে উল্লেখ করা হলো :

| | **Base Line** | **Mid-Range** | **Enhanced Mid-Range** | **PIC18** |
|---|---|---|---|---|
| পিন সংখ্যা | 6–40 | 8–64 | 8–64 | 18–100 |
| প্রোগ্রাম মেমোরি | 3 কিলোবাইট অবধি | 14 কিলোবাইট অবধি | 28 কিলোবাইট অবধি | 128 কিলোবাইট অবধি |
| ডাটা মেমোরি | 134 বাইট অবধি | 368 বাইট অবধি | 1.5 কিলোবাইট অবধি | 4 কিলোবাইট অবধি |
| নির্দেশনার দৈর্ঘ্য | 12-bit | 14-bit | 14-bit | 16-bit |

৩৮৭১- বাহ: হম: মাংস: – ৪ (খ)

| | | | | |
|---|---|---|---|---|
| নির্দেশনা সেটের সংখ্যা | 33 | 35 | 49 | 83 |
| গতি | 5 MIPS* | 5 MIPS | 8 MIPS | 16 MIPS অবধি |
| বৈশিষ্ট্য | * Comparator<br>* 8-bit ADC<br>* ডাটা মেমোরি<br>* অভ্যন্তরীণ অসিলেটর | Baseline বৈশিষ্ট্যসহ<br>* SPI<br>* I²C<br>* UART<br>* PWM<br>* 10-bit ADC<br>* OP-Amps | Mid-range বৈশিষ্ট্যসহ<br>উচ্চ কর্মক্ষমতার বহুবিধ কমিউনিকেশন পেরিফেরাল ডিভাইস | Enhanced Mid-range সহ<br>* CAN<br>* LIN<br>* USB<br>* Ethernet<br>* 12-bit ADC |
| পরিবার | PIC10, PIC12, PIC16 | PIC12, PIC16 | PIC12F1XXX, PIC16F1XXX | PIC18 |

* MIPS—Millions Instructions Per Second

## ২.৩ 8051 পরিবারের বিভিন্ন সদস্যের বৈশিষ্ট্য তুলনা

## (Compare the Features of Different Member of the 8051 Family)

### ২.৩.১ MCS-51 এর প্রকারভেদ

### (Types of MCS-51)

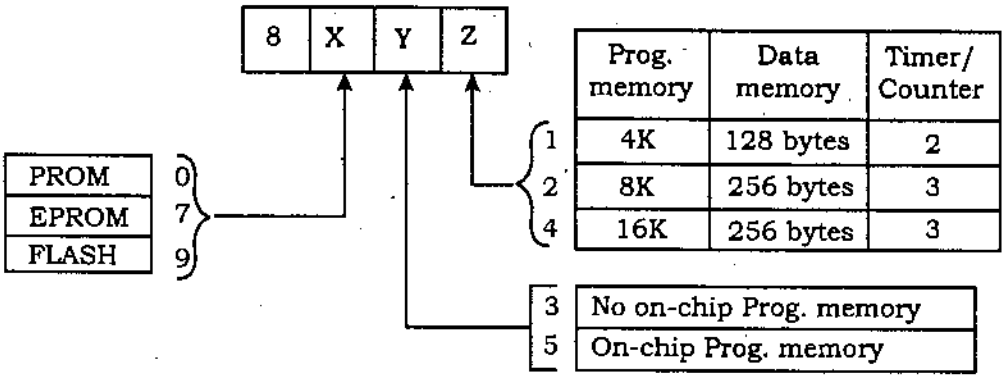
MCS-51 এর কোর বিভিন্ন প্রকারে পাওয়া যায়। নিচের MCS-51 পরিবারের কিছু সদস্যের বৈশিষ্ট্য টেবিল আকারে উল্লেখ করা হলো :

| পার্ট নং | DIP পিন সংখ্যা | On-chip প্রোগ্রাম মেমোরি | On-পযরঢ় ডাটা মেমোরি (বাইট) | I/O লাইন সংখ্যা | টাইমার/কাউন্টার সংখ্যা | Serial interface |
|---|---|---|---|---|---|---|
| 8031 | 40 | নেই | 128 | 32 | 2 | 1 |
| 8032 | 40 | নেই | 256 | 32 | 3 | 1 |
| 8051 | 40 | ROM 4K | 128 | 32 | 2 | 1 |
| 8052 | 40 | ROM 8K | 256 | 32 | 3 | 1 |
| 8054 | 40 | ROM 16 K | 256 | 32 | 3 | 1 |
| 8751 | 40 | EPROM 4K | 128 | 32 | 2 | 1 |
| 8752 | 40 | EPROM 8K | 256 | 32 | 3 | 1 |
| 8754 | 40 | EPROM 16K | 256 | 32 | 3 | 1 |
| 89C51 | 40 | FLASH 4K | 128 | 32 | 2 | 1 |
| 89C52 | 40 | FLASH 8K | 256 | 32 | 3 | 1 |
| 89C54 | 40 | FLASH 16K | 256 | 32 | 3 | 1 |
| 89C1051 | 20 | FLASH 1K | 64 | 15 | 1 | - |
| 89C2051 | 20 | FLASH 2K | 128 | 15 | 2 | 1 |
| 89C4051 | 20 | FLASH 4K | 128 | 15 | 2 | 1 |

টেবিল : MCS-51 পরিবারের বিভিন্ন সদস্যের তুলনামূলক পার্থক্য

## ২.৩.২ MCS-51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি

**(The Numbering System of the MCS-51 Microcontraller)**




চিত্র-২.১ : MCS-51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি

MCS-51 পরিবারের সদস্যদের নামকরণের একটি সুষ্ঠু নিয়ম আছে। যদি এটিকে 8 X Y Z দ্বারা প্রকাশ করা হয় তবে এর ডিজিটগুলো নিম্নোক্তভাবে প্রমাণ করা হয় :

(ক) বাম থেকে দ্বিতীয় সংখ্যা (X) প্রোগ্রাম মেমোরি প্রকাশ করে।

| | | |
|---|---|---|
| | 0 - - | PROM |
| X = | 7 - - | EPROM |
| | 9 - - | FLASH |

(খ) বাম থেকে তৃতীয় সংখ্যা (Y) প্রোগ্রাম মেমোরি প্রকাশ করে।

| | | |
|---|---|---|
| Y = | 3 - - | On-chip প্রোগ্রাম মেমোরি। |
| | 5 - - | প্রোগ্রাম মেমোরি On-chip নয়। |

(গ) শেষ বা ৪র্থ ডিজিট (Z) প্রোগ্রাম মেমোরি, ডাটা মেমোরি ও টাইমার/কাউন্টারের সংখ্যা প্রকাশ করে।

| ডিজিট | মান | প্রোগ্রাম মেমোরি | ডাটা মেমোরি | টাইমার/কাউন্টার |
|---|---|---|---|---|
| | 1 | 4K | 128 KB | 2 |
| Z = | 2 | 8K | 256 KB | 3 |
| | 4 | 16K | 256 KB | 3 |

# অনুশীলনী-০২

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **MCS-51 পরিবার কী?** *[বাকাশিবো : '১৪]*

**অথবা, 8051 পরিবার কাকে বলে?** *[বাকাশিবো : '১৪, '১৫]*

**উত্তর :** ইন্টেল কর্পোরেশন কর্তৃক MCS- 51 একটি মাইক্রোকন্ট্রোলার IC পরিবার উৎপাদন, উন্নয়ন ও বাজারজাত করা হয়। অন্যান্য IC নির্মাতা প্রতিষ্ঠান যথা– সিমেন্স, ফুজিৎসু, ফিলিপস, এটমেল ইত্যাদি MCS-51 পরিবারের ডিভাইসের দ্বিতীয় সরবরাহকারী উৎস হিসেবে লাইসেন্সপ্রাপ্ত।

২। **PIC কী?** *[বাকাশিবো : '১৪R]*

**উত্তর :** Peripharel Interface Controller কে সংক্ষেপে PIC বলে। এটি Microchip Technology এর তৈরি Modified Harvard architecture এর একটি মাইক্রোকন্ট্রোলার। General Instrument এর মাইক্রোইলেকট্রনিক্স বিভাগ প্রথম এটি তৈরি করে। কম খরচ, প্রাচুর্যতা, Free development tools এবং Serial programming এর জন্য বর্তমানে এটি খুবই জনপ্রিয়।

৩। **MCS-51 পরিবারের ৫টি মাইকোকন্ট্রোলারের নাম লিখ।** *[বাকাশিবো : '১৪R]*

**উত্তর :** ইন্টেল 8031, ইন্টেল 8032, ইন্টেল 8051, ইন্টেল 8052, ইন্টেল 8054 ইত্যাদি।

৪। **MCS-51 পরিবারের মাইক্রোকন্ট্রোলার প্রস্তুতকারী কয়েকটি প্রতিষ্ঠানের নাম লিখ।**

**উত্তর :** MCS-51 পরিবারের মাইক্রোকন্ট্রোলার প্রস্তুতকারী প্রতিষ্ঠান– সিমেন্স, ফুজিৎসু, ফিলিপস, এটমেল।

৫। **8051 মাইক্রোকন্ট্রোলারের পরিবারের 2 টি মাইক্রোকন্ট্রোলারের নাম কী?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের পরিবারের 2 টি সদস্য হলো – (ক) 8052, (খ) 8031।

৬। **8051 মাইক্রোকন্ট্রোলার কত বিটের?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলার 8-bit।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লিখ।** *[বাকাশিবো : '১৪]*

**উত্তর :** একটি আধুনিক ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য নিম্নরূপ :

(১) এতে বুলিয়ান প্রসেসর আছে এবং Bit-wise operation সমর্থন করে।

(২) এতে 16 বিটের 2টি টাইমার আছে।

(৩) এর বাহ্যিক প্রোগ্রাম মেমোরি 64 কিলোবাইট এবং ডাটা মেমোরি 64 কিলোবাইট (আধুনিক)।

(৪) এতে power saving mode আছে।

(৫) এর on-chip প্রোগ্রাম মেমোরি 4 কিলোবাইট ও ডাটা মেমোরি 128 বাইট।

(৬) এর 6টি ইন্টারপ্ট উৎস আছে।

(৭) এটি RISC ধরনের মাইক্রোকন্ট্রোলার।

(৮) এর সিরিয়াল পোর্ট (Full duplex UART) 1টি।

(৯) এটি Harvard Architecture এর অর্থাৎ প্রোগ্রাম মেমোরি ও ডাটা মেমোরি পৃথক।

(১০) এর অ্যাড্রেস বাসের আকার 16 বিট এবং Low order address Port 0 তে ডাটা বাসের সাথে multiplexed অবস্থায় থাকে।

(১১) এটি একটি 8 বিট মাইক্রোকন্ট্রোলার।

(১২) এতে 8 বিটের 4টি পোর্ট বা 32টি উভয়মুখী ও স্বতন্ত্রভাবে অ্যাড্রেসেবল ইনপুট/আউটপুট পিন আছে।

(১৩) এটি NMOS Technology তে তৈরি, তবে CMOS সংস্করণও পাওয়া যায়।

(১৪) এর ডাটা বাসের আকার 8 বিট (port 0)।

(১৫) এতে on-chip অসিলেটর আছে।

(১৬) এতে গুণন ও ভাগের নির্দেশনা আছে।

(১৭) এর সর্বোচ্চ অপারেটিং ফ্রিকোয়েন্সি (0–12) মেগাহার্টস।

২। **PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লিখ।** *[বাকাশিবো : '১৪R]*

**উত্তর :** Peripharel Interface Controller এর সংক্ষিপ্ত রূপ হলো PIC। এটি Microchip Technology এর তৈরি Modified Harvard architecture এর একটি মাইক্রোকন্ট্রোলার। General Instrument এর মাইক্রোইলেকট্রনিক্স বিভাগ প্রথম এটি তৈরি করে। কম খরচ, প্রাচুর্যতা, free development tools এবং Serial programming এর জন্য বর্তমানে এটি খুবই জনপ্রিয়। নিচে এর সাধারণ বৈশিষ্ট্য উল্লেখ করা হলো :

(১) নির্দেশনা সংখ্যা খুব কম (16 C5XX মাইক্রোকন্ট্রোলারে মাত্র 33টি নির্দেশনা আছে)।

(২) এর আউটপুট পরিচালনা ক্ষমতা বেশি। এটি সরাসরি LED, TRIAC ইত্যাদিকে পরিচালনা করতে পারে।

(৩) সমস্ত নির্দেশনা 12 বা 14 বিট প্রোগ্রাম মেমোরিতে লাগসই।

(৪) এটি In-circuity serial program করতে পারে।

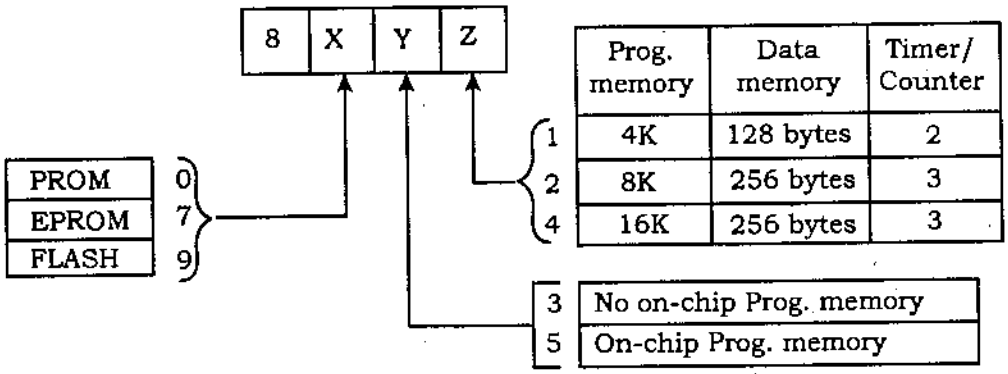
(৫) এটি Harvard আর্কিটেকচারের মাইক্রোকন্ট্রোলার।

(৬) PIC এর ক্লকের গতি তুলনামূলকভাবে অনেক বেশি। PIC16F877A এর ক্লক 20 মে. হার্টস এবং এটি 30 হার্টসেও কাজ করতে পারে। PIC 18FX এর ক্লক 40 মে. হার্টস।

(৭) এটি সম্পূর্ণভাবে একটি static মাইক্রোকন্ট্রোলার।

৩। **MCS-51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি উল্লেখ কর।** *[বাকাশিবো : '১৪R]*

**উত্তর :** MCS- 51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি নিচে উল্লেখ করা হলো:




**চিত্র :** MCS-51 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি

৪। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?** *[বাকাশিবো : '১০]*

**উত্তর :** ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(১) এটি 40 পিনের DIP প্যাকেজ।

(২) এটিতে On Chip Oscillator ও Clock সার্কিট থাকে।

(৩) এতে 8 টি বিটের CPU থাকে।

(৪) CPU একই সময়ে 8 বিট ডাটা নিয়ে কাজ করে।

(৫) 8 বিটের 4 (চার) টি I/O পোর্ট থাকে।
(৬) এতে সিরিয়াল পোর্ট বিদ্যমান।
(৭) এতে 6 টি Interrupt সুবিধা থাকে।
(৮) এতে 2 টি 16 বিটের টাইমার থাকে।
(৯) এতে 4k Byte এর ROM থাকে।
(১০) এতে 128 byte এর RAM বিদ্যমান।

**৫। 8051-এর মূল বৈশিষ্ট্য লিখ।** *[বাকাশিবো : '১০]*

**উত্তর :** 8051 family-এর প্রধান ও মূল সদস্য হলো 8051 ইন্টেল কোম্পানি একে mcs-51 নামেও ব্যবহার করে। এর মূল বৈশিষ্ট্যগুলো হলো :

| | |
|---|---|
| ROM– | 4K bytes |
| RAM– | 128 bytes |
| Tims– | 2 NOS. |
| I/O Ports– | 32 NOS. |
| Serial Port– | 1 |
| Interrupt sources– | 6 |

এ ছাড়াও 8052 এবং 8031 নামে দুটি সদস্য (Members) আছে।

**৬। 8051 মাইক্রোকন্ট্রোলারের বিভিন্ন Family-এর মধ্যে পার্থক্য দেখাও।**

**উত্তর :** নিচের Table-এর মাধ্যমে 8051 family-এর মধ্যে পার্থক্য দেখানো হলো :

| Feature | 8051 | 8052 | 8031 |
|---|---|---|---|
| ROM (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 |

**৭। বিভিন্ন ম্যানুফাকচারের তৈরি 8051 মাইক্রোকন্ট্রোলারের মধ্যে তুলনা কর।**

**উত্তর :** বিভিন্ন ধরনের ম্যানুফ্যাকচারার (Manufacturer) Company microcontroller তৈরি করে থাকে। কোম্পানিগুলো হলো–

Intel, Atmel, philips/Signaling, Infineon, Dallas semi/maxim ইত্যাদি।

1981 সালে Intel corporation 8-bit microcontrollers (যা 80051 নামে পরিচিত) বাজারজাত করে। এতে 4k bytes ROM, 128 bytes RAM, 2টি Timers, 32টি I/O Pins, 1 টি Serial port ও 6 টি Interrupt sources রয়েছে। 8051 মাইক্রোকন্ট্রোলার প্রচুর পরিমাণ চাহিদার কারণে Intel company অন্যান্য Manufacturer কে 8051 Microcontroller তৈরিতে উৎসাহ প্রদান করে। ফলে, বিভিন্ন Company বিভিন্ন Flavours এর Microcontroller বাজারে ছাড়ে। এর মধ্যে RAM, ROM, Speed, Packaging দাম ইত্যাদি পার্থক্য রয়েছে।

Philips কোম্পানি 8051 Family এর Microcontroller প্রস্তুত করে থাকে। এ Microcontroller বেশ কিছু ভিন্ন বৈশিষ্ট্য ধারণ করে; যেমন– A to D converter, D to A converter, extended I/O এবং OTP and flash.

Dallas Semicanclutor (Maxim). 8051 Microcontroller বাজারে ছেড়েছে। এখানে Flash Rom ব্যবহৃত হয়েছে।

Flash Rom ব্যবহারের কারণে উক্ত কোম্পানির 8051 microcontroller টি বেশ জনপ্রিয়তা পেয়েছে।

**৮। ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**

**উত্তর :** ইন্টেল 8048 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(ক) এটি 8 বিটের মাইক্রোকন্ট্রোলার।

(খ) এটি 40 পিনের DIP প্যাকেজ।

(গ) এটি 8 বিটের 3 টি I/O পোর্ট বিদ্যমান।

(ঘ) এতে 1k byte এর ROM বিদ্যমান।

(ঙ) এতে 64 byte-এর RAM আছে।

(চ) এতে 2 টি Maskable (মাস্কেবল) ইন্টারাপ্ট রয়েছে।

(ছ) এতে Timer রয়েছে।

**৯। ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**

**উত্তর :** ইন্টেল 8096 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো নিম্নরূপ :

(ক) এটি 16 bit- এর মাইক্রোকন্ট্রোলার।

(খ) এতে 16 বিট এর CPU রয়েছে।

(গ) এতে 10 bit ADC (Anaiog to Digital Converter) বিদ্যমান।

(ঘ) এই মাইক্রোকন্ট্রোলারে 16 বিট বিশিষ্ট 2 টি টাইমার বিদ্যমান।

(ঙ) এতে 8 বিটের 5 টি ইনপুট/আউটপুট পোর্ট (I/O Port) রয়েছে।

(চ) এটি Full-duplex Serial Port বিশিষ্ট।

## রচনামূলক প্রশ্নোত্তর

**১। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট লিখ।**

**উত্তর :** অনুচ্ছেদ ২.২.১ নং দ্রষ্টব্য।

**২। MCS-51 মাইক্রোকন্ট্রোলারের নামারিং পদ্ধতি বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ২.৩.২ নং দ্রষ্টব্য।

**৩। PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লেখ।**

**উত্তর :** অনুচ্ছেদ ২.২.২ নং দ্রষ্টব্য।

# অধ্যায় ৩ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের গঠন

## Understand Architecture of the Intel 8051 Microcontroller

### ৩.০ ভূমিকা

**(Introduction)**

ইনটেল কোম্পানি (Intel Company) বা Corporation 1981 সালে প্রথম 8 bit মাইক্রোকন্ট্রোলার তৈরি করে যাকে 8051 নামে অভিহিত করা হয়। 8051 মাইক্রোকন্ট্রোলারের প্রচণ্ড জনপ্রিয় (Popular) হবার পর Intel Company অন্যান্য Manufacturers কে 8051 মাইক্রোকন্ট্রোলার তৈরির অনুমতি প্রদান করে।

বিভিন্ন ধরনের ম্যানুফ্যাকচারিং (Manufacturer) কোম্পানি মাইক্রোকন্ট্রোলার তৈরি করে থাকে। কোম্পানিগুলো হলো :

(i) ফিলিপস/সিগনিলাইস (Philips/Signelies),

(ii) ইন্টেল (Intel),

(iii) ইনফাইনিওন (Infineon),

(iv) অ্যাটমেল (Atmel),

(v) ডালাস সেমি/ম্যাক্সিম (Dallas semi/maxim) ইত্যাদি।

**ইনটেল কোম্পানি (Intel Company) :** 1981 সালে Intel Corporation 8-bit microcontroller (যা 8051 নামে পরিচিত) বাজারজাত করে। এটি নিচের বৈশিষ্ট্যগুলো ধারণ করে :

(i) 128 bytes RAM

(ii) 4k bytes ROM

(iii) 32টি I/O Pins

(iv) 2টি Timers

(v) 1টি সিরিয়াল পোর্ট

(vi) 6টি Interrupt Source রয়েছে।

8051 মাইক্রোকন্ট্রোলারের প্রচুর পরিমাণ চাহিদার কারণে Intel company অন্যান্য Manufacturer-কে 8051 Microcontroller তৈরিতে উৎসাহ প্রদান করে। ফলে, বিভিন্ন Company বিভিন্ন Flavours-এর Microntroller বাজারে ছাড়ে। এর মধ্যে RAM, ROM, Speed, Packaging, দাম ইত্যাদির পার্থক্য আছে।

**ফিলিপস কোম্পানি (Philips Company) :** Philips কোম্পানি 8051 Family-এর Microcontroller তৈরি করে থাকে। এ Microcontroller বেশি কিছু ভিন্ন বৈশিষ্ট্য ধারণ করে, যেমন– (1) A to D converter, (2) D to A converter, (3) extended I/O, (4) OTP and flash.

**ডালাস কোম্পানি (Dallas Company) :** Dallas Semiconductor (Maxim), 8051 Microcontroller বাজারে ছেড়েছে। এখানে Flash ROM ব্যবহৃত হয়েছে।

Flash ROM ব্যবহারের কারণে উক্ত কোম্পানির 8051 microcontroller টি বেশ জনপ্রিয়তা আছে।

**অ্যাটমেল কোম্পানি (Atmel Company) :** অ্যাটমেল কোম্পানি দুই ধরনের মাইক্রোকন্ট্রোলার তৈরি করে থাকে। এক ধরনের মাইক্রোকন্ট্রোলার হলো ROM flash জাতীয়, অন্য ধরনের মাইক্রোকন্ট্রোলার হলো স্পিড ডিপেন্ডেন্ট।

নিচের টেবিলের সাহায্যে বিভিন্ন Manufacturer কর্তৃক তৈরি Microcontroller-এর মধ্যে তুলনা দেখানো হয়েছে :

**টেবিল-1.1 :** Versions of 8051 From Atmel (All ROM Flash)

| Part Number | ROM | RAM | I/O Pins | Timer | Interrupt | $V_{cc}$ | Packaging |
|---|---|---|---|---|---|---|---|
| AT89C51 | 4K | 128 | 32 | 2 | 6 | 5V | 40 |
| AT89LV51 | 4K | 128 | 32 | 2 | 6 | 3V | 40 |
| AT89C1051 | 1K | 64 | 15 | 1 | 3 | 3V | 20 |
| AT89C2051 | 2K | 128 | 15 | 2 | 6 | 3V | 20 |
| AT89C52 | 8K | 128 | 32 | 3 | 8 | 5V | 40 |
| AT89LV52 | 8K | 128 | 32 | 3 | 8 | 3V | 40 |

**Note :** "C" in the part number indicates CMOS :

**টেবিল-1.2 :** Versions of 8051/52 Microcontroller From Dallas Semiconductor (Maxim)

| Part Number | ROM | RAM | I/O pins | Timers | Interrupts | $V_{cc}$ |
|---|---|---|---|---|---|---|
| DS89C420/30 | 16K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS89C440 | 32K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS89C450 | 64K (Flash) | 256 | 32 | 3 | 6 | 5V |
| DS5000 | 8K (NVRAM) | 128 | 32 | 2 | 6 | 5V |
| DS80C320 | 0K | 256 | 32 | 3 | 6 | 5V |
| DS87520 | 16K (UVROM) | 256 | 32 | 3 | 6 | 5V |

**টেবিল- 1.3 :** Various Speeds of 8051 From Atmel

| Part Number | Speed | Pins | Packaging | Use |
|---|---|---|---|---|
| AT89C51–12PC | 12 MHz | 40 | DIP plastic | Commercial |
| AT89C51–16PC | 16 MHz | 40 | DIP plastic | Commercial |
| AT89C51–20PC | 20 MHz | 40 | DIP plastic | Commercial |

এ অধ্যায় হতে শিক্ষার্থীরা নিচের বিষয় সম্পর্কে জানতে পারবে।

(১) ইন্টেল 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম সম্পর্কে জানতে পারবে।

(২) 8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল সম্পর্কে ধারণা।

(৩) 8051 মাইক্রোকন্ট্রোলারের মেমোরি গঠন সম্পর্কে ধারণা।

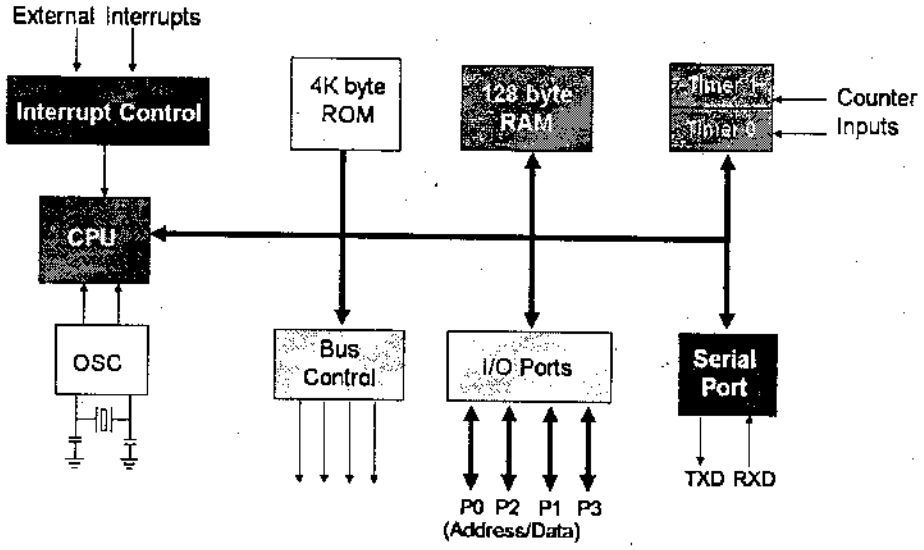
(৪) প্রোগ্রাম স্ট্যাস্টাস ও থার্ড রেজিস্টার সম্পর্কে ধারণা।

(৫) 8051 মাইক্রোকন্ট্রোলারের পিন ও সিজন্যাল ডায়াগ্রাম সম্পর্কে ধারণা।

## ৩.১ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম বর্ণনা (Describe the Simplified Block Diagram of the Intel 8051 Microcontroller)

যে মাইক্রোকন্ট্রোলার একসাথে 8 বিটের ডাটা লিখন/পঠন (Read/Write) ও প্রক্রিয়া করতে পারে সেটি হলো ইন্টেল 8051 এর একটি 8 বিট মাইক্রোকন্ট্রোলার। সাধারণত এর গাঠনিক ভিত্তি হলো হার্ভার্ড আর্কিটেকচার অর্থাৎ প্রোগ্রাম ও ডাটা মেমোরি পৃথক এবং প্রসেসরের ধরন RISC (Reduced Instruction Set Computer)। অভ্যন্তরীণ মেমোরির আকার MCS-family এর বিভিন্ন মাইক্রোকন্ট্রোলারের ক্ষেত্রে বিভিন্ন রকম হয়। 8051 মাইক্রোকন্ট্রোলারের সচরাচর ব্যবহৃত ব্লক ডায়াগ্রাম চিত্র নিচে এ দেখান হলো।

নিচে এর ব্লকগুলোর সংক্ষিপ্ত বিবরণ দেয়া হলো :




চিত্র-৩.১ : 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম

**সিপিইউ (CPU-Central Processor Unit) :**

CPU বা কেন্দ্রীয় প্রক্রিয়াকরণ অংশ হলো একটি মাইক্রোকন্ট্রোলারের মস্তিষ্ক। মাইক্রোকন্ট্রোলারের প্রায়ই সমস্ত কার্যক্রম এতে সম্পন্ন হয়। এটা ক্লক পালস পরিচালিত অংশ। ক্লক পালসের সাথে সবকিছু ধাপে ধাপে সংগঠিত হয়। এই ইউনিট অভ্যন্তরীণ টাইমিং ছাড়াও কিছু কন্ট্রোল সিগন্যাল উৎপন্ন হয়।

এটি মেশিনের বা প্রসেসের সমস্ত প্রক্রিয়াকে ব্যবস্থাপনা, পরীক্ষা-নিরীক্ষা এবং নিয়ন্ত্রণ করে। সিপিইউ প্রাথমিকভাবে মেমোরি হতে নির্দেশনা আনয়ন করে (fetch) এবং বাইনারি কোডে রূপান্তরপূর্বক নির্দেশনা সম্পাদন শেষে ফলাফল উৎপন্ন করে। 8051 মাইক্রোকন্ট্রোলারের প্রসেসিং অংশ 8 বিটের। এর ALU এবং প্রসেসিং ব্লকের গুরুত্বপূর্ণ রেজিস্টার অ্যাকুমুলেটর (Accumulator) বা A রেজিস্টার ও B রেজিস্টার 8 বিটের। ALU এক মেশিন সাইকেলে 8 বিটের গাণিতিক ও যৌক্তিক নির্দেশনা সম্পাদন করে। 8051 এ বুলিয়ান প্রসেসর সন্নিহিত থাকে, তাই এটি bit-wise অপারেশন করতে পারে।

**অসিলেটর (Oscillator) :**

8051 মাইক্রোকন্ট্রোলারের প্রাণ হলো Clock pulse উৎপাদক বা অসিলেটর, যা অভ্যন্তরীণ সমস্ত কার্যাবলির সমকালীনতা (Synchronous) রক্ষা করে। 8051 মাইক্রোকন্ট্রোলারে রেজোনেন্ট নেটওয়ার্কের সাথে সংযোগ করে একটি অসিলেটর তৈরির জন্য XTAL1 ও XTAL2 দুটি পিন আছে। সাধারণত অসিলেটরের জন্য একটি কোয়ার্টস ক্রিস্টাল ও ক্যাপাসিটর ব্যবহার করা হয়, তবে TTL clock source-ও ব্যবহৃত হয়। এই অসিলেটর মাইক্রোকন্ট্রোলারের গতি ও baud rate নির্ধারণ করে। ক্রিস্টাল ফ্রিকোয়েন্সির রেঞ্জ 4 মে. হার্টস – 30 মে. হার্টস হয় এবং অধিকাংশ ক্ষেত্রে সাধারণত 11.0592 মে. হার্টসের ক্রিস্টাল ব্যবহার করা হয়।

**অন চিপ র‍্যাম ডাটা মেমোরি (On-Chip RAM (Data Memory) :**

প্রোগ্রাম চলাকালীন সাময়িকভাবে ডাটা সংরক্ষণের জন্য উদ্বায়ী RAM ব্যবহার করা হয়। RAM অভ্যন্তরীণ (On-Chip) ও অনাভ্যন্তরীণ (External) উভয়ই হতে পারে। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ RAM এর আকার 128 বাইট। অভ্যন্তরীণ RAM এর কিছু অংশ (রেজিস্টার) পেরিফেরালের (যেমন– টাইমার/কাউন্টার, সিরিয়াল পোর্ট, ইন্টারাপ্ট ইত্যাদি) নিয়ন্ত্রণের জন্য এবং কিছু অংশ (Register bank ও Stack) সাময়িক তথ্য ধারণের জন্য ব্যবহৃত হয়। পেরিফেরালের নিয়ন্ত্রণের জন্য ব্যবহৃত রেজিস্টরসমূহকে Special Function Registers (SFRs) বলে। প্রকৃতপক্ষে ইন্টেল 8051 মাইক্রোকন্ট্রোলারের RAM পরিমাণ 256 কিলোবাইট যার 128 ব্যবহারকারীর জন্য এবং বাকি 128 কিলোবাইট SFRs এর জন্য বরাদ্দ থাকে।

অধিকাংশ 8051 মাইক্রোকন্ট্রোলার বাহ্যিক off-chip memory ক্ষেত্রকে access করতে পারে। Off-chip data RAM কে Access করার জন্য 16 বিটের address ব্যবহৃত হয়। Address Port, P0 এবং address-data Port, P2 একত্রে এই 16 বিট address ধারণ করে। Low order address (byte) ডাটা বাসের সাথে multiplex অবস্থায় থাকে।

**ইন্টারপ্টস (Interrupts) :**

মাইক্রোকন্ট্রোলারের প্রধান কাজ হলো ইন্টারাপ্ট একটি সাব-রুটিন কল (Sub-routine call) করা এবং ঐ সময়ের গুরুত্বপূর্ণ অন্য প্রোগ্রাম সম্পাদনে সহায়তা করে। এর কাজ অত্যন্ত গঠনমূলক কারণ জরুরি কাজে এটা মাইক্রোকন্ট্রোলারকে সহায়তা করে। ইন্টারাপ্ট রুটিন সম্পাদনকালীন চলমান প্রক্রিয়া স্থগিত হয় এবং ইন্টারাপ্ট সমাপ্তনান্তে চলমান কাজ আবার শুরু করে। এই বিশেষ ইন্টারাপ্ট সাব-রুটিনকে Interrupt Service Routine (ISR) বা Interrupt handler বলে। বাহ্যিক ইন্টারাপ্ট এর জন্য 8051 মাইক্রোকন্ট্রোলারের দুটি পিন INT1 ও INT2 আছে। এতে ইন্টারাপ্ট সেটিং এর জন্য বিশেষ রেজিস্টার (IE) আছে।

**টাইমার এবং কাউন্টার (Timers and Counters) :**

8051 মাইক্রোকন্ট্রোলারে 2টি 16 বিটের টাইমার/কাউন্টার আছে। এরা 8 বিটের 8টি রেজিস্টারে বিভক্ত। টাইমার দুটি ইভেন্টের মধ্যকার সময় পরিমাপ ও কালক্ষেপণ (delay) পরিমাপ, পালস গণনা, পালসের প্রশস্ততা (pulse width) পরিমাপ এবং baud rate উৎপাদন করে। এই সমস্ত কার্যকরী অ্যাসেম্বলি প্রোগ্রামের মাধ্যমে করা যায়, তবে এই 8051 মাইক্রোকন্ট্রোলারে delay উৎপাদনের জন্য দুটি হার্ডওয়্যার পিন (T1 ও T2) আছে। এই পিন দুটি বাহ্যিক ইভেন্ট গণনা করার জন্যও ব্যবহৃত হয়। এ জন্য এটা দুটি মুডে কাজ করে– টাইমার মুড ও কাউন্টার মুড। 8051 মাইক্রোকন্ট্রোলারে টাইমার/কাউন্টার অপারেশনের জন্য বিশেষ রেজিস্টারসমূহ : TCON, TMOD, TL1, TH1, TL0, TH0। টাইমার অপারেশন 4টি মুডে পরিচালিত হয়।

**সিরিয়াল পোর্ট (Serial Port) :**

8051 মাইক্রোকন্ট্রোলারের সিরিয়াল কমিউনিকেশনের জন্য TXD ও RXD দুটি পিন আছে। সাধারণত একটি বিশেষ রেজিস্টারে রক্ষিত সিরিয়াল ডাটা সম্প্রচারের জন্য TXD পিন এবং সিরিয়াল ডাটা গ্রহণের জন্য RXD পিন ব্যবহৃত হয়। সিরিয়াল কমিউনিকেশন অপারেশন নিয়ন্ত্রণের জন্য বিশেষ রেজিস্টার SCON ব্যবহৃত হয়। সিরিয়াল কমিউনিকেশন অপারেশন 4টি মুডে পরিচালিত হয়।

**ইনপুট আউটপুট পোর্ট (Input Output Ports) :**

মেশিন, ডিভাইস এবং পেরিফেরালের সাথে সংযোগের (Interfacing) জন্য 8051 এর 8 বিট প্রশস্ততার 4টি পোর্ট আছে। পোর্টগুলো হলো : Port0, Port1, Port2, Port3। প্রত্যেক পোর্টের একটি করে মোট চারটি স্বতন্ত্র রেজিস্টার (P0, P1, P2 ও P3) আছে। রেজিস্টারগুলো Bit addressable অর্থাৎ বিট নির্দেশনা (SETB, CLR) দ্বারা এদেরকে সেট/রিসেট করা যায়। পোর্টের ডাটা এই রেজিস্টারে জমা থাকে।

PORT 0 দ্বৈত কাজ করে। এটা 8 বিটের ডাটার সাথে Low Order Address-ও ধারণ করে (AD0-AD7)। PORT1 ইনপুট/আউটপুট পোর্ট। PORT2 ইনপুট/আউটপুট পোর্ট এবং এটা High Order Address ধারণ করে। তাছাড়া এর প্রত্যেক পিন একটি বিশেষ কাজের জন্য স্বতন্ত্রভাবে কাজ করে।

**বাস (Bus) :**

8 বিটের ডাটা বাস, 16 বিটের অ্যাড্রেস বাস এবং বাস কন্ট্রোল সিগন্যাল নিয়ে 8051 মাইক্রোকন্ট্রোলার সিস্টেম বাস গঠিত। ডাটা ট্রান্সফারের জন্য কমিউনিকেশন মাধ্যম বা চ্যানেল হিসেবে ব্যবহৃত একটি গুচ্ছ তার হলো মূলত বাস। বাসসমূহ 8, 16 বা ততোধিক তার নিয়ে গঠিত। এতে বাস একসাথে 8 বিট, 16 বিট ডাটা একত্রে বহন করতে পারে। বাস দুই ধরনের :

১। **ডাটা বাস (Data Bus) :** 8051 মাইক্রোকন্ট্রোলারের ডাটা বাস 8 বিট এবং এটা ডাটা বহন করার জন্য ব্যবহার করা হয়।

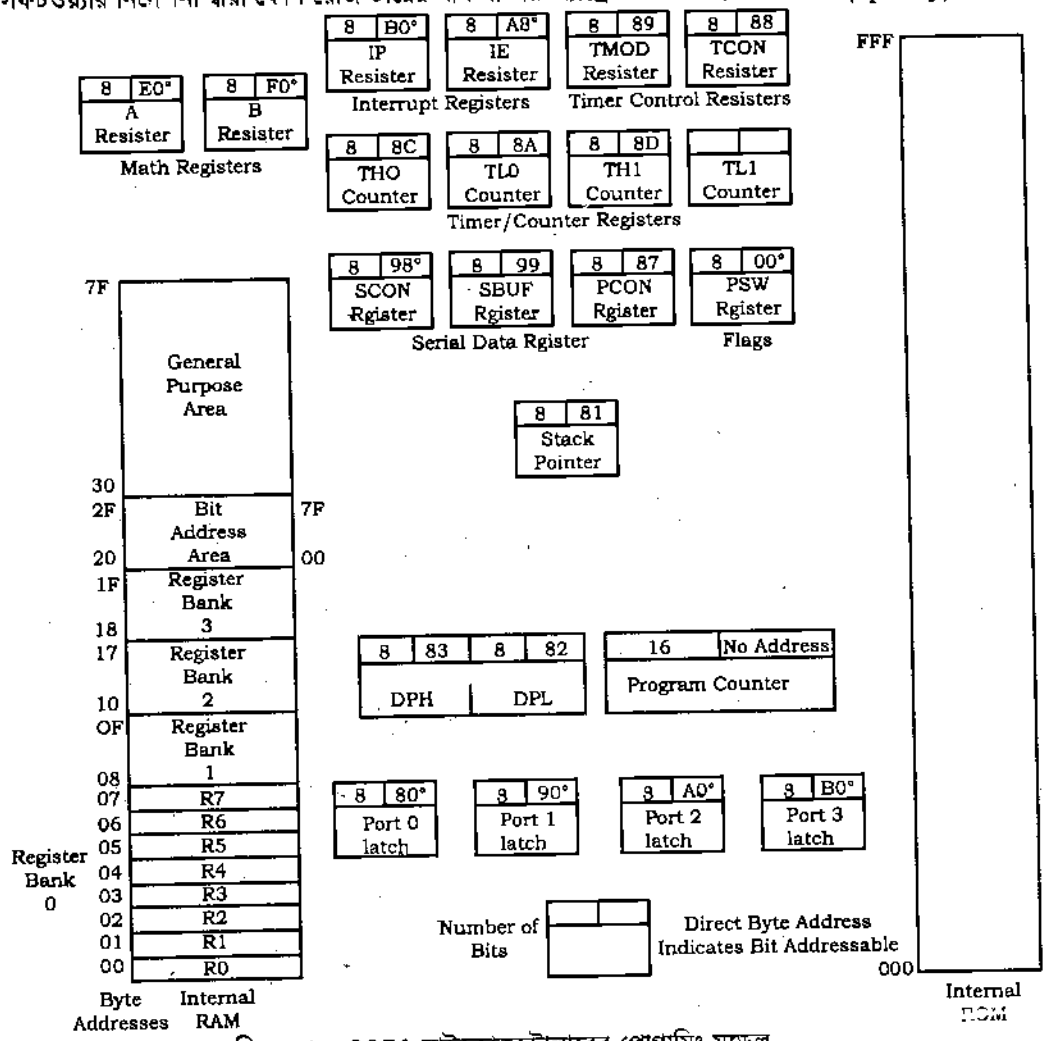
২। **অ্যাড্রেস বাস (Address Bus) :** 8051 মাইক্রোকন্ট্রোলারের অ্যাড্রেস বাস 16 বিট। এটা মেমোরি লোকেশনকে অ্যাড্রেস করার জন্য ব্যবহৃত হয়। এটা CPU থেকে মেমোরিতে অ্যাড্রেস পাঠানোর জন্য ব্যবহার করা হয়।

## ৩.২ 8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল
## (Explain the Programming Model of the 8051 Microcontroller)

8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল নিচের চিত্রে দেখান হলো। এটা 8 বিট ও 16 বিটের রেজিস্টার এবং 8 বিট মেমোরি লোকেশনের একটি পুঞ্জী (collection)। এই রেজিস্টার ও মেমোরি লোকেশনসমূহ সফটওয়্যার নির্দেশনা দ্বারা পরিচালিত যা ডিজাইনের অংশ হিসেবে একত্রীভূত। প্রোগ্রামের নির্দেশনাসমূহকে 8051 এর অভ্যন্তরীণ রেজিস্টার, ডিজিটাল ডাটা পথ এবং বাইরের মেমোরি লোকেশনকে নিয়ন্ত্রণ করতে হয়।

মাইক্রোকম্পিউটারকে মাইক্রোকন্ট্রোলারে রূপান্তরের জন্য আবশ্যিকভাবে প্রয়োজন কিছু বিশেষ রেজিস্টার (SFRs)। 8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেলকে জটিল করে। অধিকাংশ রেজিস্টারই একটি বিশেষ কাজ করে। তাই এরা একটি সাংকেতিক নাম নিয়ে একটি স্বতন্ত্র ব্লক ধারণ করে। যেমন– A, THO ইত্যাদি। আবার কিছু কিছু রেজিস্টারকে কার্যকারিতায় সাধারণত পারস্পরিক পৃথিকীকরণ করা যায় না, তাদেরকে একটি গ্রুপে নামাংকিত করা হয়, যেমন– RAM, ROM ইত্যাদি।

প্রত্যেক রেজিস্টার (PC ছাড়া) এর একটি 1 বাইটের অভ্যন্তরীণ অ্যাড্রেস থাকে যা এটাকে ধার্য (assign) করে। আবার কিছু কিছু রেজিস্টার (* চিহ্নিত) বিট ও বাইট উভয়ই দ্বারা অ্যাড্রেস করা যায়। অর্থাৎ এ ধরনের রেজিস্টার অ্যাড্রেসের ডাটার সম্পূর্ণ বাইট পঠন বা পরিবর্তন করা যায় অথবা স্বতন্ত্র বিটগুলো পঠন বা পরিবর্তন করা যায়। সফটওয়্যার নির্দেশনা দ্বারা কোন রেজিস্টারের নাম বা এর অ্যাড্রেস বা উভয়ই দ্বারা নির্দিষ্ট (specify) করা যায়।




চিত্র-৩.২ : 8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল

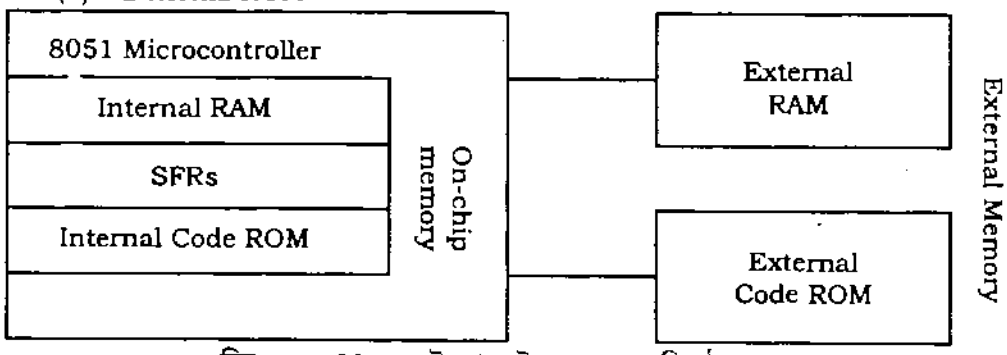
## ৩.৩ 8051 মাইক্রোকন্ট্রোলারের মেমোরির গঠন

**(Describe the Memory Organization of the 8051 Microcontroller Mentioning the Function of SFR, Register Bank, Bit Addressable & General Purpose RAM)**

8051 মাইক্রোকন্ট্রোলার হার্ভার্ড আর্কিটেকচার ভিত্তি করে তৈরি, তাই এর প্রোগ্রাম মেমোরি ও ডাটা মেমোরি পৃথক। প্রোগ্রাম অনুদায়ী মেমোরি যথা– ROM এ এবং ডাটা উদ্দায়ী মেমোরি যথা– RAM এ সংরক্ষণ করা হয়। 8051 এ এরা অভ্যন্তরীণ ও বাহ্যিক উভয়ই হতে পারে।

নিচের 8051 এর মেমোরির গঠন দেখান হলো :

1. Internal (On-chip) Memory
2. External Memory
3. Internal or On-chip Memory
   - (a) Internal RAM
     - (i) Register Bank
     - (ii) Bit addressable
     - (iii) General User RAM/Stack
   - (b) Special Function Registers (SFRs)
   - (c) Internal Code or Program Memory (ROM)
4. External Memory
   - (a) External RAM
   - (b) External ROM



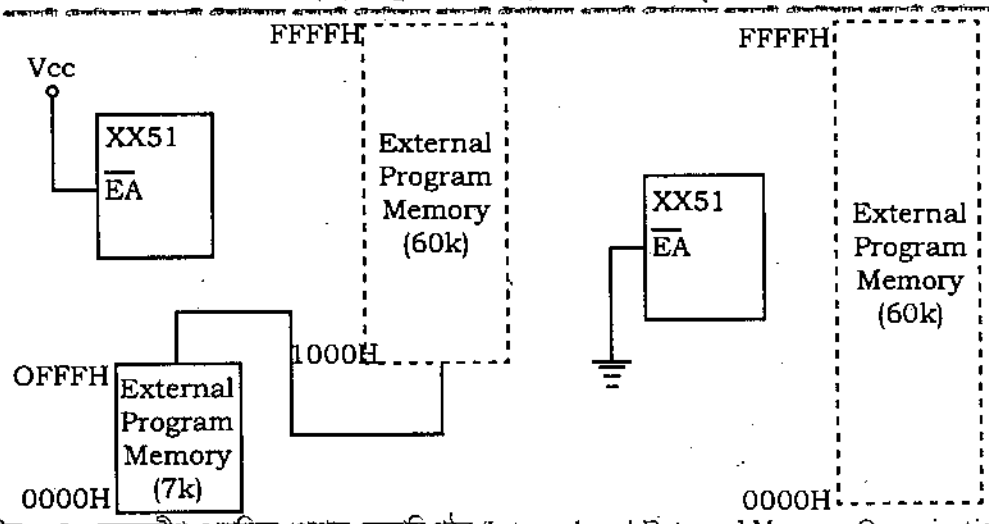

চিত্র-৩.৩ : 8051 মাইক্রোকন্ট্রোলারের মেমোরি গঠন

### ৩.৩.১ প্রোগ্রাম মেমোরি গঠন (অভ্যন্তরীণ ও বাহ্যিক)

**(Program Memory Organization (External and Internal)**

8051 মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ প্রোগ্রাম মেমোরি 4 কিলোবাইট এবং প্রয়োজনে একে 64 কিলোবাইট পর্যন্ত বাহ্যিকভাবে (Externally) বৃদ্ধি করা যায়। তবে এটা 4 কিলোবাইট, 8 কিলোবাইট বা 16 কিলোবাইটের হয় এবং তা নির্ভর করে বিভিন্ন ভার্সনের উপর। On-chip code মেমোরি ROM, EPROM বা Flash বিভিন্ন ধরনের হতে পারে।

প্রাথমিকভাবে অভ্যন্তরীণ মেমোরি হতে নির্দেশনা আনয়নের জন্য মাইক্রোকন্ট্রোলারের EA পিনকে $V_{cc}$ এর সাথে সংযোগ করতে হয় (চিত্র ৩.৪) এবং অভ্যন্তরীণ মেমোরির সীমা অতিক্রান্ত হলে পরবর্তী নির্দেশনা আনয়নের জন্য মাইক্রোকন্ট্রোলারের নিয়ন্ত্রণ স্বয়ংক্রিয়ভাবে বাহ্যিক মেমোরিতে (External Memory) প্রতিষ্ঠা হয়। অভ্যন্তরীণ মেমোরিকে এড়িয়ে (bi-pass) সরাসরি বাহ্যিক মেমোরি হতে নির্দেশনা আনয়নের জন্য EA পিনকে গ্রাউন্ড করতে হয়।

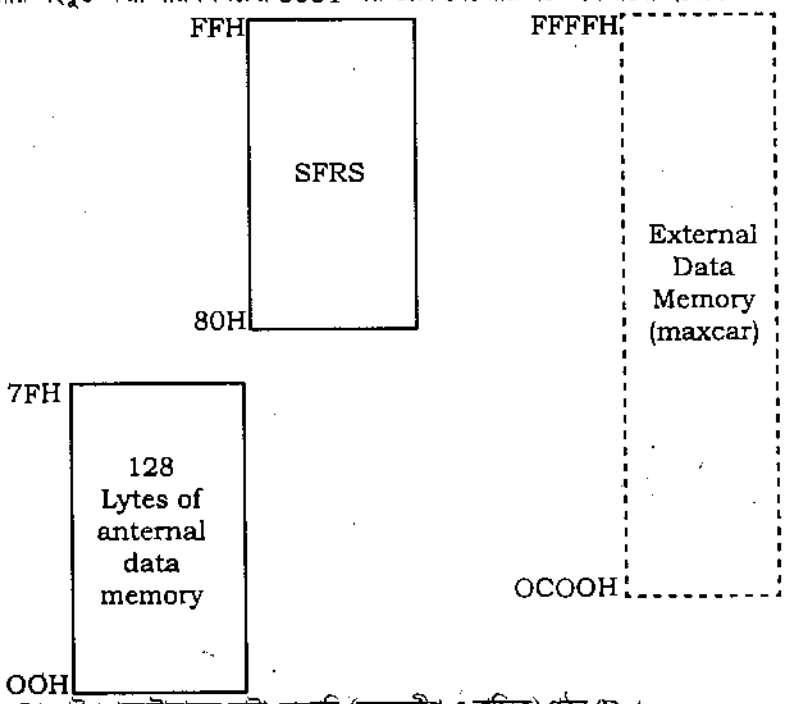



চিত্র-৩.৪ : অভ্যন্তরীণ ও বাহ্যিক প্রোগ্রাম মেমোরি গঠন (Internal and External Memory Organization)

### ৩.৩.২ ডাটা মেমোরির গঠন (অভ্যন্তরীণ ও বাহ্যিক)

**(Data memory organization (Internal)**

8051 এর অভ্যন্তরীণ ডাটা মেমোরি (RAM) আকার 128 বাইট এবং এর সাথে সর্বোচ্চ 64 কিলোবাইট পর্যন্ত বাহ্যিক মেমোরি সংযুক্ত করা যায়। নিচের 8051 এর ডাটা মেমোরি গঠন দেখানো হলো।




চিত্র-৩.৫ : 8051 মাইক্রোকন্ট্রোলারের ডাটা মেমোরি (অভ্যন্তরীণ ও বাহ্যিক) গঠন (Data memory organization)

### ৩.৩.৩ অভ্যন্তরীণ ডাটা মেমোরি

**(Internal data memory or RAM)**

অভ্যন্তরীণ মেমোরি অনেক গতিশীল এবং লিখন/পঠন (Read/Write)-ও পরিবর্তনে অনেক নমনীয়। এটা উদ্বায়ী মেমোরি। 8051 এর অভ্যন্তরীণ মেমোরির ম্যাপিং নিচের চিত্রে দেখান হলো। অভ্যন্তরীণ মেমোরিকে তিনটি ক্ষেত্রে বিভক্ত করা হয় :

**1. রেজিস্টার ব্যাংক (Register Bank) :**

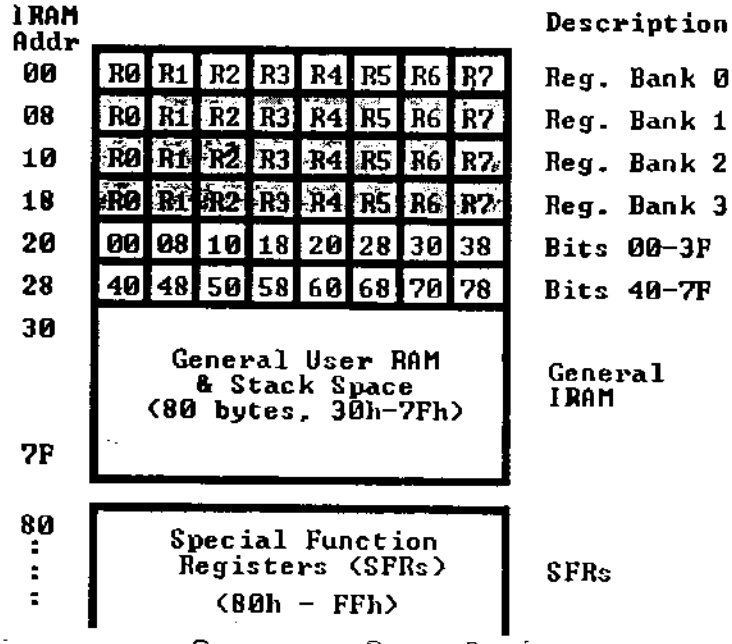
অভ্যন্তরীণ মেমোরির নিচের 32 বাইট (00h–1Fh) এর ক্ষেত্রটি চারটি রেজিস্টার ব্যাংক (Register Bank # 0, Register Bank # 1, Resister Bank # 2, Resister Bank # 3) নিয়ে গঠিত। প্রত্যেক ব্যাংক আটটি করে আট বিটের (বাইট) রেজিস্টার (R0, R1, R2, R3, R4, R5, R6, R7) নিয়ে গঠিত। প্রথম আট বাইট (R0–R7) এর সমন্বয়ে Register Bank # 0 (00h–07h) গঠিত। কাজের জন্য একবারে শুধুমাত্র একটি রেজিস্টার ব্যাংক নির্বাচিত হয় এবং নির্বাচিত ব্যাংকের রেজিস্টারগুলোকে R0, R1 .... ইত্যাদি নেমোনিক ব্যবহার করে অ্যাক্সেস করা হয়। ডাইরেক্ট অ্যাড্রেসিং এর মাধ্যমে অন্যান্য রেজিস্টারগুলোকে যুগপৎ (Simultaneously) অ্যাক্সেস করা যায়। প্রোগ্রাম সম্পাদনের সময় রেজিস্টারগুলো ডাটা বা অপারেন্ড সংরক্ষণ করে। সিস্টেম রিসেটে Default register bank হলো Bank # 0। একটি বিশেষ রেজিস্টার PSW এর দ্বারা Resister Bank নির্বাচন করা হয়।

**2. বিট অ্যাড্রেসেবল র‍্যাম (Bit Addressable RAM) :**

সাধারণত bit variable জমা রাখার জন্য Bit addressable register ব্যবহৃত হয়। রেজিস্টার ব্যাংকের পরের 16 বাইট (128 বিট) হলো bit addressable area। বিট অ্যাড্রেস 00H (20H এর প্রথম বাইটের LSB) হতে 7FH (2FH এর শেষ বাইটের MSB) পর্যন্ত চিহ্নিত। Bit addressable area প্রধানত ব্যবহারিক প্রোগ্রামের bit variable (কোন আউটপুট ডিভাইসের অবস্থা যেমন– LED, MOTOR ইত্যাদির ON/OFF) জমা রাখার জন্য ব্যবহৃত হয়। এই মেমোরির ক্ষেত্রে শুধুমাত্র বিট জমা রাখা ভাল কিন্তু একটি সম্পূর্ণ বাইট জমা রাখা উচিত নয়, কেননা এতে মেমোরি অপচয় হয়।

**3. জেনারেল পারপাস র‍্যা/স্ট্যাক (General Purpore RAM/Stack) :**

অভ্যন্তরীণ মেমোরির বাকি 80 বাইট যার অ্যাড্রেস 30H থেকে 7FH ব্যবহারকারী কর্তৃক ঘন ঘন ব্যবহৃত বা উচ্চ গতির variable এর জন্য একটি সাধারণ storage হিসেবে ব্যবহৃত হয়। এ ক্ষেত্রটি operating stack এর একটি storage area হিসেবেও মাইক্রোকন্ট্রোলার কর্তৃক ব্যবহৃত হয়। যেহেতু 8051 এর অভ্যন্তরীণ RAM এর আকার 128 বাইটে সীমাবদ্ধ তাই কোন কোন প্রয়োগে RAM এর আকার বাড়ানোর প্রয়োজন। 8051 সর্বোচ্চ 64 কিলোবাইট বাহ্যিক RAM সমর্থন করে। বাহ্যিক RAM নমনীয় নয় এবং এর গতি ধীর হয়।




চিত্র-৩.৬ : অভ্যন্তরীণ মেমোরির গঠন

### ৩.৩.৪ স্পেশাল ফাংশন রেজিস্টারস (এসএফআর)

**(Special Function Registers (SFRs)**

SFRs হলো On-chip RAM মেমোরির উপরের 80h হতে FFh পর্যন্ত মেমোরি অ্যাড্রেসে 128 বাইটের একটা ক্ষেত্র। এরা 8051 মাইক্রোকন্ট্রোলার কর্তৃক সম্পাদিত কোন বিশেষ কাজের নিয়ন্ত্রণ ও অবস্থা (Status) প্রকাশ করে। এদের উদ্দেশ্য নির্মাতা কর্তৃক পূর্বনির্ধারিত এবং ফলে অপরিবর্তনীয়। যেহেতু তাদের বিটগুলো মাইক্রোকন্ট্রোলারের অভ্যন্তরে কোন বিশেষ সার্কিটের সাথে সংযুক্ত তাই বিটগুলোর যে-কোন পরিবর্তন মাইক্রোকন্ট্রোলার বা ঐ সার্কিটসমূহের কার্যক্রমে প্রভাব ফেলে। নিচের চিত্রে SFRs map দেখানো হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 80 | PS | SP | DPL | DPH | | | | PCON | 87 |
| 88 | ICON | TMOD | TL0 | TL1 | TH0 | TH1 | | | 8F |
| 90 | P1 | | | | | | | | 97 |
| 98 | SCON | SBUF | | | | | | | 9F |
| A0 | P2 | | | | | | | | A7 |
| A8 | IE | | | | | | | | AF |
| B0 | P3 | | | | | | | | B7 |
| B8 | IP | | | | | | | | B9 |
| C0 | | | | | | | | | C7 |
| C8 | | | | | | | | | CF |
| D0 | PSW | | | | | | | | D7 |
| D8 | | | | | | | | | DF |
| E0 | ACC | | | | | | | | E7 |
| E8 | | | | | | | | | EF |
| F0 | B | | | | | | | | F7 |
| F8 | | | | | | | | | FF |

চিত্র-৩.৭ : SFRs map

**বিভিন্ন প্রকার SFRs ও তাদের কাজ :**

**(ক) আই/ও পোর্ট এসএফআর (I/O Port SFRs) :**

(১) P0- (Port0, Address 80h, 8 bit, Bit-Addressable)

(২) P1- (Port1, Address 90h, 8 bit, Bit-Addressable)

(৩) P2- (Port2, Address A0h, 8 bit, Bit-Addressable)

(৪) P3- (Port3, Address B0h, 8 bit, Bit-Addressable)

এরা প্রত্যেকে 8 বিটের ইনপুট/আউটপুট পোর্ট। এই বিশেষ রেজিস্টারগুলোর প্রত্যেক বিট মাইক্রোকন্ট্রোলারের পিনের সাথে সঙ্গতিপূর্ণ, যেমন– P0 পোর্টের bit 0 হলো মাইক্রোকন্ট্রোলারের P0.0 পিন। এই রেজিস্টারের কোন বিটে 1 লিখলে মাইক্রোকন্ট্রোলারের ঐ পিন High হয় এবং অনুরূপভাবে কোন পিনকে Low করার জন্য রেজিস্টারের ঐ বিটে 0 লিখতে হয়।

**(খ) কন্ট্রোল এসএফআর (Control SFRs) :**

(১) **PCON (Power Control, 8 bit, Addresses 87h) :** এই রেজিস্টার 8051 এর পাওয়ার কন্ট্রোল মুডগুলোকে নিয়ন্ত্রণের জন্য ব্যবহৃত হয়।

(২) **TCON (Timer Control, 8 bit, Addresses 88h, Bit-Addressable) :** এই রেজিস্টার 8051 এর টাইমারদ্বয়ের অপারেশন সাজানো ও পরিবর্তনের জন্য ব্যবহৃত হয়।

(৩) **TMOD (Timer Control, 8 bit, Addresses 89h) :** এই রেজিস্টার টাইমারদ্বয়ের অপারেশনের মুড নির্ধারণ করে।

(৪) **SCON (Serial Control, 8 bit, Addresses 98h, Bit-Addressable) :** এই রেজিস্টার 8051 এর on-board সিরিয়াল পোর্টকে সাজানোর জন্য ব্যবহৃত হয়। এটি baud rate কেও নিয়ন্ত্রণ করে।

(৫) **IE (Interrupt Enable, 8 bit, Addresses A8h) :** এই রেজিস্টার নির্দিষ্ট ইন্টারাপ্টকে কার্যকর বা অকার্যকর করার জন্য ব্যবহৃত হয়।

(৬) **IP (Interrupt Priority 8 bit, Addresses B8h, Bit-Addressable) :** এই রেজিস্টার প্রত্যেক ইন্টারাপ্টের আপেক্ষিক অগ্রগণ্যতা (priority) নির্দিষ্টকরণের জন্য ব্যবহৃত হয়।

(৭) **PSW (Program Status Word, 8 bit; Addresses D0h, Bit-Addressable) :** এই রেজিস্টার গাণিতিক হিসাবনিকাশের ফলাফলের অবস্থা নির্ণয় করে এবং রেজিস্টার ব্যাংক নির্ধারণ করে।

(গ) অন্যান্য **SFRs :**

(১) **SP : SP (Stack Pointer, 8 bit, Address 81h) :** এই রেজিস্টার স্ট্যাক হতে নেওয়া কোন মানকে অভ্যন্তরীণ মেমোরির কোথায় পঠিত (Read) হবে তা নির্দেশ করে।

(২) **DPL/DPH (Data Pointer Low/High, Adresses 82h/83h) :** এটি DPL ও DPH রেজিস্টার একত্রে 16 বিটের একটি Data Pointer. এই ডাটা পয়েন্টার প্রাথমিকভাবে বাহ্যিক মেমোরিকে অ্যাড্রেসিং করার জন্য ব্যবহৃত হয়। তাছাড়া এটি সচরাচর ডাটা সংরক্ষণ ও মধ্যবর্তী ফলাফল সংরক্ষণের জন্যও ব্যবহৃত হয়।

(৩) **TL0/TH0 (Timer 0 Low/High Addresses 8Ah/8Ch) :** TL0 ও TH0 রেজিস্টারদ্বয় একত্রে 16 বিটের টাইমার রেজিস্টার timer0. তাদের যথার্থ আচরণ নির্ভর করে Timer0 কে TMOD রেজিস্টারে কীভাবে সাজানো হয়।

(৪) **TL1/TH1 (Timer 1 Low/High, Addresses 8Bh/8Dh) :** TL1 ও TH1 রেজিস্টারদ্বয় একত্রে 16 বিটের টাইমার রেজিস্টার timer1. তাদের যথার্থ আচরণ নির্ভর করে Timer1 কে TMOD রেজিস্টারে কীভাবে সাজানো হয়।

(৫) **TL1/TH1 (Timer 1 Low/High, Addresses 8Bh/8Dh) :** TL1 ও TH1 রেজিস্টারদ্বয় একত্রে 16 বিটের টাইমার রেজিস্টার timer 1. তাদের যথার্থ আচরণ নির্ভর করে Timer1 কে TMOD রেজিস্টারে কীভাবে সাজানো হয়।

(৬) **SBUF (Serial Control, Addresses 99h) :** এই রেজিস্টার 8051 এর on-board সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণ ও গ্রহণের জন্য ব্যবহৃত হয়।

(৭) **ACC (Accumulator, Addresses E0h, Bit-Addressable) :** এটি বহুল ব্যবহৃত রেজিস্টার যা গাণিতিক হিসাবনিকাশের সমস্ত ফলাফল জমা রাখে, নির্দেশনা সম্পাদনের পূর্বে যে-কোন সংখ্যা বা অপারেন্ডও জমা রাখে। এটি A register নামে অধিক পরিচিত। অ্যাকুমুলেটর ছাড়া কোন মাইক্রোকন্ট্রোলার কল্পনাই করা যায় না।

(৮) **B (B Register, Addresses F0h, Bit-Addressable) :** এটিকে সাময়িক তথ্য জমা রাখার জন্য সহায়ক মেমোরি হিসাবে প্রোগ্রামারগণ ব্যবহার করে। তবে এটি প্রধানত গুণন ও ভাগের কাজের নির্দেশনায় ব্যবহৃত হয়।

## ৩.৪ প্রোগ্রাম স্ট্যাটাস ওয়ার্ড (পিএসডব্লিউ) রেজিস্টার বর্ণনা

## (State the Function of Each Flag of the PSW Register)

**Program Status Word (PSW) :** এটি অত্যন্ত গুরুত্বপূর্ণ একটি 8 বিটের বিশেষ রেজিস্টার যা ALU এর গাণিতিক ও যৌক্তিক হিসাবনিকাশের ফলাফলের অবস্থা (Status) প্রকাশ করে এবং রেজিস্টার ব্যাংক নির্বাচন করে। এটি একটি বিট অ্যাড্রেসেবল রেজিস্টার নিচে এর বিটগুলোর বর্ণনা দেয়া হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ← রিসেটের পর মান |
| PSW | CY | AC | FO | RS1 | RS2 | OV | | P | ← bit এর মান |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

Bit 0 : PSW.1 (Parity Flag)- এটি সাধারণত সিরিয়াল কমিউনিকেশনে ডাটা সম্প্রচার বা গ্রহণে ব্যবহৃত হয়।

1 = অ্যাকুমুলেটর ফলাফল odd parity

0 = অ্যাকুমুলেটর ফলাফল even parity

Bit 1 : এই বিট সংরক্ষিত।

Bit 2 : PSW.2 : OV (Over flow flag)

1 = যদি signed number operation এর ফলাফল খুব বড় হয় তবে sign bit এর high order bit overflow হয়।

0 = Overflow না হলে।

Bit 3 & Bit 4 : PSW.4 – PSW.3 : রেজিস্টার ব্যাংক নির্বাচন বিট। রেজিস্টার ব্যাংক নির্বাচনের বিট মিশ্রণ (Bit combination) নিচে দেখানো হলো :

| RSI | RS0 | RB |
|---|---|---|
| 0 | 0 | BANK 0 |
| 0 | 1 | BANK 1 |
| 1 | 0 | BANK 2 |
| 1 | 1 | BANK 3 |

Bit 5 : SW.5 : FO (User flag) : এটি সাধারণ ব্যবহারের একটি flag; ব্যবহারকারী কর্তৃক প্রয়োজনে এটি Set/Reset হয়।

Bit 6 : PSW.6 : AC (Auxiliary Carry)

1 = BCD অপারেশনে যদি যোগ ও বিয়োগের ফলে ডাটার ৪র্থ বিট (D3) থেকে ৫ম বিটে (D4) carry বের হয়। 0 = carry না থাকলে।

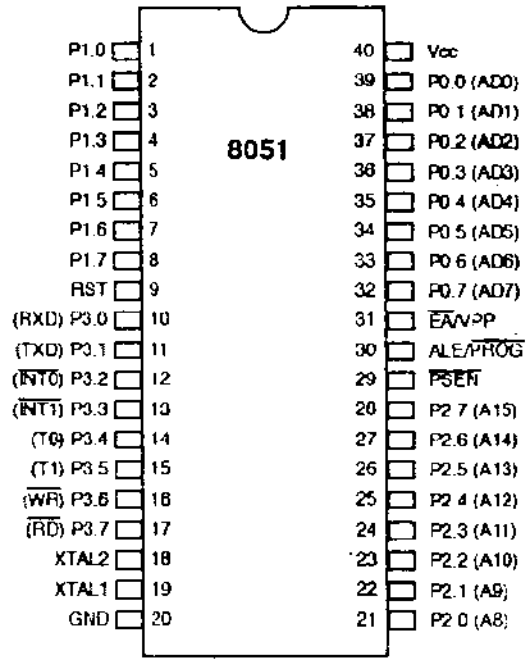
Bit 7 : PSW.7 : CY (Carry Flag)

1 = যদি সকল গাণিতিক ও যৌক্তিক কাজের ফলে অথবা শিফট নির্দেশনার ফলে ডাটার ৮ম বিট (D7) থেকে carry বের হয়।

0 = carry না থাকলে।

## ৩.৫ 8051 মাইক্রোকন্ট্রোলারের পিন ও সিগন্যাল ডায়াগ্রাম ব্যাখ্যা (Explain the Pin and Signal of the 8051 Microcontroller)

8051 মাইক্রোকন্ট্রোলার একটি 40 পিনের DIP (Dual-in-Line Package) IC (চিত্রে দেখানো হয়েছে)। প্রত্যেক পিন সিগন্যালের পূর্ণ বা সংক্ষিপ্ত নামসহ 8051 মাইক্রোকন্ট্রোলারের পিন ও সিগন্যাল ডায়াগ্রাম নিচে অঙ্কন করা হলো। উল্লেখ্য, অনেক পিন দ্বৈত কাজ করে (চিত্রে বিকল্প কাজ লঘু বন্ধনীতে দেখানো হয়েছে)।




চিত্র-৩.৮ : ইন্টেল 8051 মাইক্রোকন্ট্রোলারের পিন ডায়াগ্রাম

**পিনের কাজ**

| পিন নং | বিট/পিনের নাম | পিনের কাজ |
|---|---|---|
| 1–8 | P1.0-P1.7 (PORT1) | 8 বিটের ইনপুট বা আউটপুট পোর্ট। এর প্রত্যেক পিনকে স্বতন্ত্রভাবে ইনপুট বা আউটপুট হিসাবেও ব্যবহার করা যায়। |
| 9 | RST | এটা মাইক্রোকন্ট্রোলারের রিসেট পিন। এতে একটি পজেটিভ পালস মাইক্রোকন্ট্রোলারকে রিসেট করে। কোন ডিভাইসকে রিসেট করার জন্য অসিলেটর চালু অবস্থায় এই পিনকে দুই মেশিন সাইকেলের জন্য high করা হয়। |
| 10–17 | P3.0-P3.7 (PORT3) | এটি পোর্ট 1 এর মতো। এই পিনগুলোকে সর্বজনীন ইনপুট বা আউটপুট হিসেবে ব্যবহার করা যায়। পিনগুলো দ্বৈত কাজ করে। নিচে প্রত্যেক পিনের বিকল্প কাজ উল্লেখ করা হলো : |

10–17 (PORT3) নং পিনের বিকল্প কাজ

| পিন নং | বিট | পিনের নাম | বিকল্প কাজ |
|---|---|---|---|
| 10 | P3.0 | RXD | সিরিয়াল পোর্টে রিসিভ ডাটা (Serial asynchronous communication input বা Serial synchronous communication output.) |
| 11 | P3.1 | TXD | সিরিয়াল পোর্টে ট্রান্সমিট ডাটা (Serial asynchronous communication output বা Serial synchronous communication clock output.) |
| 12 | P3.2 | $\overline{INT0}$ | বাহ্যিক ইন্টারাপ্ট ইনপুট 0 (External Interrupt 0) |
| 13 | P3.3 | $\overline{INT1}$ | বাহ্যিক ইন্টারাপ্ট ইনপুট 1 (External Interrupt 1) |
| 14 | P3.4 | T0 | টাইমার/কাউন্টার 0 (বাহ্যিক ক্লক ইনপুট) |
| 15 | P3.5 | T1 | টাইমার/কাউন্টার 1 (বাহ্যিক ক্লক ইনপুট) |
| 16 | P3.6 | $\overline{WR}$ | বাহ্যিক ডাটা মেমোরি লিখন সিগন্যাল (External data memory write signal) |
| 17 | P3.7 | $\overline{RD}$ | বাহ্যিক ডাটা মেমোরি পঠন সিগন্যাল (External data memory write signal) |
| 18 | XTAL1 | | ইনভার্টিং অ্যাম্প্লিফায়ারের আউটপুট যা অসিলেটরের অংশ এবং অভ্যন্তরীণ ক্লক জেনেরেটরের ইনপুট। বাহ্যিক ক্লক এটির সাথে সংযোগ করতে হয়। |
| 19 | XTAL2 | | ইনভার্টিং অ্যাম্প্লিফায়ারের আউটপুট যা অসিলেটরের অংশ। বাহ্যিক ক্লকের ক্ষেত্রে একে গ্রাউন্ডের সাথে সংযোগ করতে হয়। |
| 20 | GND | | সার্কিট গ্রাউন্ড। |

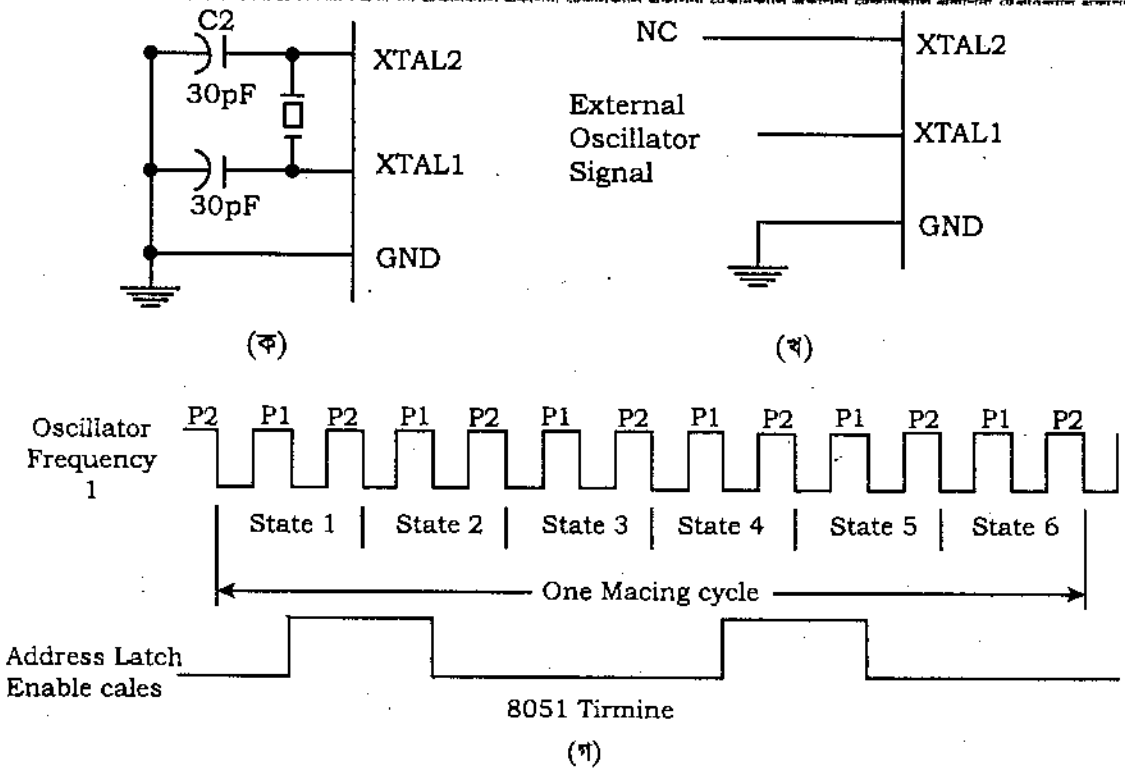
| পিন নং | বিট/পিনের নাম | পিনের কাজ |
|---|---|---|
| 21–28 | P2.0–P2.7 (PORT2) | এটাও একটি 8 বিটের ইনপুট বা আউটপুট পোর্ট। এর পিনগুলো স্বতন্ত্রভাবে ইনপুট/আউটপুট হিসেবে ব্যবহার করা যায়। High order address (A8–A15) এতে ডাটার সাথে multiplexed অবস্থায় থাকে। বাহ্যিক মেমোরি ব্যবহারের সময় এটা তাই high order address bus হিসেবে ব্যবহৃত হয়। |

| পিন নং | বিট/পিনের নাম | পিনের কাজ |
|---|---|---|
| 29 | $\overline{PSEN}$ | **Program Store Enable (PSEN) output :** এটি একটি আউটপুট কন্ট্রোল সিগন্যাল। এটা বাহ্যিক প্রোগ্রাম মেমোরি (ROM) পঠনের সময় প্রতি 6 অসিলেটর ফ্রিকোয়েন্সিতে সচল হয়, অর্থাৎ LOW। অভ্যন্তরীণ প্রোগ্রাম সম্পাদনের সময় এটা অকার্যকর অর্থাৎ High থাকে। |
| 30 | ALE/$\overline{PROG}$ | **Addres latch enable (ALE) output :** এটি বাহ্যিক মেমোরি access করার সময় low order address কে latch করার জন্য ব্যবহৃত হয়। এটা অসিলেটর ফ্রিকোয়েন্সির 1/6 হারে পর্যায়ক্রমিকভাবে সচল হয়। একাধিক মেমোরি ব্যবহার করলে এই পিন তাদেরকে পার্থক্য করার জন্য ব্যবহৃত হয়। তাছাড়া EEPROM কে প্রোগ্রামিং করার সময় এই পিন প্রোগ্রাম পালস ইনপুট করে। |
| 31 | $\overline{EA}$/Vpp | **External access (EA) :** যদি অভ্যন্তরীণ ও বাহ্যিক উভয় মেমোরি ব্যবহৃত হয় তবে এটা High হলে অভ্যন্তরীণ প্রোগ্রাম মেমোরি থেকে 0FFFh পর্যন্ত অ্যাড্রেসের নির্দেশনা সম্পাদিত হয় এবং এরপর বাহ্যিক মেমোরি হতে নির্দেশনা সম্পাদিত হয়। এটি Low হলে সব নির্দেশনাই বাহ্যিক মেমোরি হতে Fetch হয়ে নির্বাহ হবে। বাহ্যিক মেমোরি না থাকলে এটা $V_{cc}$ এর সাথে High করে রাখা হয়। |
| 32–39 | P0.0–P0.7 (PORT0) | Port0, একটি প্রকৃত দ্বিমুখী ইনপুট/আউটপুট পোর্ট। এতে Low order address (A0-A7) এবং data bus multiplexed অবস্থায় থাকে। এর বিটগুলোকেও স্বতন্ত্রভাবে ইনপুট/আউটপুট হিসেবে ব্যবহার করা যায়। |
| 40 | $V_{cc}$ | পাওয়ার সাপ্লাই পিন। এর অপারেটিং ভোল্টেজ + 5 ভোল্ট, কারেন্ট 125 মিলিঅ্যাম্পিয়ার এবং সর্বোচ্চ পাওয়ার খরচ 1 ওয়াট। |

## ৩.৬ 8051 মাইক্রোকন্ট্রোলারের ক্লক ও রিসেট সার্কিট
## (The clock and reset circuits of the 8051 microcontroller)

### ৩.৬.১ ক্লক সার্কিট
### (The clock circuits)

একটি মাইক্রোকন্ট্রোলারের হার্ট হলো ক্লক পালস উৎপাদক। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের একটি on-chip অসিলেটর আছে। যদিও বা এটি মাইক্রোকন্ট্রোলারের অভ্যন্তরে সমস্ত সার্কিটে synchronous operation করে তথাপি stable frequency এর জন্য বাহ্যিক একটি ক্লক দরকার। ক্লক হিসেবে কোয়ার্টস ক্রিস্টাল বা সিরামিক রিজোনেটর ব্যবহৃত হয়। সিরামিক অসিলেটর সস্তা হলেও এর stability কম। নিচের চিত্রে 8051 মাইক্রোকন্ট্রোলারের সাথে কোয়ার্টস ক্রিস্টালের সংযোগ দেখানো হলো। ক্রিস্টালকে XTAL1 (pin19) ও XTAL2 (pin18) এর মাঝে সংযোগ করা হয় (চিত্র ক)। এতে 30 পিকো ফ্যারাডের দুটি ক্যাপাটির ব্যবহৃত হয়। ক্রিস্টাল অসিলেটর ছাড়া অন্য কোন frequency source যেমন– TTL oscillator ব্যবহার করলে এটিকে XTAL1 এর সাথে সংযোগ করতে হয় এবং XTAL2 কে সংযোগহীন (NC) রাখা হয় (চিত্র খ)। ক্রিস্টালের সর্বোচ্চ ফ্রিকোয়েন্সিই মাইক্রোকন্ট্রোলারের গতি হিসেবে বিবেচিত। ক্রিস্টাল ফ্রিকোয়েন্সির রেঞ্জ 8 মে. হার্টস – 30 মে. হার্টস হয় এবং সাধারণত 11.0592 মে. হার্টস ক্রিস্টাল ব্যবহৃত হয়, কারণ UART তে বহুল ব্যবহৃত baud rate এ এটি যথার্থ ফ্রিকোয়েন্সি প্রদান করে।




**চিত্র-৩.৯ :** 8051 মাইক্রোকন্ট্রোলারের ক্লক সার্কিট

(ক) Crystal Oscillator (খ) TTL Oscillator (গ) Clock state

ক্রিস্টাল, ক্যাপাসিটর এবং অন-চিপ ইনভার্টার দ্বারা তৈরি অসিলটরে একটি pulse train উৎপন্ন করে (চিত্র গ)। ক্লক ফ্রিকোয়েন্সি f, মাইক্রোকন্ট্রোলারের অভ্যন্তরে পালস টাইম (P) নামে একটি ক্ষুদ্রতর টাইম ইন্টারভাল বা টাইম স্টেট প্রতিষ্ঠা করে। 8051 এ প্রত্যেক নির্দেশনায় ছয়টি টাইম স্টেট আছে এবং প্রত্যেক স্টেটে দুটি পালস থাকে। কোন সরল নির্দেশনা বা কোন জটিল নির্দেশনার অংশ সম্পাদনে প্রয়োজনীয় সংখ্যক ক্ষুদ্রতম টাইম ইন্টারভালকে মেশিন সাইকেল বলে। প্রয়োজনভেদে প্রোগ্রাম নির্দেশনা সম্পাদনায় এক, দুই বা চার মেশিন সাইকেল আবশ্যক হতে পারে।

### ৩.৬.২ রিসেট সার্কিট

**(The Reset Circuit)**

অন্তত দুই মেশিন সাইকেলের জন্য RST (8051 মাইক্রোকন্ট্রোলারের ৯নং পিন) কে High করে এবং পরে Low তে প্রত্যাবর্তন করে 8051 মাইক্রোকন্ট্রোলারকে রিসেট করা হয়। RST কে একটি সুইচের মাধ্যমে হাতের দ্বারা (Manuaaly) অথবা একটি R-C Network ব্যবহার করে Power up এর মাধ্যমে দুই ভাবে রিসেট করা হয়।

**Power-On Reset :** এতে পাওয়ার অন হলেই 8051 মাইক্রোকন্ট্রোলার রিসেট হয়।

– প্রাথমিকভাবে ক্যাপাসিটর চার্জিং এ RST সিগন্যাল High হয়।

– যখন ক্যাপাসিটর সম্পূর্ণ চার্জ হয় তখন এটি DC কে ব্লক করে।

**Manual reset :** এতে ম্যানুয়ালি একটি পুশ বাটন সুইচের মাধ্যমে 8051 মাইক্রোকন্ট্রোলারকে রিসেট করা হয়।

– সুইচকে সাময়িক সময়ের জন্য বন্ধ করলে RST সিগন্যাল High হয়।

**রিসেটের পর :**

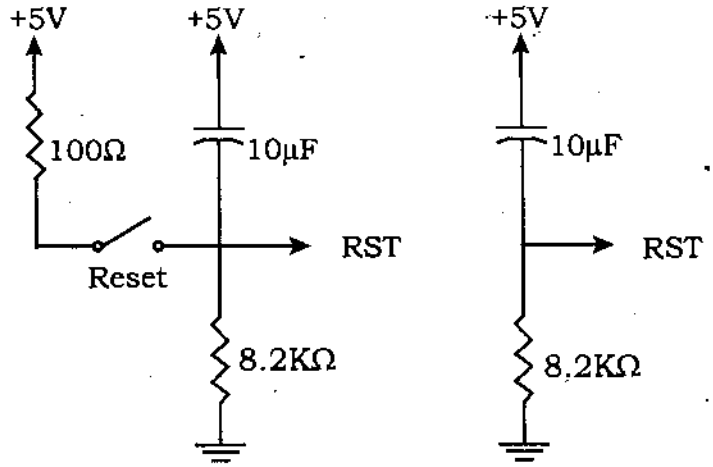
(ক) প্রোগ্রাম কাউন্টার 0000H বোঝাই (Load) করে।

(খ) Register Bank # 0 নির্বাচিত হয়।

(গ) Stack Pointer এ 07H সূচিত (Initialize) হয়।

(ঘ) সব পোর্ট FFH এ (Load) হয়।

(ঙ) কিন্তু RAM এ রক্ষিত তথ্যে কোন প্রভাব পড়ে না।




Manual Reset Power on Reset

**চিত্র-৩.১০ :** 8051 এর রিসেট সার্কিট

# অনুশীলনী-০৩

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **Sepcial function registers (SFRs) কী?** *[বাকাশিবো : '১৪R]*

**উত্তর :** SFRs হলো On-chip RAM মেমোরির উপরের 80h হতে FFh পর্যন্ত মেমোরির অ্যাড্রেসে 128 বাইটের একটা ক্ষেত্র। এরা 8051 মাইক্রোকন্ট্রোলার কর্তৃক সম্পাদিত কোন বিশেষ কাজের নিয়ন্ত্রণ ও অবস্থা (Status) প্রকাশ করে। এদের উদ্দেশ্য নির্মাতা কর্তৃক পূর্বনির্ধারিত এবং ফলে অপরিবর্তনীয়। যেহেতু তাদের বিটগুলো মাইক্রোকন্ট্রোলারের অভ্যন্তরে কোন বিশেষ সার্কিটের সাথে সংযুক্ত তাই বিটগুলোর যে-কোন পরিবর্তন মাইক্রোকন্ট্রোলার বা ঐ সার্কিটসমূহের কার্যক্রমে প্রভাব ফেলে।

২। **8051 মাইক্রোকন্ট্রোলারের গাঠনিক ভিত্তি কী?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের আর্কিটেকচার- হার্ভার্ড আর্কিটেকচার এবং প্রসেসরের ধরন- RISC (Reduced Instruction Set Computer).

৩। **8051 মাইক্রোকন্ট্রোলারের ক্লক ফ্রিকোয়েন্সি রেঞ্জ (range) কত?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের ক্লক ফ্রিকোয়েন্সি রেঞ্জ 3–30 মে. হার্টস।

৪। **8051 মাইক্রোকন্ট্রোলার আইসি এ সিরিয়াল কমিউনিকেশনে ব্যবহৃত পিন দুটির নাম লিখ।**

**উত্তর :** 8051 এ সিরিয়াল কমিউনিকেশনে ব্যবহৃত পিন দুটি হলো Txd ও Rxd।

৫। **8051 এ ইনপুট/আউটপুট পোর্ট কয়টি ও কী কী?**

**উত্তর :** 8051 এ ইনপুট/আউটপুট পোর্ট ৪টি। এরা হলো : PORT0, PORT1, PORT2 ও PORT3.

৬। **8051 এর বাহ্যিক ইন্টারাপ্ট এর জন্য ব্যবহৃত পিনগুলোর নাম লিখ।**

**উত্তর :** 8051 এ ইনপুট/আউটপুট পোর্টে ব্যবহৃত বিশেষ রেজিস্টারগুলো হলো- P0, P1, P2 ও P3.

৭। **8051 এর বাহ্যিক ইন্টারাপ্ট এর জন্য ব্যবহৃত পিনগুলোর নাম লিখ।**

**উত্তর :** 8051 এর বাহ্যিক ইন্টারাপ্ট এর জন্য ব্যবহৃত পিন দুটি হলো- INT1, INT2.

৮। **8051 এ 16 বিটের অ্যাড্রেস কীভাবে পাওয়া যায়?**

**উত্তর :** PORT0 তে ডাটা বিটের সাথে Low order address (8 বিট) multiplexed অবস্থায় থাকে এবং PORT2, High order address (8 বিট) ধারণ করে। কন্ট্রোল সিগন্যাল ALE এর মাধ্যমে De-multiplexing হয়ে 16 বিটের অ্যাড্রেস তৈরি হয়।

৯। **8051 মাইক্রোকন্ট্রোলারে টাইমার/ কাউন্টার অপারেশনে ব্যবহৃত বিশেষ রেজিস্টারগুলোর নাম লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারে টাইমার/কাউন্টারে ব্যবহৃত বিশেষ রেজিস্টারসমূহ হলো : TCON, TMOD, TL1, TH1, TL0, TH0।

১০। **8051 এর প্রোগ্রাম মেমোরির আকার কত?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ প্রোগ্রাম মেমোরি 4 কিলোবাইট এবং প্রয়োজনে একে 64 কিলোবাইট পর্যন্ত বাহ্যিকভাবে (Externally) বৃদ্ধি করা যায়। তবে এটি 4 কিলোবাইট, 8 কিলোবাইট বা 16 কিলোবাইটের হয় এবং তা নির্ভর করে বিভিন্ন ভার্সনের উপর।

১১। **8051 এর অভ্যন্তরীণ ডাটা মেমোরির (RAM) বিভিন্ন স্তরগুলোর নাম লিখ।**

**উত্তর :** 8051 এর অভ্যন্তরীণ ডাটা মেমোরির স্তরগুলো :

(ক) Register Bank

(খ) Bit addressable memory

(গ) General purpose memory/Stack.

**১২। A register বা Accumulator কী এবং এর কাজ কী?**

**উত্তর :** Accumulator একটি 8 বিটের Bit-Addressable রেজিস্টার। এটি বহুল ব্যবহৃত রেজিস্টার যা গাণিতিক হিসাবনিকাশের সমস্ত ফলাফল জমা রাখে। নির্দেশনা সম্পাদনের পূর্বে যে-কোন সংখ্যা বা নির্দেশনার অপারেন্ডও জমা রাখে। এটি A register নামে অধিক পরিচিত। অ্যাকুমুলেটর ছাড়া কোন মাইক্রোকন্ট্রোলার কল্পনাই করা যায় না।

**১৩। B রেজিস্টারের কাজ কী?**

**উত্তর :** B Register একটি Bit-Addressable রেজিস্টার। এটি সাময়িক তথ্য জমা রাখার জন্য সহায়ক মেমোরি প্রোগ্রামারগণ ব্যবহার করে। তবে এটি প্রধানত গুণন ও ভাগের কাজের নির্দেশনায় ব্যবহৃত হয়।

**১৪। PSW রেজিস্টারের কাজ কী?**

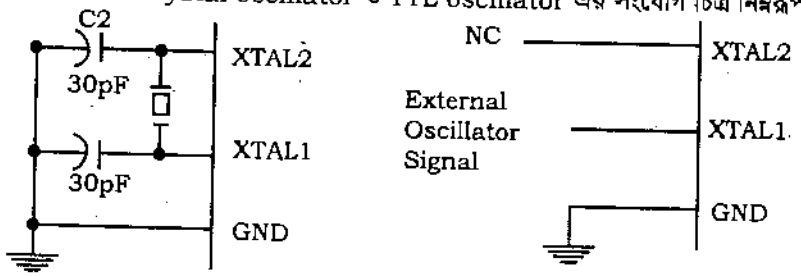
**উত্তর :** Program Status Word (PSW) একটি অত্যন্ত গুরুত্বপূর্ণ একটি 8 বিটের বিট অ্যাড্রেসেবল বিশেষ রেজিস্টার যা–

(ক) ALU এর গাণিতিক ও যৌক্তিক হিসাব-নিকাশের ফলাফলের অবস্থা (Status) প্রকাশ করে এবং

(খ) রেজিস্টার ব্যাংক নির্বাচন করে।

**১৫। 8051 এর সাথে Crystal oscillator ও TTL oscillator এর সংযোগ চিত্র দেখাও।**

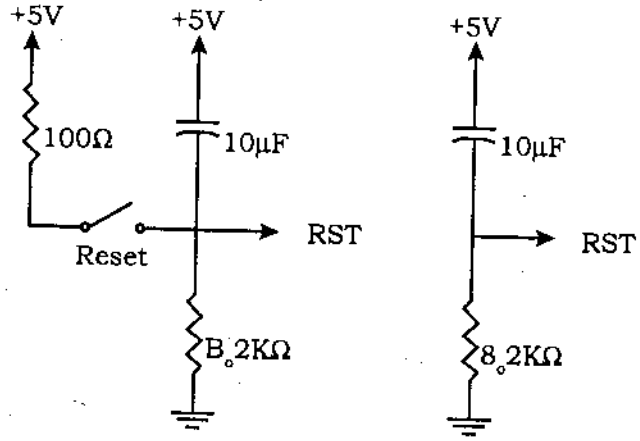
**উত্তর :** 8051 এর সাথে Crystal oscillator ও TTL oscillator এর সংযোগ চিত্র নিম্নরূপ :




(ক) (খ)

**১৬। 8051 এর রিসেট সার্কিট অঙ্কন কর।**

**উত্তর :** 8051 এর রিসেট সার্কিট অঙ্কন করা হলো :




Manual reset power on reset

চিত্র : 8051 এর রিসেট সার্কিট

**১৭। ক্রিস্টাল ফ্রিকোয়েন্সির রেঞ্জ কত?**

**উত্তর :** ক্রিস্টাল ফ্রিকোয়েন্সির রেঞ্জ 4 – 30 মে. হার্টস।

**১৮। অধিকাংশ ক্ষেত্রে কত মানের ক্রিস্টাল ব্যবহার করা হয়?**

**উত্তর :** অধিকাংশ ক্ষেত্রে সাধারণত 11.0592 মে. হার্টস।

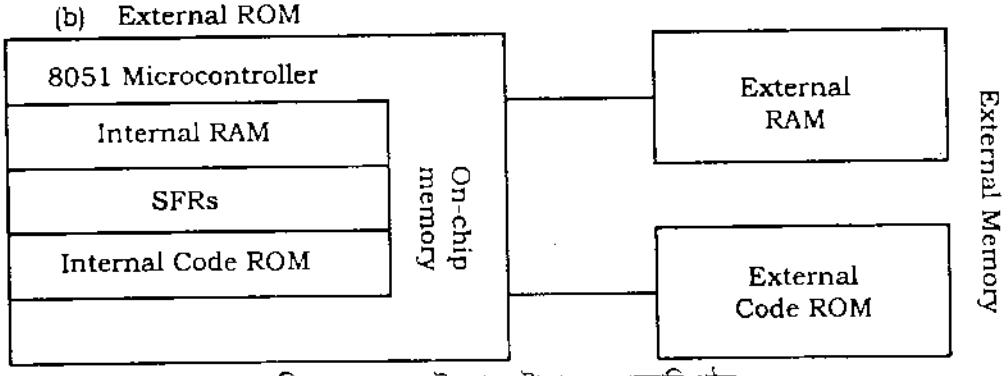
## সংক্ষিপ্ত প্রশ্নোত্তর

১। **8051 মাইক্রোকন্ট্রোলারের মেমোরির গঠন চিত্র অঙ্কন কর ও বর্ণনা কর।** *[বাকাশিবো : '১৫]*

**উত্তর :** 8051 মাইক্রোকন্ট্রোলার হার্ভার্ড আর্কিটেকচার ভিত্তি করে তৈরি, তাই এর প্রোগ্রাম মেমোরি ও ডাটা মেমোরি পৃথক। প্রোগ্রাম অনুদ্বায়ী মেমোরি যথা– ROM এ এবং ডাটা উদ্বায়ী মেমোরি যথা– RAM এ সংরক্ষণ করা হয়। 8051 এ এরা অভ্যন্তরীণ ও বাহ্যিক উভয়ই হতে পারে।

চিত্র 8051 এর মেমোরির গঠন দেখান হলো :

1. Internal (On-chip) Memory
2. External Memory
3. Internal or On-chip Memory
   - (a) Internal RAM
     - (i) Register Bank
     - (ii) Bit addressable
     - (iii) General User RAM/Stack
   - (b) Special Function Registers (SFRs)
   - (c) Internal Code or Program Memory (ROM)
4. External Memory
   - (a) External RAM
   - (b) External ROM




চিত্র : 8051 মাইক্রোকন্ট্রোলারের মেমোরি গঠন

২। **8051 এর কন্ট্রোল SFR-গুলোর নাম লিখ।**

**অথবা, অ্যাড্রেসহ স্পেশাল ফাংশন রেজিস্টারগুলোর নাম লেখ।** *[বাকাশিবো : '১৫]*

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের কন্ট্রোল SFR 8টি, যথা–

(ক) PCON (Power Control) (Address 87h)

(খ) TCON (Timer Control) (Address 88h)

(গ) TMOD (Timer Mode) (Address 89h)

(ঘ) SCON (Serial Control) (Address 98h)

(ঙ) PSW (Program Status Word) (Address D0)

(চ) IP (Interrupt Priority) (Address B8)

(ছ) IE (Interrupt Enable) (Address A8)

৩। **8051 মাইক্রোকন্ট্রোলারের রিসেট সার্কিট বর্ণনা কর।** *[বাকাশিবো : '১৪]*

**উত্তর :** অন্তত দুই মেশিন সাইকেলের জন্য RST (8051 মাইক্রোকন্ট্রোলারের ৯নং পিন) কে High করে এবং পরে Low তে প্রত্যাবর্তন করে 8051 মাইক্রোকন্ট্রোলারকে রিসেট করা হয়। RST কে একটি সুইচের মাধ্যমে হাতের দ্বারা (Manuaaly) অথবা একটি R-C Network ব্যবহার করে Power up এর মাধ্যমে দুভাবে রিসেট করা হয়।

**Power-On Reset :** এতে পাওয়ার অন হলেই 8051 মাইক্রোকন্ট্রোলার রিসেট হয়।

– প্রাথমিকভাবে ক্যাপাসিটর চার্জিং এ RST সিগন্যাল High হয়।

– যখন ক্যাপাসিটর সম্পূর্ণ চার্জ হয় তখন এটি DC কে ব্লক করে।

**Manual reset :** এতে ম্যানুয়ালি একটি পুশ বাটন সুইচের মাধ্যমে 8051 মাইক্রোকন্ট্রোলারকে রিসেট করা হয়।

– সুইচকে সাময়িক সময়ের জন্য বন্ধ করলে RST সিগন্যাল High হয়।

**রিসেটের পর :**

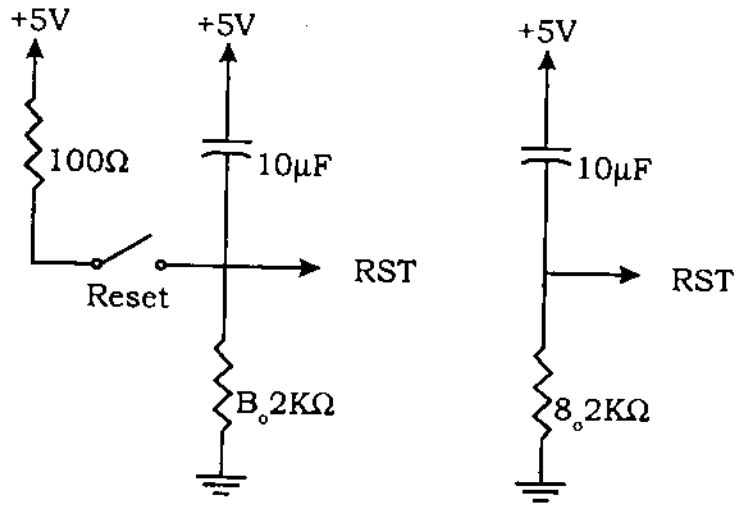
(ক) প্রোগ্রাম কাউন্টার 0000H বোঝাই (Load) করে।

(খ) Register Bank # 0 নির্বাচিত হয়।

(গ) Stack Pointer এ 07H সূচিত (Initialize) হয়।

(ঘ) সব পোর্ট FFH এ (Load) হয়।

(ঙ) কিন্তু RAM এ রক্ষিত তথ্যে কোন প্রভাব পড়ে না।




Manual reset power on reset

**চিত্র :** 8051 এর রিসেট সার্কিট

৪। **PSW রেজিস্টারের বিটগুলো বর্ণনা কর।** *[বাকাশিবো : '১৪R]*

**উত্তর :** এটি অত্যন্ত গুরুত্বপূর্ণ একটি 8 বিটের বিশেষ রেজিস্টার যা ALU এর গাণিতিক ও যৌক্তিক হিসাবনিকাশের ফলাফলের অবস্থা (Status) প্রকাশ করে এবং রেজিস্টার ব্যাংক নির্বাচন করে। এটি একটি বিট অ্যাড্রেসেবল রেজিস্টার নিচে এর বিটগুলোর বর্ণনা দেয়া হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ← রিসেটের পর মান |
| PSW | CY | AC | FO | RS1 | RS2 | OV | | P | ← bit এর মান |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

Bit 0 : PSW.1 (Parity Flag)– এটি সাধারণত সিরিয়াল কমিউনিকেশনে ডাটা সম্প্রচার বা গ্রহণে ব্যবহৃত হয়।

1 = অ্যাকুমুলেটর ফলাফল odd parity

0 = অ্যাকুমুলেটর ফলাফল even parity

Bit 1 : এই বিট সংরক্ষিত।

Bit 2 : PSW.2 : OV (Over flow flag)

1 = যদি signed number operation এর ফলাফল খুব বড় হয় তবে sign bit এর high order bit overflow হয়।

0 = Overflow না হলে।

Bit 3 & Bit 4 : PSW.4 – PSW.3 : রেজিস্টার ব্যাংক নির্বাচন বিট। রেজিস্টার ব্যাংক নির্বাচনের বিট মিশ্রণ (Bit combination) নিচে দেখানো হলো :

| RSI | RS0 | RB |
|---|---|---|
| 0 | 0 | BANK 0 |
| 0 | 1 | BANK 1 |
| 1 | 0 | BANK 2 |
| 1 | 1 | BANK 3 |

Bit 5 : SW.5 : FO (User flag) : এটি সাধারণ ব্যবহারের একটি flag; ব্যবহারকারী কর্তৃক প্রয়োজনে এটি Set/Reset হয়।

Bit 6 : PSW.6 : AC (Auxiliary Carry)

1 = BCD অপারেশনে যদি যোগ ও বিয়োগের ফলে ডাটার ৪র্থ বিট (D3) থেকে ৫ম বিটে (D4) carry বের হয়। 0 = carry না থাকলে।

Bit 7 : PSW.7 : CY (Carry Flag)

1 = যদি সকল গাণিতিক ও যৌক্তিক কাজের ফলে অথবা শিফট নির্দেশনার ফলে ডাটার ৮ম বিট (D7) থেকে carry বের হয়।

0 = carry না থাকলে।

**৫। 8051 এ টাইমার/কাউন্টারের কাজ কী?**

**উত্তর :** 8051 এ টাইমার/কাউন্টারের কাজ :

(ক) দুটি ইভেন্টের মধ্যকার সময় পরিমাপ,

(খ) কালক্ষেপণ (delay) পরিমাপ,

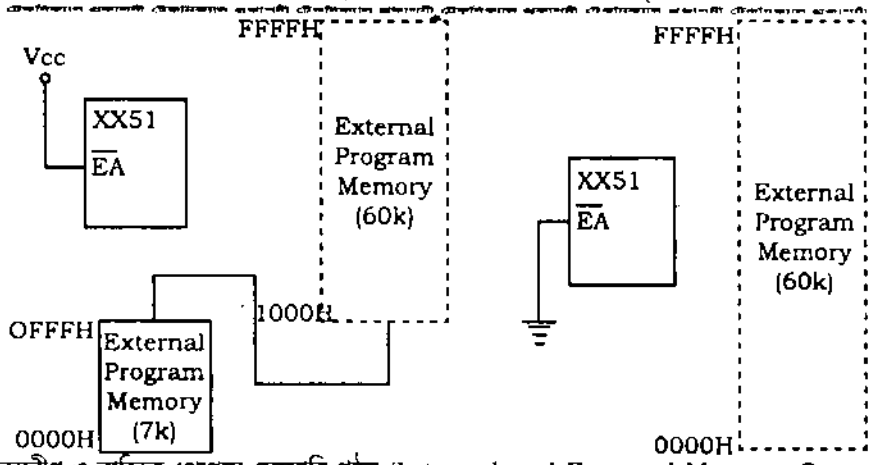
(গ) পালস গণনা,

(ঘ) পালসের প্রশস্ততা (pulse width) পরিমাপ এবং

(ঙ) baud rate উৎপাদন।

**৬। 8051 এ প্রোগ্রাম মেমোরি হতে নির্দেশনা পঠন (Read) প্রক্রিয়া উল্লেখ কর।**

**উত্তর :** প্রাথমিকভাবে অভ্যন্তরীণ মেমোরি হতে নির্দেশনা আনয়নের জন্য 8051 মাইক্রোকন্ট্রোলারের EA পিনকে $V_{cc}$ এর সাথে সংযোগ করতে হয় এবং অভ্যন্তরীণ মেমোরির সীমা অতিক্রান্ত হলে পরবর্তী নির্দেশনা আনয়নের জন্য মাইক্রোকন্ট্রোলারের নিয়ন্ত্রণ স্বয়ংক্রিয়ভাবে বাহ্যিক মেমোরিতে (External Memory) প্রতিষ্ঠা হয়। অভ্যন্তরীণ মেমোরিকে এড়িয়ে (bi-pass) সরাসরি বাহ্যিক মেমোরি হতে নির্দেশনা আনয়নের জন্য EA পিনকে গ্রাউন্ড করতে হয়। চিত্রে এটি দেখান হলো :




চিত্র : অভ্যন্তরীণ ও বাহ্যিক প্রোগ্রাম মেমোরি গঠন (Internal and External Memory Organization)

**৭। 8051 মাইক্রোকন্ট্রোলারের SFR Map অঙ্কন কর।**

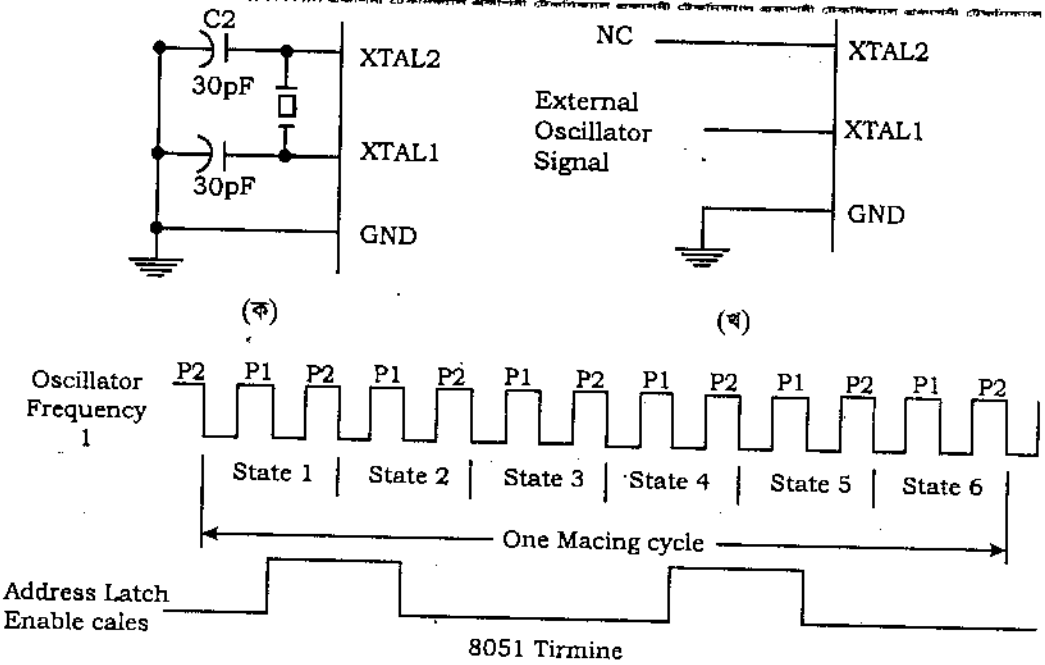
**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের SFR Map অংকন করা হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 80 | PS | SP | DPL | DPH | | | | PCON | 87 |
| 88 | ICON | TMOD | TL0 | TL1 | TH0 | TH1 | | | 8F |
| 90 | P1 | | | | | | | | 97 |
| 98 | SCON | SBUF | | | | | | | 9F |
| A0 | P2 | | | | | | | | A7 |
| A8 | IE | | | | | | | | AF |
| B0 | P3 | | | | | | | | B7 |
| B8 | IP | | | | | | | | B9 |
| C0 | | | | | | | | | C7 |
| C8 | | | | | | | | | CF |
| D0 | PSW | | | | | | | | D7 |
| D8 | | | | | | | | | DF |
| E0 | ACC | | | | | | | | E7 |
| E8 | | | | | | | | | EF |
| F0 | B | | | | | | | | F7 |
| F8 | | | | | | | | | FF |

চিত্র : SFRs map

**৮। 8051 মাইক্রোকন্ট্রোলারের ক্লক সার্কিট বর্ণনা কর।**

**উত্তর :** একটি মাইক্রোকন্ট্রোলারের হার্ট হলো ক্লক পালস উৎপাদক। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের একটি on-chip অসিলেটর আছে। যদিও বা এটি মাইক্রোকন্ট্রোলারের অভ্যন্তরে সমস্ত সার্কিটে synchronous operation করে তথাপি stable frequency এর জন্য বাহ্যিক একটি ক্লক দরকার। ক্লক হিসাবে কোয়ার্টস ক্রিস্টাল বা সিরামিক রিজোনেটর ব্যবহৃত হয়। সিরামিক অসিলেটর সস্তা হলেও এর stability কম। নিচের চিত্রে 8051 মাইক্রোকন্ট্রোলারের সাথে কোয়ার্টস ক্রিস্টালের সংযোগ দেখানো হলো। ক্রিস্টালকে XTAL1 (pin19) ও XTAL2 (pin18) এর মাঝে সংযোগ করা হয় (চিত্র ক)। এতে 30 পিকো ফ্যারাডের দুটি ক্যাপাটির ব্যবহৃত হয়। ক্রিস্টাল অসিলেটর ছাড়া অন্য কোন frequency source যেমন– TTL oscillator ব্যবহার করলে এটিকে XTAL1 এর সাথে সংযোগ করতে হয় এবং XTAL2 কে সংযোগহীন (NC) রাখা হয় (চিত্র খ)। ক্রিস্টালের সর্বোচ্চ ফ্রিকোয়েন্সিই মাইক্রোকন্ট্রোলারের গতি হিসাবে বিবেচিত। ক্রিস্টাল ফ্রিকোয়েন্সির রেঞ্জ 8 মে. হার্টস – 30 মে. হার্টস হয় এবং সাধারণত 11.0592 মে. হার্টস ক্রিস্টাল ব্যবহৃত হয়, কারণ UART তে বহুল ব্যবহৃত baud rate এ এটি যথার্থ ফ্রিকোয়েন্সি প্রদান করে।



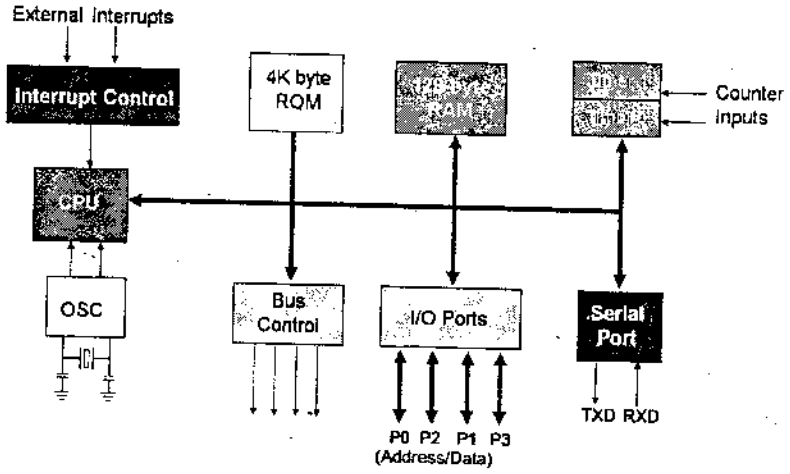

চিত্র : 8051 মাইক্রোকন্ট্রোলারের ক্লক সার্কিট

(ক) Crystal Oscillator (খ) TTL Oscillator (গ) Clock state

ক্রিস্টাল, ক্যাপাসিটর এবং অন-চিপ ইনভার্টার দ্বারা তৈরি অসিলটরে একটি pulse train উৎপন্ন করে (চিত্র গ)। ক্লক ফ্রিকোয়েন্সি f, মাইক্রোকন্ট্রোলারের অভ্যন্তরে পালস টাইম (P) নামে একটি ক্ষুদ্রতর টাইম ইন্টারভাল বা টাইম স্টেট প্রতিষ্ঠা করে। 8051 এ প্রত্যেক নির্দেশনায় ছয়টি টাইম স্টেট আছে এবং প্রত্যেক স্টেটে দুটি পালস থাকে। কোন সরল নির্দেশনা বা কোন জটিল নির্দেশনার অংশ সম্পাদনে প্রয়োজনীয় সংখ্যক ক্ষুদ্রতম টাইম ইন্টারভালকে মেশিন সাইকেল বলে। প্রয়োজনভেদে প্রোগ্রাম নির্দেশনা সম্পাদনায় এক, দুই বা চার মেশিন সাইকেল আবশ্যক হতে পারে।

৯। **ইন্টেল 8051 মাইক্রো কন্ট্রোলারের সরল ব্লক ডায়াগ্রাম অঙ্কন কর।**

**উত্তর :** নিচে এর ব্লকগুলোর সংক্ষিপ্ত বিবরণ দেয়া হলো :




চিত্র : 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম

১০। **ইন্টেল কোম্পানি 8051 মাইক্রো কন্ট্রোলারের বৈশিষ্ট্য লেখ।**

**উত্তর :** এটি নিচের বৈশিষ্ট্যগুলো ধারণ করে :

(i) 128 bytes RAM

(ii) 4k bytes ROM

(iii) 32টি I/O Pins

(iv) 2টি Timers

(v) 1টি সিরিয়াল পোর্ট

(vi) 6টি Interrupt Source রয়েছে।

## রচনামূলক প্রশ্নোত্তর

১। **8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন ব্লকের বর্ণনা দাও।** *[বাকাশিবো : '১৪, '১৪R, '১৫R]*

**উত্তর :** অনুচ্ছেদ ৩.১ নং দ্রষ্টব্য।

২। **8051 মাইক্রোকন্ট্রোলারের পিন ডায়াগ্রাম অঙ্কন করে পিনগুলোর কাজের সংক্ষিপ্ত বর্ণনা দাও।** *[বাকাশিবো : '১৪R]*

**উত্তর :** অনুচ্ছেদ ৩.৫ নং দ্রষ্টব্য।

৩। **8051 মাইক্রোকন্ট্রোলারের প্রোগ্রামিং মডেল অঙ্কন করে সংক্ষেপে বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৩.২ নং দ্রষ্টব্য।

৪। **8051 মাইক্রোকন্ট্রোলারের মেমোরি গঠন বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৩.৩ নং দ্রষ্টব্য।

৫। **প্রোগ্রাম স্ট্যার্টাস ওয়ার্ড (পিএসডব্লিউ) রেজিস্ট্যারের বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৩.৪ নং দ্রষ্টব্য।

# অধ্যায় 8 ইন্টেল 8051 মাইক্রোকন্ট্রোলারের নির্দেশনার সেট
## Understand the Instruction Set of 8051 Microcontroller

### ৪.০ ভূমিকা
**(Introductin) :**

কতকগুলো বিটের অনুক্রম (Sequence) যা ডিজিটাল কম্পিউটারের কেন্দ্রীয় প্রক্রিয়াকরণ অংশকে কোন বিশেষ কার্য সম্পাদনের জন্য অবগত হওয়া হলো নির্দেশনা বা Instruction।

**ডাটা ট্রান্সফার ইন্সট্রাকশন (Data Transfer Instruction) :** যে সকল instruction এর মাধ্যমে computer এর কোন data কে computer এর মধ্যকার এক stage হতে অন্য stage এ (যেমনঃ এক register হতে অন্য register এ অথবা register হতে memory তে বা memory হতে register এ ইত্যাদি) কিংবা এক location হতে অন্য কোন memory Location- এ স্থানান্তরিত করা হয়। সে সকল instruction কে ডাটা transfer instruction বলে।

আর computer এর উক্ত প্রকার কাজ বা operation কে data transfer operation বলে। সাধারণত load বা move instruction ই হচ্ছে data transfer instruction, যেমন ঃ MOV B, A; MOV C, A ইত্যাদি। বিভিন্ন প্রকার ডাটা ট্রান্সফার অপারেশন রয়েছে। তা হল-

- ❑ রেজিস্টার হতে রেজিস্টার (Register to Register)
- ❑ ইমিডিয়েট ডাটা রেজিস্টার (Immediate data to Register)
- ❑ ইমিডিয়েট ডাটা মেমরিতে (Immediate data to Memory)
- ❑ রেজিস্টার হতে মেমরিতে (Register to Memory)
- ❑ মেমরি হতে রেজিস্টার (Memory to Register)
- ❑ বিনিময় (Exchange) ইত্যাদি।

**অ্যারিথমেটিক অপারেশন ইন্সট্রাকশন (Arithmatic Operation of Instruction) :** যে সকল instruction এর মাধ্যমে arithmatic কার্য্য (যেমনঃ addition, subtruction, increment এবং descriment ইত্যাদি) সম্পন্ন করা হয়, যে সকল instruction কে arithmatic instruction বলে। আর এই প্রকার কার্য্যকে arithmatic operation বলে। 8-bit arithmatic operation এ সাধারণত accumulator এর সহিত এবং result accumulator এ, তাছাড়া 16-bit operation HL register এর সহিত এবং result ও HL operation সম্পন্ন হওয়ার পর result ও HL register এ জমা হয়। যেমন : ADD B; ACI 90; SBB M ইত্যাদি। সাধারণতঃ নিম্নোক্ত গণিতিক অপারেশনসমূহ ইন্টেল ৮০৫৮ মাইক্রোপ্রসেসরে সংগঠিত হয়। যথা-

- ❑ যোগ/বিয়োগ-রেজিস্টার (Add/Subtract Register)
- ❑ যোগ/বিয়োগ-মেমরি (Add/Subtact Memory)
- ❑ যোগ/বিয়োগ-ইমিডিয়েট (Add/Subtract Immediate)
- ❑ যোগ/বিয়োগ-রেজিস্টার ক্যারি/বোরোসহ (Add/Subtract-Register with carry/borrow)
- ❑ যোগ/বিয়োগ-মেমরি ক্যারি/বোরোসহ (Add/Subtract-Immediate with carry/borrow)
- ❑ ইনক্রিমেন্ট/ডিক্রিমেন্ট-রেজিস্টার (Increment/Decrement–Register)
- ❑ ইনক্রিমেন্ট/ডিক্রিমেন্ট-মেমরি (Increment/Decrement–Memory)
- ❑ ইনক্রিমেন্ট/ডিক্রিমেন্ট-রেজিস্টার যুগল (Increment /Decrement–Register Pair)
- ❑ ডেসিমাল এডজাস্ট-অ্যাকিউমুলেটর (Decimal Adjust Accumulator, DAA)
- ❑ রেজিস্টার যুগল যোগ-(Add Register Pair) ইত্যাদি।

**লজিক ইন্সট্রাকশন (Logic Instruction) :** যে সকল instruction এর মাধ্যমে দুটি ডাটা এর মধ্যে OR, AND অথবা EX-OR ইত্যাদি কিংবা ডাটাকে compare করা কিংবা কোন ডাটাকে compliment ইত্যাদি করা যায় সে সকল instruction কে logic instruction বলে। Computer এ logic operation bit to bit ঘটে থাকে। যেমন : A = 10 11 এবং B = 1001 যদি হয় এবং এমতাবস্থায় যদি logical operation করতে বলা হয় A এর সহিত B এর ডাটাকে OR function তবে operation এর পরে result হবে 1011। যেমন : ANA B; ANA M; CPI 90; XRA M.

- ❑ AND /OR-Register
- ❑ AND /OR-Memory
- ❑ AND/OR-Immediate
- ❑ EXCLUSIVE-OR-Register/Memory
- ❑ EXCLUSIVE-OR-Immediate
- ❑ Compare Register/Memory
- ❑ Compare-Immediate
- ❑ Complemen-Accumulator
- ❑ Complement Carry
- ❑ Set Carry ইত্যাদি।

**রোটেট ইন্সট্রাকশন (Rotate Instruction) :** মাইক্রোপ্রসেসসেরর যখন accumulator content কে এক step বা bit ডানে বা বামে rotate করানো হয়, তখন তাকে rotate operation বলে। আর যে instruction এর মাধ্যমে এই rotate operation সম্পন্ন করা হয়, তাকে rotate instruction বলে।

Rotate operation carry bit সহ অথবা carry bit ছাড়া হতে পারে। যেমন :

1. Rotate left : RLC (Rotate left without carry)
2. Rotate right : RRC (Rotate right without carry)
3. Rotate All left : RAL (Rotate all left with carry)
4. Rotate Right : RAR (Rotate all right with carry)

**ব্রাঞ্চ ইন্সট্রাকশন (Branch Instruction) :** যে instruction এর মাধ্যমে কোন program এর ধারাবাহিক operation কে ব্যহত করে program এর কোন একটি memory location হতে পরবর্তী memory location এ না গিয়ে jump করে অন্য কোন memory location এ চলে যাওয়া হয় সে instruction কে branch instruction বলে।

Branch instruction দ্বারা সংঘটিত operation কে branch operation বলে। উক্ত instruction দ্বারা মূলতঃ program counter এর o/p কে পরিবর্তন করে branch instruction কে সম্পন্ন করা হয়। Branch instruction সাধারণত : দুই প্রকার হয়ে থাকে। যথা :

1. Un-conditional instruction. যেমন : JMP 3000
2. Conditional branch instruction. যেমন : JNZ (JPO/LM) ইত্যাদি।

**স্ট্যাক অ্যান্ড সাবরুটিন (Stack & Subroutine Instruction) :** Stack বলতে RAM এর একটি নির্ধারিত memory location কে বুঝায়, যেখানে 16-bit এর কোন ডাটা কে Store করে পরবর্তী ঐ ডাটাকে address number হিসেবে ব্যবহার করে বিভিন্ন ধরনের operation সম্পন্ন করা হয়। যাকে stack operation বলে। আর ঐ নির্ধারিত memory location যেখানে 16-bit এর address বা ডাটা store করা যায়, তাকে stack pointer বা stack register বলা হয়। Stack pointer subroutine operation এর সময় বিশেষ ভূমিকা পালন করে।

Subroutine operation এর সময় stack যে instruction এর মাধ্যমে নির্দিষ্ট memory location হতে ডাটা কে পয়েন্ট আউট করে ঐ instruction কে stack subroutine instruction বলে। যেমন : POP B/CALL/RZ/CNC.

## ৪.১ নির্দেশনা ও নির্দেশনা সেট

### (Define Instruction and Instruction Set)

কতকগুলো বিটের অনুক্রম (Sequence) যা ডিজিটাল কম্পিউটারের কেন্দ্রীয় প্রক্রিয়াকরণ অংশকে কোন বিশেষ কার্য সম্পাদনের জন্য অবগত হওয়া হলো নির্দেশনা বা Instruction। এটি কার্যসম্পাদনে ব্যবহৃত ডাটাও ধারণ করতে পারে। এটি প্রোগ্রামের একটি ছোট অংশ (Segment) যা কম্পিউটার প্রসেসর কর্তৃক সম্পাদিত কাজের জন্য প্রয়োজনীয় ধাপ ধারণ করে। এটি একটি অপারেশন কোড (OP Code) যা সম্পাদিত কাজকে নির্দিষ্ট করে এবং এক বা একাধিক অপারেন্ড (operand) বা তাদের ঠিকানা (address) এবং এক বা একাধিক modifier বা তাদের অ্যাড্রেস নিয়ে গঠিত।

কোন প্রোগ্রাম কর্তৃক কোন কম্পিউটার প্রসেসরে প্রদেয় একটি আদেশ (Command) হলো নির্দেশনা। কোন প্রসেসর কর্তৃক বোধগম্য সমস্ত নির্দেশনা বা কমান্ডসমূহকে একত্রে ইনস্ট্রাকশন সেট বলে। 8051 মাইক্রোকন্ট্রোলারের নির্দেশনা সেটে মোট 111টি নির্দেশনা আছে।

## ৪.২ 8051 এর নির্দেশনার শ্রেণিবিভাগ

### (Classify the 8051 Instruction)

নিচে 8051 এর নির্দেশনার শ্রেণিবিভাগ দেওয়া হলো :

(১) নির্দেশনার আকারের উপর ভিত্তি করে (On instruction size) :

- (i) One/single byte instruction.
- (ii) Two/double byte instruction.
- (iii) Three/triple byte instruction.

(২) কাজের বৈশিষ্ট্যের উপর ভিত্তি করে (on working characteristics) :

- (i) Arithmetic Instructions
- (ii) Branch Instructions
- (iii) Data Transfer Instructions
- (iv) Logic Instructions
- (v) Bit-oriented Instructions.

## ৪.৩ প্রত্যেকটি নির্দেশনার কার্যাবলি বর্ণনা

### (Describe the Function of Each Instruction)

**(1) এক বাইট নির্দেশনা :**

কোন নির্দেশনায় যদি অপারেন্ড না থাকে তবে নির্দেশনা (Instruction) সম্পূর্ণভাবে শুধুমাত্র এক বাইটের একটি অপকোড বা Instruction code দ্বারা গঠিত হয়। এই ধরনের নির্দেশনাকে এক বাইট নির্দেশনা (Instruction) বলে।

**(2) দুই বাইট নির্দেশনা :**

কোন নির্দেশনায় যদি এক বাইটের অপারেন্ড থাকে তবে তাকে দুই বাইট নির্দেশনা (Instruction) বলে।

নির্দেশনার 1ম বাইট – অপকোড (OPCODE) বা Instruction code

2য় বাইট – 2 বাইট বা 16 বিটের অপারেন্ডের 8 বিট MSB (Most Sinificant Bit)

8051 মাইক্রোকন্ট্রোলারের মোট 111টি নির্দেশনা রয়েছে। নিচে এদের আকারসহ বিভিন্ন নির্দেশনার কাজ বর্ণনা করা হলো :

**Arithmetic instructions :**

Arithmetic instruction বা গাণিতিক কিছু মৌলিক হিসাব-নিকাশ যেমন– যোগ, বিয়োগ, গুণ, ভাগ ইত্যাদি সম্পূর্ণ করে এবং কার্য শেষে ফলাফল ১ম অপারেন্ডে জমা করে। নিচে গাণিতিক নির্দেশনাসমূহ উল্লেখ করা হলো :

**ARITHMETIC INSTRUCTIONS**

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| ADD A, Rn | অ্যাকুমুলেটর ও রেজিস্টার ধারক যোগ করে। | 1 | 1 |
| ADD A, direct | অ্যাকুমুলেটরের ডাটার সাথে Direct Byte কে যোগ করে। | 2 | 2 |
| ADD, A, @Ri | অ্যাকুমুলেটরের সাথে Indirect RA M এর তথ্যের সাথে যোগ করে। | 1 | 1 |
| ADD A, #data | Immediate data এর সাথে অ্যাকুমুলেটরের ডাটার যোগ করে। | 2 | 2 |
| ADDC A, Rn | Carry flag সহ রেজিস্টারের ডাটার সাথে অ্যাকুমুলেটরের ডাটার যোগ করে। | 1 | 1 |
| ADDC A, direct | Carry flag সহ অ্যাকুমুলেটরের ডাটার সাথে Direct Byte কে যোগ করে। | 2 | 2 |
| ADDC A, @Ri | Carry flag সহ অ্যাকুমুলেটরের সাথে Indirect RAM এর তথ্যের সাথে যোগ করে। | 1 | 2 |
| ADDC A, #data | Carry flag সহ Immediate data এর সাথে অ্যাকুমুলেটরের ডাটার যোগ করে। | 2 | 2 |
| SUBB A, Rn | Borrow সহ অ্যাকুমুলেটর হতে রেজিস্টারের তথ্য বিয়োগ করে। | 1 | 1 |
| SUBB A, direct | Borrow সহ অ্যাকুমুলেটর হতে direct byte কে বিয়োগ করে। | 2 | 2 |
| SUBB A, @Ri | Borrow সহ অ্যাকুমুলেটর হতে Indirect RAM এর তথ্যকে বিয়োগ করে। | 1 | 2 |
| SUBB A, #data | Borrow সহ অ্যাকুমুলেটর হতে immediate data কে বিয়োগ করে। | 2 | 3 |
| INC A | অ্যাকুমুলেটরের মানকে 1 বৃদ্ধি করে। | 1 | 1 |
| INC Rn | রেজিস্টারের মানকে 1 বৃদ্ধি করে। | 1 | 2 |
| INC Rx | Direct byte এর মানকে 1 বৃদ্ধি করে। | 2 | 3 |
| INC @Ri | Indirect RAM এর মানকে 1 হ্রাস করে। | 2 | 3 |
| INC DPTR | Data Pointer এর মানকে 1 বৃদ্ধি করে। | 1 | 3 |
| MUL AB | A ও B কে গুণ করে। | 1 | 5 |
| DIV AB | A কে B দ্বারা ভাগ করে। | 1 | 5 |
| DAA | BCD code মোতাবেক অ্যাকুমুলেটরের দশমিক মানকে বিন্যস্ত করে। | 1 | 1 |

**Branc instructions :**

8051 মাইক্রোকন্ট্রোলারে দুই ধরনের Branch instruction আছে।

(i) **Unconditional jump instructions :** কোনরূপ শর্ত ব্যতিরেকে এই ধরনের নির্দেশনা সমাপ্তনান্তে প্রোগ্রাম সম্পাদনের বর্তমান অবস্থান থেকে একটি নতুন অবস্থানে স্থানান্তরিত হয়।

(ii) **Conditional jump instructions :** এতে একটি নির্দিষ্ট শর্ত পূরণ হলে প্রোগ্রাম একটি নতুন অবস্থানে স্থানান্তরিত হয়, অন্যথায় প্রোগ্রাম স্বাভাবিকভাবে পরবর্তী নির্দেশনায় অগ্রসর হয়। নিচে নির্দেশনাসমূহ দেখানো হলো।

**BRANCH INSTRUCTIONS**

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| ACALL addr11 | Absolute subroutine আহ্বান (Call) | 2 | 6 |
| LCALL addr16 | Long subroutine আহ্বান (Call) | 3 | 6 |
| RET | Sub routine থেকে প্রত্যাবর্তন (Returns) | 1 | 4 |
| RETI | == subroutine থেকে প্রত্যাবর্তন (Returns) | 1 | 4 |
| AJMP addr11 | Absolute jump | 2 | 3 |
| LJMP addr 16 | Long jump | 3 | 4 |
| SJMP rel | Short jump (নিম্নবর্ণিত নির্দেশনা সাপেক্ষে 128 থেকে + 127 স্থানে) | 2 | 4 |
| JC rel | Carry flag সেট (set) হলে jump করে। (Short jump.) | 2 | 3 |
| JC rel | Carry flag সেট (set) হলে jump করে। (Short jump.) | 2 | 3 |
| JB bit, rel | Direct bit সেট (set) হলে jump করে। (Short jump.) | 3 | 4 |
| JBC bite, rel | Direct bit সেট (set) এবং খালি (Clear) হলে Jump করে। (Short jump) | 3 | 4 |
| JMP @ A+DPTR | ডাটা পয়েন্টারের সাপেক্ষে পরোক্ষ Jump করে। | 1 | 2 |
| JZ rel | অ্যাকুমুলেটর শূন্য হলে Jump করে। (Short jump) | 2 | 3 |
| JNZ rel | অ্যাকুমুলেটর শূন্য না হলে Jump করে। (Short jump) | 2 | 3 |
| CJNE A, direct,rel | Direct byte কে অ্যাকুমুলেটরের সাথে তুলা করে এবং সমান না হলে Jump করে। Short jump. | 3 | 4 |
| CJNE A, #data, rel | Immediate data কে অ্যাকুমুলেটরের সাথে তুলনা করে এবং সমান না হলে Jump করে। Short jump. | 3 | 4 |

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| CJNE Rn, #data, rel | Immediate data কে রেজিস্টারের সাথে তুলনা করে এবং সমান না হলে Jump করে। Short jump. | 3 | 4 |
| CJNE @Ri, #data, rel | Immediate data কে Indirect register এর সাথে তুলনা করে এবং সমান না হলে Jump করে। Short jump. | 3 | 4 |
| DJNZ Rn, rel | রেজিস্টারের মান হ্রাস করে এবং শূন্য না হলে Jump করে। Short jump. | 2 | 3 |
| DJNZ Rx, rel | Direct byte কে স্থাপন করে এবং শূন্য না হলে Jump করে। Short jump | 3 | 4 |
| NOP | কোন কাজ করে না (No operation) | 1 | 1 |

**Data transfer instructions :**

Data transfer instruction বা ডাটা স্থানান্তর নির্দেশনাসমূহ এক রেজিস্টার থেকে অন্য রেজিস্টারে তথ্য স্থানান্তর করে। যে রেজিস্টারের তথ্য স্থানান্তরিত হয় তা অপরিবর্তিত থাকে। যদি নির্দেশনায় X প্রত্যয় (Suffix) থাকে (যেমন, MOVX) তবে ডাটা বাহ্যিক মেমোরি থেকে বিনিময় বুঝায়। নিচে Data transfer instruction দেখানো হলো।

**DATA TRANSFER INSTRUCTIONS**

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| MOV A, Rn | রেজিস্টারের তথ্যকে অ্যাকুমুলেটরে স্থানান্তর করে। | 1 | 1 |
| MOV A, direct | Direct বাইটকে অ্যাকুমুলেটরে স্থানান্তর করে। | 2 | 2 |
| MOV A, @Ri | Indirect RAM থেকে তথ্যকে অ্যাকুমুলেটরে স্থানান্তর করে। | 1 | 2 |
| MOV A, #data | Immediate data কে অ্যাকুমুলেটরে স্থানান্তর করে। | 2 | 2 |
| MOV Rn, A | অ্যাকুমুলেটরের তথ্যকে রেজিস্টারে স্থানান্তর করে। | 1 | 2 |
| MOV Rn, direct | Direct byte কে রেজিস্টারে স্থানান্তর করে। | 2 | 4 |
| MOV Rn, #data | Immediate data কে রেজিস্টারে স্থানান্তর করে। | 2 | 2 |
| MOV direct, A | অ্যাকুমুলেটরের তথ্যকে direct byte এ স্থানান্তর করে। | 2 | 3 |
| MOV direct, Rn | রেজিস্টারের তথ্যকে direct byte এ স্থানান্তর করে। | 2 | 3 |
| MOV direct, direct | Direct byte এর তথ্যকে direct byte এ স্থানান্তর করে। | 3 | 4 |
| MOV direct, @Ri | Indirect RAM এর তথ্যকে direct byte এ স্থানান্তর করে। | 2 | 4 |
| MOV direct, #data | Immediate data কে direct byte এ স্থানান্তর করে। | 3 | 3 |
| MOV @Ri, A | অ্যাকুমুলেটরের তথ্যকে indirect RAM এ স্থানান্তর করে। | 1 | 3 |

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| MOV DPTR,#data | 16 বিটের ডাটাকে ডাটা পয়েন্টারে স্থানান্তর করে। | 3 | 3 |
| MOVC A,@A+DPTR | ডাটা পয়েন্টার সম্পর্কিত কোড ডাটাকে অ্যাকুমুলেটরে স্থানান্তরিত করে। (address = A + DPTR) | 1 | 3 |
| MOVC A, @A+PC | Program counter (PC) সম্পর্কিত কোড বাইটকে অ্যাকুমুলেটরে স্থানান্তরিত করে। (address=A+PC) | 1 | 3 |
| MOVX A, @Ri | বাহ্যিক (External) RAM (8-bit address) এর ডাটাকে অ্যাকুমুলেটরে স্থানান্তরিত করে। | 1 | 3-10 |
| MOVX A, @DPTR | বাহ্যিক (External) RAM (16-bit address) এর ডাটাকে অ্যাকুমুলেটরে স্থানান্তরিত করে। | 1 | 3-10 |
| MOVX @Ri, A | অ্যাকুমুলেটরের ডাটাকে বাহ্যিক (External) RAM (8-bit address) এ স্থানান্তরিত করে। | 1 | 4-11 |
| MOVX @DPTR, A | অ্যাকুমুলেটরের ডাটাকে বাহ্যিক (External) RAM (16-bit address) এ স্থানান্তরিত করে। | 1 | 4-11 |
| PUSH direct | Direct byte কে stack এর উপর PUSH করা হয়। | 2 | 4 |
| POP direct | stack থেকে Direct byte কে POP করা হয়। | 2 | 3 |
| XCH A, Rn | অ্যাকুমুলেটরের সাথে রেজিস্টার তথ্য বিনিময় করে। | 1 | 2 |
| XCH A, direct | অ্যাকুমুলেটরের সাথে Indirect RAM এর তথ্য বিনিময় করে। | 1 | 3 |
| XCHD A, @Ri | অ্যাকুমুলেটরের সাথে Low-order nibble indirect RAM এর তথ্য বিনিময় করে। | 1 | 3 |

**Logic instructions :**

Logic instruction বা যৌক্তিক নির্দেশনা দুটি রেজিস্টারে রক্ষিত বিটের উপর লজিক কাজ সম্পন্ন করে এবং নির্দেশনা সমাপনান্তে ফলাফল প্রথম অপারেন্ডে জমা করে। নিচে যৌক্তিক নির্দেশনাসমূহ দেখানো হলো।

**LOGIC INSTRUCTIONS**

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| ANL A, Rn | রেজিস্টারের তথ্যের সাথে অ্যাকুমুলেটরের তথ্যের AND করে। | 1 | 1 |
| ANL A, direct | Direct byte এর সাথে অ্যাকুমুলেটরের তথ্যের AND করে। | 2 | 2 |
| ANL A, @Ri | Indirect RAM এর সাথে অ্যাকুমুলেটরের তথ্যের AND করে। | 1 | 2 |
| ANL A, #data | Immediate data এর সাথে অ্যাকুমুলেটরের তথ্যের AND করে। | 2 | 2 |
| ANL direct, A | অ্যাকুমুলেটরের তথ্যকে Direct byte এর সাথে AND করে। | 2 | 3 |

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| ANL direct, #data | Immediate data কে Direct register এর সাথে AND করে। | 3 | 4 |
| ORL A, Rn | রেজিস্টারের তথ্যের সাথে অ্যাকুমুলেটরের তথ্যের OR করে। | 1 | 1 |
| ORL A, direct | Direct byte কে অ্যাকুমুলেটরের তথ্যের সাথে OR করে। | 2 | 2 |
| ORL A, @Ri | Indirect RAM কে অ্যাকুমুলেটরের তথ্যে সাথে OR করে। | 1 | 2 |
| ORL direct, A | অ্যাকুমুলেটরের তথ্যের সাথে Direct byte কে OR করে। | 2 | 3 |
| ORL direct, #data | Immediate data কে direct byte এর সাথে OR করে। | 3 | 4 |
| XRL A, Rn | রেজিস্টারের তথ্যের সাথে অ্যাকুমুলেটরের তথ্যের Exclusive OR করে। | 1 | 1 |
| XRL A, direct | Direct byte কে অ্যাকুমুলেটরের তথ্যের সাথে Exclusive OR করে। | 2 | 2 |
| XRL A, @Ri | Indirect RAM কে অ্যাকুমুলেটরের তথ্যের সাথে Exclusive OR করে। | 1 | 2 |
| XRL A, #data | Immediate data কে অ্যাকুমুলেটরের তথ্যের সাথে Exclusive OR করে। | 2 | 2 |
| XRL direct, A | অ্যাকুমুলেটর তথ্যের সাথে direct byte কে Exclusive OR করে। | 2 | 3 |
| XORL direct, #data | Immediate data কে direct byte এর সাথে Exclusive OR করে। | 3 | 4 |
| CLR A | অ্যাকুমুলেটর তথ্যকে মুছে ফেলে (Clear)। | 1 | 1 |
| CPL A | অ্যাকুমুলেটরের তথ্যকে Complements (1 = 0, 0 = 1) করে। | 1 | 1 |
| SWAP A | অ্যাকুমুলেটরের অভ্যন্তরীণ nibble কে অদলবদল করে। | 1 | 1 |
| RL A | অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে বামদিকে আবর্তিত করে। | 1 | 1 |
| RLC A | অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে Carry সহ বামদিকে চক্রাকারে আবর্তিত করে। | 1 | 1 |
| RR A | অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে ডানদিকে চক্রাকারে আবর্তিত করে। | 1 | 1 |
| RRC A | অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে Carry সহ ডানদিকে চক্রাকারে আবর্তিত করে। | 1 | 1 |

**Bit-oriented instructions :**

এরা যৌক্তিক নির্দেশনারই মতো লজিক কাজ সম্পাদন করে, তবে পার্থক্য হলো এরা একক বিটের উপর সম্পাদিত হয়। নিচে bit-oriented নির্দেশনাসমূহ দেখানো হলো।

**BIT-ORIENTED INSTRUCTIONS**

| নেমনিক (Mnemonic) | বর্ণনা | বাইট | সাইকেল |
|---|---|---|---|
| CLR C | carry flag কে মুছে ফেলে (Clear)। | 1 | 1 |
| CLR bit | Direct bit কে মুছে ফেলে (Clear)। | 2 | 3 |
| SETB C | carry flag কে Set করে। | 1 | 1 |
| SETB bit | Direct bit কে Set করে। | 2 | 3 |
| CPL C | Carry flag কে Complement করে। | 1 | 1 |
| CPL bit | Drect bit কে Complement করে। | 2 | 3 |
| ANL C, bit | Direct bit কে carry flag এর সাথে AND করে। | 2 | 3 |
| ANL C,/bit | Direct bit এর complement কে carry flag এর সাথে OR করে। | 2 | 2 |
| ORL C, bit | Direct bit কে OR carry flag এর সাথে OR করে। | 2 | 2 |
| ORL C,/bit | Direct bit এর complement কে carry flag এর সাথে OR করে। | 2 | 2 |
| MOV C, bit | Direct bit কে carry flag এ স্থানান্তর করে। | 2 | 2 |
| MOV bit, C | Carry flag কে direct bit এ স্থানান্তর করে। | 2 | 3 |

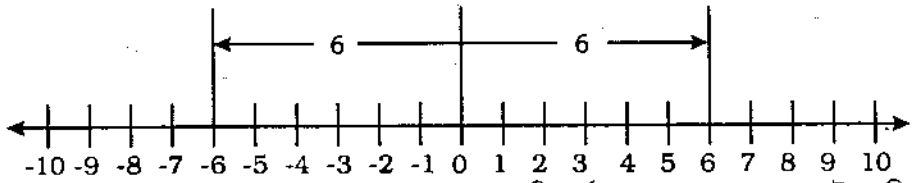
## ৪.৪ শর্ট রেঞ্জ, অ্যাবসলিউট ভ্যালু এবং লং রেঞ্জ আলোচনা

## (Discuss Short, Absolute and Long Range)

কোন রাশির সর্বোচ্চ ও সর্বনিম্ন মানের পার্থক্যকে পাল্লা (Range) বলে। কোন একটি ফাংশনের সমস্ত আউটপুট মানকেও Range বলে। (4, 6, 9, 3, 7) এর সর্বনিম্ন মান হলো 3 এবং সর্বোচ্চ মান 9, সুতরাং এর Range হলো 9 – 3 = 6।

(1) **শর্টরেঞ্জ (Short range) :** এটি একটি পূর্ণসংখ্যা যা একই মেশিনে কোন একটি আদর্শ পূর্ণসংখ্যার তুলনায় ক্ষুদ্র range এর কারণে কম জায়গা (storage) দখল করে।

(2) **অ্যাবসলিউট ভ্যালু (Absolute value) :** কোন সংখ্যার পরম মান বা absolute value হলো শূন্য থেকে ঐ সংখ্যার দূরত্ব। নিচের গ্রাফে 0 থেকে 6 এর দূরত্ব সবসময় 6 এবং 0 থেকে – 6 এর দূরত্বও 6। সুতরাং 6 ও – 6 এর পরম মান 6। পরম মান সবসময় ধনাত্মক।




(3) **লংরেঞ্জ (Long range) :** Long range হলো একটি পূর্ণসংখ্যা যার range একই মেশিনে একটি আদর্শ পূর্ণসংখ্যার সমান বা বড়। এটি সাধারণত একটি আদর্শ পূর্ণসংখ্যার তুলনায় দ্বিগুণ জায়গা (storage) দখল করে।

## ৪.৫ 8051 এর অ্যাড্রেসিং মুড ব্যাখ্যা

## (Explain the Addressing Modes of 8051)

নির্দেশনা একটি অপারেশন কোড (OP Code), যা সম্পাদিত কাজকে নির্দিষ্ট করে এবং এক বা একাধিক অপারেন্ড (operand) বা তাদের ঠিকানা (address) এবং এক বা একাধিক modifier বা তাদের অ্যাড্রেস নিয়ে গঠিত। অপারেন্ড হিসেবে নিম্নলিখিত রাশিগুলো ব্যবহৃত হয় :

১। একটি সিপিইউ রেজিস্টার (CPU register)
২। একটি মেমোরি স্থান (Memory location)
৩। ডাটা নিজেই (Data value itself)
৪। একটি ইনপুট/আউটপুট পোর্ট (I/O port)।

নির্দেশনা অপারেন্ডসমূহকে সুনির্দিষ্ট ও সুস্পষ্টভাবে প্রকাশ করাকে অ্যাড্রেসিং মুড (Addressing mode) বলে। 8051 মাইক্রোকন্ট্রোলার নিম্নবর্ণিত অ্যাড্রেসিং মুড সমর্থন করে :

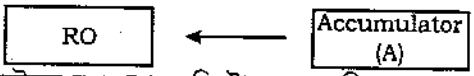
(i) Register Addressing mode
(ii) Direct Addressing mode
(iii) Register Indirect Addressing mode
(iv) Immediate Addressing mode
(v) Relative Addressing mode
(vi) Absolute Addressing ode
(vii) Long Addressing mode
(viii) Base Ressister plus Index Register Addressing.

**রেজিস্টার অ্যাড্রেসিং মুড (Register addressing mode) :**

রেজিস্টার অ্যাড্রেসিং মুডে নির্দেশনা রেজিস্টারসমূহ (যথা : নির্বাচিত ব্যাংকের রেজিস্টার, R0–R7, অ্যাকুমুলেটর, B রেজিস্টার ইত্যাদি) এর মধ্যে তথ্য স্থানান্তর করার কাজে সংশ্লিষ্ট।

**উদাহরণ :**

MOV RO, A



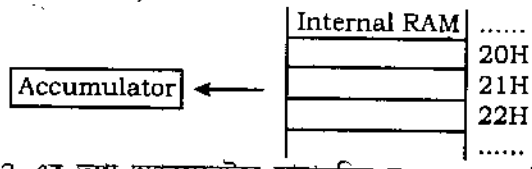

এই নির্দেশনা শেষে অ্যাকুমুলেটরের তথ্য R0 রেজিস্টারে স্থানান্তরিত হয়। তবে এই নির্দেশনার আগেই রেজিস্টার ব্যাংক (Bank 0, 1, 2 বা 3) অবশ্যই নির্বাচন করতে হবে। অ্যাকুমুলেটরকে A রেজিস্টার বলে। এই অ্যাড্রেসিং মোডের একটি সুবিধা হলো– এটি একক বাইটের একটি Short নির্দেশনা হয়।

**ডাইরেক্ট অ্যাড্রেসিং (Direct addressing) :**

এই অ্যাড্রেসিং মুডে source operand এর প্রকৃত ঠিকানা (Hexadecimal) সরাসরি বা এর সংক্ষিপ্ত নামের মাধ্যমে নির্দিষ্ট করা হয়। এতে অভ্যন্তরীণ RAM এবং SFR এর নিচের 128 বিট (Lower 128 bit) ব্যবহৃত হয়। এটি সাধারণত অভ্যন্তরীণ RAM ও SFR access এর জন্য ব্যবহৃত হয়। এতে অভ্যন্তরীণ RAM এবং SFR এর Lower 128 bit ব্যবহৃত হয়।

**উদাহরণ :**

MOV A, P3
MOV A, 021H



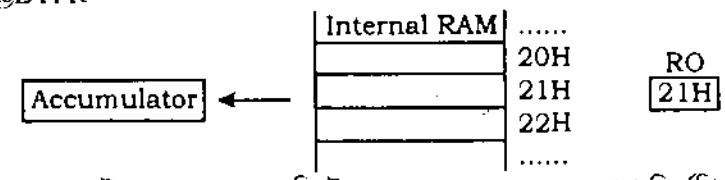

১ম নির্দেশনায় PORT 3 এর তথ্য অ্যাকুমুলেটরে স্থানান্তরিত হয় এবং ২য় নির্দেশনায় 21H ঠিকানার তথ্য অ্যাকুমুলেটরে স্থানান্তরিত হয়। প্রথমটিতে source operand এর সংক্ষিপ্ত নাম P3 এবং ২য়টিতে সরাসরি মেমোরি ঠিকানা ব্যবহৃত হয়।

**রেজিস্টার ইনডাইরেক্ট অ্যাড্রেসিং (Register Indirect Addressing) :**

এই মুডে অপারেন্ডের কার্যকরী ঠিকানা ধারণের জন্য একটি পয়েন্টার ব্যবহৃত হয়। পয়েন্টার রেজিস্টার হিসেবে এতে শুধুমাত্র রেজিস্টার R0, R1 ও DPTR ব্যবহৃত হয়। R0 ও R1 রেজিস্টার 8 বিটের ঠিকানা ধারণ করতে পারে এবং DPTR 16 বিটের ঠিকানা ধারণ করতে পারে। এটি উপরের ডাটা মেমোরি ক্ষেত্র (upper data memor area) এর জন্য ব্যবহৃত হয়। বাহ্যিক মেমোরি থেকে অপারেন্ড access করার জন্য DPTR কার্যকর। @ চিহ্ন দ্বারা Register Indirect Addressing mode নির্দেশ করে।

**উদাহরণ :**

MOV A, @R0,
MOVX A, @DTPR




১ম নির্দেশনার অ্যাকুমুলেটর (A) কে R0 রেজিস্টারের আধেয় (Content) দ্বারা নির্দেশিত মেমোরি স্থানের তথ্য দ্বারা পূরণ করে। R0 রেজিস্টারে 8-বিটের একটি ঠিকানা (21H) থাকে।

২য় নির্দেশনায় DPTR-এর আধেয় 16 বিটের মেমোরি ঠিকানা (যথা : 1234H) নির্দেশিত তথ্য অ্যাকুমুলেটরে জমা করে।

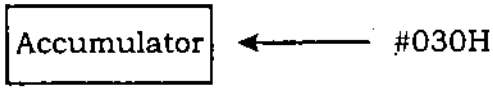
**ইমিডিয়েট অ্যাড্রেসিং মুড (Immediate Addressing Mode) :**

এটিকে Immediate constant addressing mode-ও বলে। এই মুডে source operand হিসেবে 8 বিট বা 16 বিটের ধ্রুবক (Constant value) ব্যবহৃত হয়। কোন রেজিস্টার বা মেমোরি স্থান দ্বারা নয় এবং নির্দেশনাতেই এই ধ্রুবককে নির্দিষ্ট করা হয়। গন্তব্য রেজিস্টার (destination register) একই ডাটা আকার (data size) ধারণ করে যা source operand দ্বারা নির্দিষ্ট করা হয়।

**উদাহরণ :**

ADD A, # 030H
MOV DPTR, # FE00H

১ম নির্দেশনায় এর 8 বিটের মানকে (30H) অ্যাকুমুলেটরের তথ্যের সাথে যোগ করে, অ্যাকুমুলেটর 8 বিটের রেজিস্টার। ২য় নির্দেশনায় 16 বিটের মানকে (0FE00) 16 বিটের মানকে (DPTR) এ স্থানান্তর করে। # চিহ্ন দ্বারা অ্যাসেম্বলার বুঝতে পারে যে, নির্দেশনাটিতে Immediate addressing mode ব্যবহার করা হয়েছে।




**রিলেটিভ অ্যাড্রেসিং মুড (Relative addressing mode) :**

রিলেটিভ অ্যাড্রেসিং মুড কিছু Jump Instruction যথা– SJMP, JNZ ইত্যাদি এর সাথে ব্যবহৃত হয়। এই নির্দেশনাসমূহ প্রোগ্রামের এক অংশ থেকে অন্য অংশে নিয়ন্ত্রণ স্থানান্তর করে। Relative address (প্রায়শই offset হিসেবে পরিচিত) একটি 8 বিট sign number যা স্বয়ংক্রিয়ভাবে program counter এর সাথে যুক্ত হয়ে পরবর্তী নির্দেশনার ঠিকানা উৎপন্ন করে। যেহেতু 8 বিটের offset ($2^8$ = 256) ব্যবহৃত হয় তাই গন্তব্য ঠিকানা অবশ্যই বর্তমান নির্দেশনার ঠিকানা থেকে – 128 ও + 128 এর মধ্যে থাকে।

**উদাহরণ :**

```
GOBACK :   DEC   A
JNZ        GOBACk
```

১ম নির্দেশনায় অ্যাকুমুলেটরের (A) মান 1 হ্রাস করে। ২য় নির্দেশনায় A এর মান শূন্য না হলে GOBACK-এ JUMP করে।

**অ্যাবসুলিউট অ্যাড্রেসিং মুড (Absolute addressing mode) :**

Absolute addressing Mode এ একটি লেবেল কর্তৃক absolute address, যাতে প্রোগ্রামের নিয়ন্ত্রণ স্থানান্তরিত হয়, নির্দিষ্ট হয়। এই অ্যাড্রেসিং মুডে ACALL ও ADJUMP নির্দেশনাদ্বয় সংশ্লিষ্ট। নির্দেশনাদ্বয়ের আকার 2 বাইট যেখানে অপারেন্ড হিসেবে 11 বিট absolute address নির্দিষ্ট করা হয়। 16 বিট Program counter (PC) এর Upper 5 bit পরিবর্তন হয় না বরং Lower 11 bit এই নির্দেশনা থেকে বোঝাই হয়। তাই Branch address অবশ্যই প্রোগ্রাম মেমোরির 2KB ($2^{11}$ = 2048) এ থাকতে হবে।

**উদাহরণ :**

```
ACALL          PORT_INIT
PORT_INIT : MOV          P0, #0FH
```

১ম নির্দেশনা দ্বারা বুঝায় PORT_INIT অবশ্যই 2KB তে বোঝাই হতে হবে। ২য়টি PORT_INIT নামের একটি সাবরুটিন।

**লং অ্যাড্রেসিং মুড (Long addressing mode) :**

এই অ্যাড্রেসিং মুড LCALL ও LJUMP নির্দেশনাদ্বয়ের সাথে ব্যবহৃত হয়। এটি একটি 3 বাইট নির্দেশনা এবং শেষের 2 বাইট 16 বিটের একটি গন্তব্য স্থান নির্দিষ্ট স্থান করে যেখানে প্রোগ্রাম শাখায়িত (Program branching) হয়। ফলে এতে সম্পূর্ণ 64 কিলোবাইট code space ব্যবহৃত হয়। এতে পূর্ববর্তী অবস্থান বিবেচনা না করেই প্রোগ্রাম সবসময় এই স্থানে শাখায়িত (Branch) করে।

```
LCALL          TIMER_INIT
TIMER_INIT :   ORL        TMOD, #01H
```

১ম নির্দেশনায় TIMER_INIT address (16 bit long) কে অপারেন্ড হিসেবে নির্দিষ্ট করা হয়। ২য়টি একটি TIMER_INIT নামের সাবরুটিন।

**বেস রেজিস্টার প্লাস ইনডেক্স রেজিস্টার অ্যাড্রেসিং (Base Resister Plus Index Register Addressing) :**

যখন কোন Look-up Table (LUT) থেকে কোন ডাটা ফিরে পাওয়ার (retrieve) প্রয়োজন হয় তখন এই অ্যাড্রেসিং মুড উপকারী। 16 বিটের একটি রেজিস্টার (DPTR/PC) Base address ধারণ করে এবং অ্যাকুমুলেটর 8 বিটের Index মান ধারণ করে। এই দুই রেজিস্টারের সমষ্টি নির্দেশনার কার্যকরী ঠিকানা তৈরি করে। 8051 এর Index addressing সাধারণত JMP ও MOVC নির্দেশনার সাথে ব্যবহৃত হয়।

```
MOV            A, #08H
MOV            DPTR, #01F00H
MOVC           A, @A + DPTR
```

উপরের নির্দেশনাসমূহ সমাপ্তনান্তে প্রোগ্রাম কার্যকরী ঠিকানা 1F08H (1F00H + 08H) শাখায়িত (Branch) করে এবং ঐ ঠিকানা থেকে ডাটা অ্যাকুমুলেটরে স্থানান্তরিত হয়।

# অনুশীলনী-০৪

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **DJNZ R5 ইন্সট্রাকশন কী?** *[বাকাশিবো : '১৫]*

**উত্তর :** রেজিস্টারের মান হ্রাস করে এবং শূন্য না হলে Jump করে। Short jump.

২। **ইনস্ট্রাকশন বা নির্দেশনা বলতে কী বুঝ?**

**উত্তর :** নির্দেশনা হলো কতকগুলো বিটের অনুক্রম (Sequence) যা ডিজিটাল কম্পিউটারের কেন্দ্রীয় প্রক্রিয়াকরণ অংশকে কোন বিশেষ কার্যসম্পাদনের জন্য অবগত করে। এটি কার্যসম্পাদনে ব্যবহৃত ডাটাও ধারণ করতে পারে। এটি প্রোগ্রামের একটি ছোট অংশ (Segment) যা কম্পিউটার প্রসেসর কর্তৃক সম্পাদিত কাজের জন্য প্রয়োজনীয় ধাপ ধারণ করে।

৩। **নির্দেশনা সেট বলতে কী বুঝ?** *[বাকাশিবো : '১৪]*

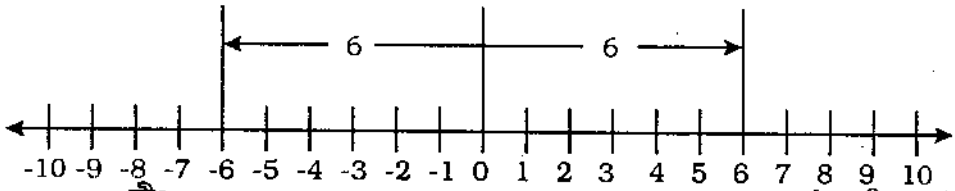
**উত্তর :** নির্দেশনা কোন প্রোগ্রাম কর্তৃক কোন কম্পিউটার প্রসেসরে প্রদেয় একটি আদেশ (Command)। কোন প্রসেসর কর্তৃক বোধগম্য সমস্ত নির্দেশনা বা কমান্ডসমূহকে একত্রে ইনস্ট্রাকশন সেট বলে। 8051 মাইক্রোকন্ট্রোলারের নির্দেশনা সেটে মোট 111টি নির্দেশনা আছে।

৪। **RLA নির্দেশনা দ্বারা কী বুঝায়?** *[বাকাশিবো : '১৪R]*

**উত্তর :** RLA নির্দেশনা অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে বামদিকে চক্রাকারে আবর্তিত করে।

৫। **Absolute value কী?** *[বাকাশিবো : '১৪R]*

**উত্তর :** কোন সংখ্যার পরম মান বা absolute value হলো শূন্য থেকে ঐ সংখ্যার দূরত্ব। নিচের গ্রাফে 0 থেকে 6 এর দূরত্ব সবসময় 6 এবং 0 থেকে – 6 এর দূরত্বও 6। সুতরাং 6 ও – 6 এর পরম মান 6। পরম মান সবসময় ধনাত্মক।




৬। **Long range কী?** *[বাকাশিবো : '১৪, '১৪R]*

**উত্তর :** Long range হলো একটি পূর্ণসংখ্যা যার range একই মেশিনে একটি আদর্শ পূর্ণসংখ্যার সমান বা বড়। এটি সাধারণত একটি আদর্শ পূর্ণসংখ্যার তুলনায় দ্বিগুণ জায়গা (Storage) দখল করে।

৭। **আকারের উপর ভিত্তি করে নির্দেশনাকে কয় ভাগে ভাগ করা যায় ও কী কী?**

**উত্তর :** নির্দেশনার আকার (Instruction size) এর উপর ভিত্তি করে–

(ক) One/single byte instruction
(খ) Two/double byte instruction
(গ) Three/triple byte instruction.

৮। **কাজের বৈশিষ্ট্যের উপর ভিত্তি করে 8051 মাইক্রোকন্ট্রোলের নির্দেশনাকে কয় ভাগে ভাগ করা হয় ও কী কী?**

**উত্তর :** কাজের বৈশিষ্ট্যের উপর ভিত্তি করে– *[বাকাশিবো : '১৪]*

(ক) Arithmetic instructions
(খ) Branch instructions
(গ) Data transfer instructions
(ঘ) Logic instructions
(ঙ) Bit-oriented instructions.

**৯। এক বাইট নির্দেশনা কাকে বলে? উদাহরণ দাও।**

**উত্তর :** যদি নির্দেশনায় কোন অপারেন্ড না থাকে তবে নির্দেশনা সম্পূর্ণভাবে শুধুমাত্র এক বাইটের একটি অপকোড বা Instruction code দ্বারা গঠিত হয়। এ ধরনের নির্দেশনাকে এক বাইট নির্দেশনা বলে :

**১০। দুই বাইট নির্দেশনা কাকে বলে? উদাহরণ দাও।**

**উত্তর :** যদি নির্দেশনায় এক বাইটের অপারেন্ড থাকে তবে তাকে দুই বাইট নির্দেশনা বলে।

নির্দেশনার ১ম বাইট – অপকোড (OPCODE) বা Instruction code

২য় বাইট – 8 বিট ডাটা বা ইনপুট/আউটপুট পোর্ট (Operand)

**১১। তিন বাইট নির্দেশনা কাকে বলে? উদাহরণ দাও।**

**উত্তর :** যদি কোন নির্দেশনায় 2 বাইটের অপারেন্ড তাকে তবে এটিকে তিন বাইট নির্দেশনা বলে।

নির্দেশনার ১ম বাইট – অপকোড (OPCODE) বা Instruction code

২য় বাইট – 2 বাইট বা 16 বিটের অপারেন্ডের 8 বিট LSB (Least Significant Bit)

৩য় বাইট – 1 বাইট বা 16 বিটের অপারেন্ডের 8 বিট MSB (Most Significant Bit)

**১২। INC A নির্দেশনা দ্বারা কী বুঝায়?**

**উত্তর :** INC A নির্দেশনা অ্যাকুমুলেটরের মানকে 1 বৃদ্ধি করে।

**১৩। CLR A নির্দেশনা দ্বারা কী বুঝায়?**

**উত্তর :** CLR A নির্দেশনা অ্যাকুমুলেটর তথ্যকে মুছে ফেলে (Clear)।

**১৪। RRC A নির্দেশনা দ্বারা কী বুঝায়?**

**উত্তর :** RRC A দ্বারা অ্যাকুমুলেটরের অভ্যন্তরীণ বিটগুলোকে carry সহ ডানদিকে চক্রাকারে আবর্তিত করে।

**১৫। SETB C নির্দেশনা দ্বারা কী বুঝায়?**

**উত্তর :** SETB C নির্দেশনা carry flag কে Set করে।

**১৬। Short Range কী?**

**উত্তর :** Short range হলো একটি পূর্ণসংখ্যা যা একই মেশিনে কোন একটি আদর্শ পূর্ণসংখ্যার তুলনায় ক্ষুদ্র range এর কারণে কম জায়গা (storage) দখল করে।

**১৭। অ্যাড্রেসিং মুড কাকে বলে?**

**উত্তর :** নির্দেশনা একটি অপারেশন কোড (OP Code), যা সম্পাদিত কাজকে নির্দিষ্ট করে এবং এক বা একাধিক অপারেন্ড (Operand) বা তাদের ঠিকানা (address) এবং এক বা একাধিক modifier বা তাদের অ্যাড্রেস নিয়ে গঠিত। নির্দেশনার অপারেন্ডসমূহকে সুস্পষ্টভাবে প্রকাশ করাকে অ্যাড্রেসিং মুড বলে।

**১৮। Register indirect addressing mode এ পয়েন্টার হিসেবে কোন কোন রেজিস্টার ব্যবহৃত হয়?**

**উত্তর :** Register Indirect Addressing mode এ পয়েন্টার হিসেবে শুধুমাত্র রেজিস্টার R0, R1 ও DPTR ব্যবহৃত হয়।

**১৯। কোন অ্যাড্রেসিং মোডে @ চিহ্ন ব্যবহৃত হয়?**

**উত্তর :** Register indirect addressing mode এ @ চিহ্ন ব্যবহৃত হয়।

**২০। কোন অ্যাড্রেসিং মোডে # চিহ্ন ব্যবহৃত হয়?**

**উত্তর :** Immediate addressing mode এ # চিহ্ন ব্যবহৃত হয়।

**২১। Absolute addressing mode-এ কোন কোন নির্দেশনায় ব্যবহৃত হয়?**

**উত্তর :** ACALL ও AJUMP নির্দেশনাদ্বয়ে Absolute Addressing Mode ব্যবহৃত হয়।

**২২। রিলেটিভ অ্যাড্রিসিং মুড কীভাবে ব্যবহৃত হয়।**

**উত্তর :** রিলেটিভ অ্যাড্রেসিং মুড কিছু Jump Instruction যথা– SJMP, JNZ ইত্যাদি এর সাথে ব্যবহৃত হয়।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **8051 ইনস্ট্রাকশনের বর্ণনা দাও।** *[বাকাশিবো : '১৪]*

**উত্তর :** কতকগুলো বিটের অনুক্রম (Sequence) যা ডিজিটাল কম্পিউটারের কেন্দ্রীয় প্রক্রিয়াকরণ অংশকে কোন বিশেষ কার্য সম্পাদনের জন্য অবগত হওয়া হলো নির্দেশনা বা Instruction। এটি কার্যসম্পাদনে ব্যবহৃত ডাটাও ধারণ করতে পারে। এটি প্রোগ্রামের একটি ছোট অংশ (Segment) যা কম্পিউটার প্রসেসর কর্তৃক সম্পাদিত কাজের জন্য প্রয়োজনীয় ধাপ ধারণ করে। এটি একটি অপারেশন কোড (OP Code) যা সম্পাদিত কাজকে নির্দিষ্ট করে এবং এক বা একাধিক অপারেন্ড (operand) বা তাদের ঠিকানা (address) এবং এক বা একাধিক modifier বা তাদের অ্যাড্রেস নিয়ে গঠিত।

কোন প্রোগ্রাম কর্তৃক কোন কম্পিউটার প্রসেসরে প্রদেয় একটি আদেশ (Command) হলো নির্দেশনা। কোন প্রসেসর কর্তৃক বোধগম্য সমস্ত নির্দেশনা বা কমান্ডসমূহকে একত্রে ইনস্ট্রাকশন সেট বলে। 8051 মাইক্রোকন্ট্রোলারের নির্দেশনা সেটে মোট 111টি নির্দেশনা আছে।

২। **8051 মাইক্রোকন্ট্রোলার কী কী অ্যাড্রেসিং মুড সমর্থন করে নাম লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলার নিম্নবর্ণিত অ্যাড্রেসিং মুড সমর্থন করে।

(ক) Register addressing mode
(খ) Direct addressing mode
(গ) Register indirect addressing mode
(ঘ) Immediate addressing mode
(ঙ) Relative addressing mode
(চ) Absolute addressing mode
(ছ) Long addressing mode
(জ) Base resister plus index register addressing.

৩। **নির্দেশনার অপারেন্ড হিসেবে কী কী ব্যবহৃত হয়?**

**উত্তর :** নির্দেশনার অপারেন্ড হিসেবে নিম্নলিখিত রাশিগুলো ব্যবহৃত হয় :

(১) ডাটা নিজেই (Data valueitself)
(২) একটি সিপিইউ রেজিস্টার (CPU register)
(৩) একটি মেমোরি স্থান (Memory location)
(৪) একটি ইনপুট/আউটপুট পোর্ট (I/O port)।

৪। **8051 এর নির্দেশনায় শ্রেণিবিভাগ আলোচনা কর।**

**উত্তর :** নিচে 8051 এর নির্দেশনার শ্রেণিবিভাগ দেওয়া হলো :

(১) নির্দেশনার আকারের উপর ভিত্তি করে (On instruction size) :
  (i) One/single byte instruction.
  (ii) Two/double byte instruction.
  (iii) Three/triple byte instruction.

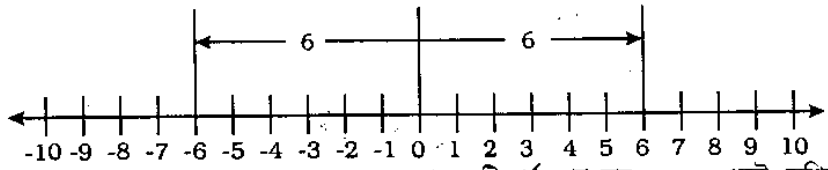
(২) কাজের বৈশিষ্ট্যের উপর ভিত্তি করে (on working characteristics) :
  (i) Arithmetic Instructions
  (ii) Branch Instructions
  (iii) Data Transfer Instructions
  (iv) Logic Instructions
  (v) Bit-oriented Instructions.

**৫। শর্ট রেঞ্জ, অ্যাবসলিউট ভ্যালু এবং লং রেঞ্জ আলোচনা কর।**

**উত্তর :** শর্ট রেঞ্জ, অ্যাবসলিউট ভ্যালু এবং লং রেঞ্জ আলোচনা নিম্নরূপ :

(1) **শর্টরেঞ্জ (Short range) :** এটি একটি পূর্ণসংখ্যা যা একই মেশিনে কোন একটি আদর্শ পূর্ণসংখ্যার তুলনায় ক্ষুদ্র range এর কারণে কম জায়গা (storage) দখল করে।

(2) **অ্যাবসলিউট ভ্যালু (Absolute value) :** কোন সংখ্যার পরম মান বা absolute value হলো শূন্য থেকে ঐ সংখ্যার দূরত্ব। নিচের গ্রাফে 0 থেকে 6 এর দূরত্ব সবসময় 6 এবং 0 থেকে – 6 এর দূরত্বও 6। সুতরাং 6 ও – 6 এর পরম মান 6। পরম মান সবসময় ধনাত্মক।




(3) **লংরেঞ্জ (Long range) :** Long range হলো একটি পূর্ণসংখ্যা যার range একই মেশিনে একটি আদর্শ পূর্ণসংখ্যার সমান বা বড়। এটি সাধারণত একটি আদর্শ পূর্ণসংখ্যার তুলনায় দ্বিগুণ জায়গা (storage) দখল করে।

## রচনামূলক প্রশ্নোত্তর

**১। 8051 মাইক্রোকন্ট্রোলারের অ্যাড্রেসিং মুডগুলো উদাহরণসহসংক্ষেপে বর্ণনা কর।** *[বাকাশিবো : '১৫]*

**উত্তর :** অনুচ্ছেদ ৪.৫ নং দ্রষ্টব্য।

**২। নির্দেশনা ও নির্দেশনা সেট কাকে বলে? 8051 এর নির্দেশনার শ্রেণিবিভাগ দেখাও।**

**উত্তর :** অনুচ্ছেদ ৪.১ ও ৪.২ নং দ্রষ্টব্য।

# অধ্যায় ৫ ইন্টেল 8051 মাইক্রোকন্ট্রোলারের অ্যাসেম্বলি ভাষা প্রোগ্রামিং

# Understand the Assembly Language Programming of the Intel 8051 Microcontroller

## ৫.০ ভূমিকা

## (Introduction)

অ্যাসেম্বলি ভাষা হলো কম্পিউটার বা অন্যান্য প্রোগ্রামযোগ্য ডিভাইসের কেন্দ্রীয় প্রক্রিয়াকরণ অংশের জন্য মানুষের বোধগম্য নির্দেশনা ব্যবহার করে রচিত একটি নিম্নস্তরের প্রোগ্রামিং ভাষা (Low level language)। এটি মেশিনের ভাষা (বাইনারিতে রচিত) থেকে এক ধাপ উপরে কিন্তু মানুষের অত্যন্ত সহজ বোধগম্য ভাষা, (High level language) এর নিচে।

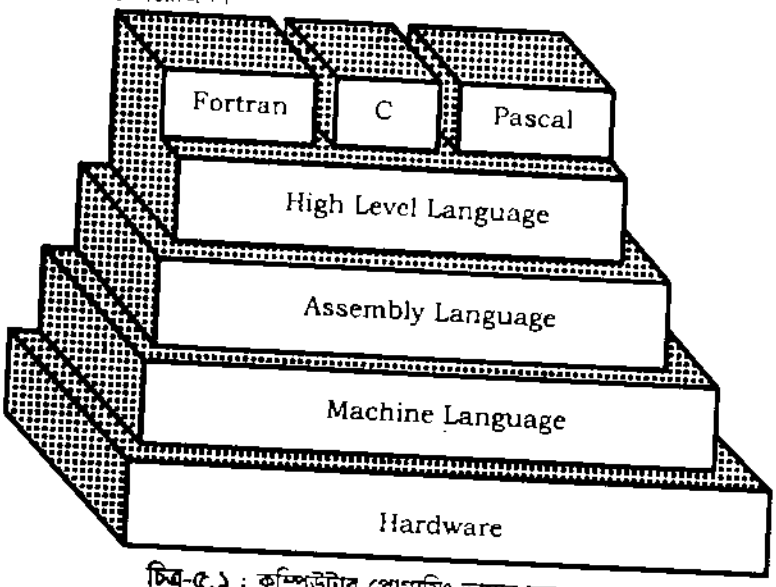
অ্যাসেম্বলার একটি প্রোগ্রাম যা অ্যাসেম্বলি ভাষায় রচিত প্রোগ্রামকে সম্পাদনযোগ্য মেশিনের ভাষায় রূপান্তর করে। রূপান্তরিত অ্যাসেম্বলি ফাইলকে Object file বলে। অবজেক্ট ফাইল বাইনারি ডিজিট 1 ও 0 নিয়ে গঠিত, যা মেশিন সরাসরি বুঝতে ও সম্পাদন করতে পারে। 8051 মাইক্রোকন্ট্রোলারের একটি জনপ্রিয় অ্যাসেম্বলার হলো ASM51।

## ৫.১ অ্যাসেম্বলি প্রোগ্রামিং সম্পর্কিত সংজ্ঞা

## (Define (i) Assembly Language (ii) Assembler and (iii) Linker (iv) IDE)

### (1) অ্যাসেম্বলি ভাষা (Assembly language) :

কম্পিউটার বা অন্যান্য প্রোগ্রামযোগ্য ডিভাইসের কেন্দ্রীয় প্রক্রিয়াকরণ অংশের জন্য মানুষের বোধগম্য নির্দেশনা ব্যবহার করে রচিত একটি নিম্নস্তরের প্রোগ্রামিং ভাষা (Low level language) হলো অ্যাসেম্বলি ভাষা। এটি মেশিনের ভাষা (বাইনারিতে রচিত) থেকে এক ধাপ উপরে, কিন্তু মানুষের অত্যন্ত সহজ বোধগম্য ভাষা (High level language)। অ্যাসেম্বলি ভাষার নির্দেশনাসমূহকে mnemonic বলে। এটি এমন একটি কম্পিউটার ভাষা, যার প্রকাশ (expression) কোন বিশেষ কম্পিউটারের মেশিনের ভাষার নির্দেশনার মানুষের বোধগম্য সাংকেতিক সমতুল্য। এটি এমন একটি Low level language, যা প্রোগ্রামকে মেশিন কোডের উপর সম্পূর্ণ নিয়ন্ত্রণের সুবিধা দেয়। প্রত্যেক অ্যাসেম্বলি ভাষা কোন বিশেষ কম্পিউটারের জন্য নির্দিষ্ট এবং এটিকে সম্পাদনযোগ্য মেশিনের ভাষায় রূপান্তরের জন্য অনুবাদক হিসেবে নির্দিষ্ট assembler প্রয়োজন।



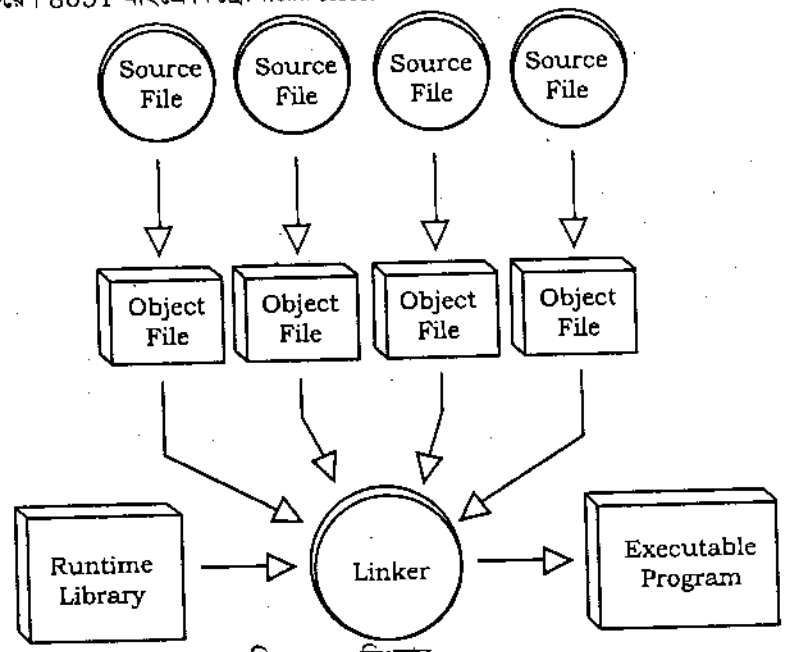

চিত্র-৫.১ : কম্পিউটার প্রোগ্রামিং ভাষার স্তর

**(2) অ্যাসেম্বলার (Assembler) :**

অ্যাসেম্বলি ভাষায় রচিত প্রোগ্রামকে সম্পাদনযোগ্য মেশিনের ভাষায় রূপান্তর করা প্রক্রিয়াকে অ্যাসেম্বলার প্রোগ্রাম বলে। রূপান্তরিত অ্যাসেম্বলি ফাইলকে Object file বলে। বাইনারি ডিজিট 1 ও 0 নিয়ে অবজেক্ট ফাইল গঠিত, যা মেশিন সরাসরি বুঝতে ও সম্পাদন করতে পারে। অ্যাসেম্বলার সফটওয়্যার ও অ্যাপ্লিকেশন উন্নয়নকারীকে কম্পিউটার হার্ডওয়্যার আর্কিটেকচার এবং এর উপাদানসমূহকে পরিচালনা বা ব্যবস্থাপনা এবং ফাইল থেকে তথ্য উদ্ধার কাজে সহায়তা করে। এটি অ্যাসেম্বলি ভাষা এবং কম্পিউটার প্রসেসর, মেমোরি ও অন্যান্য ডিভাইসের মাঝখানে সেতুবন্ধ হিসেবে কাজ করে। 8051 মাইক্রোকন্ট্রোলারের ASM51 একটি হলো জনপ্রিয় অ্যাসেম্বলার।

**(3) লিংকার (Linker) :**

লিংকার এমন একটি প্রক্রিয়া যে প্রোগ্রাম utility program এর মাধ্যমে কোন একটি বিশেষ পরিবেশে দুই বা ততোধিক অনুবাদিত মেশিন ফাইল বা অবজেক্ট ফাইলকে সংযুক্ত করে একটি standard library function-এ রূপান্তরপূর্বক সম্পাদনযোগ্য প্রোগ্রামে পরিণত করে। এর আউটপুট একটি load module, যা কম্পিউটারের সম্পাদনযোগ্য কোড। একে Linker editor-ও বলে, কারণ এটি প্রোগ্রাম মডিউল ও প্রয়োজনীয় সাব-রুটিন লাইব্রেরিসমূহকে একতাবদ্ধ করে নিচের চিত্র একটিমাত্র সম্পাদনযোগ্য ফাইলে রূপান্তর করে। 8051 মাইক্রোকন্ট্রোলারের Intel RL51 একটি জনপ্রিয় লিংকার হলো।




চিত্র-৫.২ : লিংকার

**(4) সমন্বিত উন্নয়ন পরিবেশ (Integrated Development Environment, IDE) :**

সমন্বিত উন্নয়ন প্রোগ্রামিং পরিবেশ এর সংক্ষিপ্তরূপ হলো IDE। যা একটি application program হিসেবে মোড়ককৃত (package)। এটি একটি source code editor, একটি compiler, একটি debugger এবং একটি graphical user interface (GUI) builder নিয়ে গঠিত। এটি প্রোগ্রামারকে সফটওয়্যার উন্নয়নে সমন্বিত সুবিধা প্রদান করে। আধুনিক অধিকাংশ IDE- তে Intelligent code completion বৈশিষ্ট্য থাকে। এটি স্বতন্ত্র একটি প্রোগ্রাম বা অন্য কোন প্রোগ্রামের অংশ হতে পারে। যেমন– BASIC programming language মাইক্রোসফট অফিসে ব্যবহৃত হতে পারে। IDE ব্যবহারকারীদেরকে অনেক আধুনিক প্রোগ্রামিং ভাষার (যেমন : Visual Basic, Java ও Power Builder) জন্য একটি সহজ কাঠামো সংস্থান করে।

## ৫.২ অ্যাসেম্বলি ভাষার নির্দেশনার ক্ষেত্রসমূহ (Mention the Fields of an Assembly Language Instruction)

নিম্নবর্ণিত ক্ষেত্রসমূহ নিয়ে একটি অ্যাসেম্বলি ভাষার নির্দেশনা গঠিত :

[Label] [Opcode] [Operand] [Comments]

নিচে ফিল্ডগুলোর সংক্ষিপ্ত বর্ণনা দেওয়া হলো :

**Label (ঐচ্ছিক) :** প্রোগ্রামের কোন নির্দেশনার লাইনকে নাম দ্বারা প্রকাশ করার জন্য এটা ব্যবহৃত হয়। এটা অনুসরণকারী নির্দেশনার ঠিকানাও প্রকাশ করে। যখন অনুসরণকারী নির্দেশনাকে শাখায়িত করা (Branching) হয়, তখন এই লেবেল Branch বা Jump Instruction-এ অপারেন্ড হিসেবে ব্যবহৃত হয়। লেবেল একটি কোলন চিহ্ন (:) দ্বারা সমাপ্ত হয়। লেবেল অবশ্যই অক্ষর, প্রশ্নবোধক চিহ্ন ("?") বা Underscore ("_") দ্বারা শুরু হয় এবং অক্ষর, সংখ্যা, "?" ও "_" চিহ্ন দিয়ে সর্বোচ্চ 31 অক্ষরের মধ্যে শেষ করতে হয়। Instruction mnemonic ও Assembler directive সমূহকে সংরক্ষিত শব্দ বিধায় লেবেল হিসেবে ব্যবহার করা যায় না।

**Opcode (আবশ্যিক) :** অ্যাসেম্বলি ভাষার Mnemonic বা assembler directive সমূহ Operation code (opcode) উৎপন্ন করে। এটি নির্দেশনার আবশ্যিক অংশ। এটি সিপিইউকে কী করতে হবে তা প্রকাশ করে।

**Operand (ঐচ্ছিক) :** যে তথ্যের উপর নির্দেশনা আরোপিত হয় তা হলো Operand। অপারেন্ড হিসেবে নিচের রাশিগুলো ব্যবহৃত হতে পারে :

(১) ডাটার মান (Data value)
(২) একটি সিপিইউ রেজিস্টার (CPU Register)
(৩) একটি মেমোরি লোকেশন (Memory location)
(৪) একটি I/O port।

অপারেন্ডের সম্ভাব্যতা সম্পূর্ণ নির্ভর করে অপারেশনের উপর। কোন কোন নির্দেশনায় এক বা একাধিক অপারেন্ড থাকতে পারে, আবার কোনটিতে অপারেন্ড থাকেই না (যেমন : RET)। যদি একাধিক অপারেন্ড থাকে, তবে এর গঠন নিম্নরূপ :

**Instructi- : Destination, Source**

**Comments (ঐচ্ছিক) :** এটি নির্দেশনার মন্তব্য। এটি নির্দেশনা লাইনের শেষে বা একটি স্বতন্ত্র লাইনেও হতে পারে। মন্তব্য লাইন সেমিকোলন (;) দিয়ে শুরু হয়। অ্যাসেম্বলার এটিকে উপেক্ষা করে। সাবরুটিন ও প্রোগ্রামের কোন বড় অংশ একটি Comment block (কয়েক লাইনের মন্তব্য) দ্বারা শুরু হয়।

## ৫.৩ সিপিইউ নির্দেশনা ও অ্যাসেম্বলার ডিরেকটিভ বর্ণনা (State the Meaning of CPU Instruction and Assembler Directives)

### (ক) সিপিইউ নির্দেশনা (CPU instruction) :

সিপিইউ নির্দেশনা বলতে কোন কম্পিউটার প্রোগ্রাম কর্তৃক কম্পিউটার প্রসেসরে প্রদেয় আদেশকে বুঝায়। নিম্নস্তরে প্রতিটি নির্দেশনা 0 ও 1 এর একটি ধারা, যা কম্পিউটারের করণীয় একটি ভৌত কাজের বর্ণনা করে। অ্যাসেম্বলি ভাষায় প্রত্যেক বিবরণ (statement) সাধারণত একটি সিপিইউ নির্দেশনার সঙ্গতিপূর্ণ, কিন্তু উচ্চস্তরের ভাষায় একটি বিবরণ সাধারণত বহু সিপিইউ নির্দেশনার সাথে সঙ্গতিপূর্ণ।

### (খ) অ্যাসেম্বলার ডিরেকটিভ (Assembler Directive) :

অ্যাসেম্বলার প্রোগ্রামের প্রতি কতকগুলো বিশেষ নির্দেশনা, যা কোন বিশেষ কাজের জন্য অ্যাসেম্বলারকে নির্দেশ প্রদান করে তখন তাকে অ্যাসেম্বলার ডিরেকটিভ বলে। এরা কোন প্রতীকের মান ও চলক জমা রাখার ক্ষেত্রকে (Storage area) সংরক্ষণ ও সূচনা করার জন্য ব্যবহৃত হয়। কোন বিশেষ অ্যাসেম্বলারের জন্য এরা নির্দিষ্ট। মাইক্রোকন্ট্রোলারের কাজে এদের কোন প্রভাব নেই। কোন কোন ডিরেকটিভ প্রত্যেক প্রোগ্রামের জন্য বাধ্যতামূলক কিন্তু কোন কোনটি শুধুমাত্র মাইক্রোকন্ট্রোলারের গতি সহজতর করার জন্য ব্যবহৃত হয়। দুয়েকটি ব্যতিক্রম ছাড়া (DB, DW) এরা মেমোরির ধারকে (content) সরাসরি কোন প্রভাব বিস্তার করে না।

## ৫.৪ সচরাচর ব্যবহৃত অ্যাসেম্বলার ডিরেকটিভ এর তালিকা

**(List the Commonly Used Assembler Directives)**

8051 মাইক্রোকন্ট্রোলারের জনপ্রিয় অ্যাসেম্বলার হলো ASM51। শ্রেণিবিভাগসহ এর কিছু ডিরেকটিভ নিচে উল্লেখ করা হলো :

(i) Assembler state control : ORG, END, USING

(ii) Symbol definition : SEGMENT, EQU, SET, DATA, IDATA, XDATA, BIT, CODE

(iii) Storage initialization/reservation : DS, DBIT, DB, DW

(iv) Program linkage : PUBLIC, EXTRN, NAME

(v) Segment selection : RSEG, CSEG, DSEG, ISEG, BSEG, XSEG

## ৫.৫ এবং ৫.৬ অ্যাসেম্বলি প্রোগ্রাম তৈরি ও সম্পাদনের ধাপ উল্লেখ করণ

**(Mention the Steps to Create and Execute Assembly Language Program)**

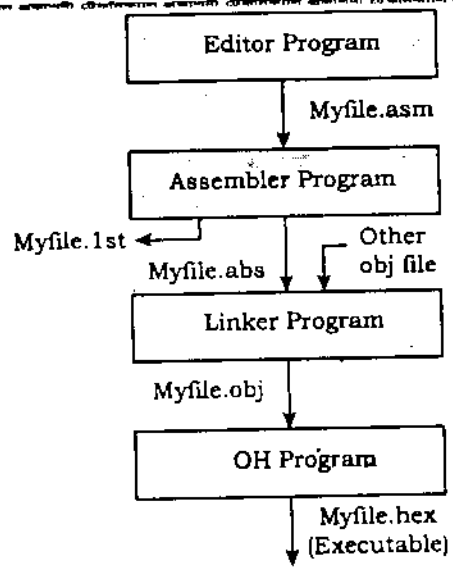
একটি সম্পাদনযোগ্য অ্যাসেম্বলি প্রোগ্রাম রচনার ধাপসমূহ নিচে উল্লেখ করা হলো :

১। প্রথম পর্যায় প্রোগ্রাম লিখার জন্য একটি Editor ব্যবহার করতে হবে। বর্তমানে প্রোগ্রাম লিখন ও সংশোধনের জন্য অনেক Editor বা Word processor আছে। তবে উল্লেখ থাকে যে, উক্ত Editor এর ASCII ফাইল উৎপাদনে সমর্থ থাকতে হবে। Editor এ প্রোগ্রাম লিখিত এই ফাইলকে Source file বলে। অধিকাংশ অ্যাসেম্বলারের ক্ষেত্রে Source file এর নাম DOS এর রীতি অনুসারে হয়ে থাকে, কিন্তু অ্যাসেম্বলি প্রোগ্রামের উপর ভিত্তি করে ফাইলের বর্ধিত নাম "asm" বা "src" হয়।

২। প্রোগ্রাম কোড সমৃদ্ধ এই "asm" source file কে 8051 অ্যাসেম্বলারে প্রয়োগ করা হয়। অ্যাসেম্বলার প্রোগ্রামের নির্দেশনাসমূহকে মেশিন কোডে রূপান্তর করে এবং একটি object file ও একটি list file উৎপন্ন করে। এই ফাইলের বর্ধিত নাম যথাক্রমে "obj" ও "1st"। List file সমস্ত Opcode, ঠিকানা, এমনকি অ্যাসেম্বলার কর্তৃক নিরূপিত ত্রুটির তালিকা তৈরি করে। এটি ঐচ্ছিক হলেও প্রোগ্রামারের জন্য এটি খুবই জরুরি; কারণ প্রোগ্রামার নির্দেশনার syntax error ও debugging করার জন্য এটিকে ব্যবহার করতে হয়।

৩। সম্পাদনযোগ্য অ্যাসেম্বলি ফাইল তৈরির ৩য় ধাপ হলো Linking. লিংকার প্রোগ্রাম এক বা একাধিক অবজেক্ট ফাইল নিয়ে একটি Absolute object file উৎপন্ন করে। এই ফাইলের বর্ধিত নাম হলো "abs"। এই Absolute ফাইল মনিটর প্রোগ্রাম সমৃদ্ধ 8051 ট্রেইনারে ব্যবহৃত হয়।

৪। Absolute ফাইলকে পরবর্তীতে 8051 অ্যাসেম্বলারের সাথে আসা একটি object to hex converte প্রোগ্রামে প্রয়োগ করা হয়। এটি Absolute ফাইলকে মাইক্রোকন্ট্রোলারের সম্পাদনযোগ্য hex ফাইলে রূপান্তর করে। এই ফাইলের বর্ধিত নাম "hex"। পরে এটিকে মাইক্রোকন্ট্রোলারের Program ROM জমা করা (Burn) হয়। আধুনিক Windows based assembler 2নং থেকে 4নং ধাপকে একটি ধাপে সংক্ষিপ্ত করা হয়েছে। নিচে চিত্রে Assembly language program এর বিভিন্ন ফাইলগুলোকে চিহ্নিত করে একটি সম্পাদনযোগ্য ফাইল রচনার ধাপগুলো দেখানো হলো।




চিত্র-৫.৩ : একটি সম্পাদনযোগ্য Assembly language program রচনার ধাপ ও বিভিন্ন ফাইল

## ৫.৭ 8051 মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর ধাপে ধাপে এর কাজের ব্যাখ্যা (Explain the Step-by-step Action of the 8051 Upon Applying Power on It)

8051 পরিবারের সদস্যরা তাদেরকে মেমোরি অ্যাড্রেস 0000H থেকে শুরু করে যখন বিদ্যুৎশক্তি প্রয়োগ করা হয়। Program counter (PC) এর মান হয় 0000। যেহেতু Booting এর সময় 8051 প্রথম নির্দেশনাটি খুঁজে, তাই ROM এ প্রোগ্রাম জমা করার (Burn) সময় প্রথম Opcodeটি মেমোরির 0000H ঠিকানায় জমা করা হয়। এই কাজটি source program এ ORG নির্দেশনা দ্বারা করা যায়। 8051 মাইক্রোকন্ট্রোলারে শক্তি প্রয়োগের পর ধাপে ধাপে এর কাজের বর্ণনার জন্য নিচে একটি অ্যাসেম্বলি ভাষা প্রোগ্রাম উল্লেখ করা হলো :

```
        ORG     OH          ;   start (origin) at 0
        MOV     R5, #25H    ;   load 25H into R5
        MOV     R7, #35     ;   load 34H into R7
        MOV     A, #0       ;   load 0 into A
        ADD     A, R5       ;   add contents or R7 to A
        ADD     A, R7       ;   add contents of R7 to A
        ADD     A, #12H     ;   add to value 12H
HERE :  SJMP    HERE        ;   stay in this loop
        END                 ;   end of asm source file
```

এই প্রোগ্রাম অ্যাসেম্বলার কর্তৃক অনুবাদের পর এর List file নিম্নরূপ :

| ROM Address | Machine Language | Assembly Language |
|---|---|---|
| 0000 | 7D25 | MOV R5, # 25H |
| 0002 | 7F34 | MOV R7, #34H |
| 0004 | 7400 | MOV A, #0 |
| 0006 | 2D | ADD A, R5 |
| 0007 | 2F | ADD A, R7 |
| 0008 | 2412 | ADD A, #12H |
| 000A | 80EF | HERE : SJMP HERE |

ROM এ প্রোগ্রাম জমা করার (burn) পর Opcode ও operand অবস্থান মেমোরির 0000 স্থান থেকে শুরু হয়। নিচে ROM এর আধেয় দেখানো হলো :

| Address | Code |
|---|---|
| 0000 | 7D |
| 0001 | 25 |
| 0002 | 7F |
| 0003 | 34 |
| 0004 | 74 |
| 0005 | 00 |
| 0006 | 2D |
| 0007 | 2F |
| 0008 | 24 |
| 0009 | 12 |
| 000A | 80 |
| 000B | FE |

মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর ধাপে ধাপে এর কাজ উল্লেখ করা হলো :

১। 8051 মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর এটি PC 0000 তে Reset হয় এবং প্রোগ্রাম ROM এর 0000 স্থান থেকে প্রথম নির্দেশনা আনয়ন (fetch) হয়। Opcode 7D সম্পাদনের পর সিপিইউ 25 কে আনয়ন করে এবং এটিকে R5 রেজিস্টারে জমা করে। এখন একটি নির্দেশনা শেষ হলো এবং PC বৃদ্ধি পেয়ে 0002 তে সেট হলো, যে ঠিকানায় opcode, 7F আছে।

২। Opcode 7F সম্পাদনের পর সিপিইউ 34 কে আনয়ন করে এবং R7 রেজিস্টারে জমা করে। PC আবার 0004 এ বর্ধিত হয়।

৩। 0004 মেমোরি স্থানের নির্দেশনা সম্পাদিত হয় এবং PC এর মান 0006 এ উন্নীত হয়।

৪। মেমোরির 0006 স্থানের 1 বাইট নির্দেশনা সমাপ্তনান্তে PC = 0008। সমস্ত নির্দেশনা আনয়ন ও সম্পাদন শেষ না হওয়া পর্যন্ত এই প্রক্রিয়া চলতে থাকে।

৫। 8051 পরিবারের কোন সদস্য 64 কিলোবাইটের বেশি Opcode access করতে পারে না।

## ৫.৮ একটি সরল অ্যাসেম্বলি ভাষা প্রোগ্রাম

### (Write Simple Assembly Language Program for the 8051)

নিচে 25H ও 34H দুটি সংখ্যার যোগ করে ফলাফল অ্যাকুমুলেটরে জমা রাখার একটি সরল অ্যাসেম্বলি প্রোগ্রাম উল্লেখ করা হলো :

```
MOV     A,  #25H      ; load 25H into A
MOV     R2, #34H      ; load 34H into R2
ADD     A, R2         ; add R2 to Accumulator
                      ; (A = A + R2)
```

**নির্দেশনার ব্যাখ্যা :**

**১ম নির্দেশনা :** 25H কে অ্যাকুমুলেটরে (A) জমা করা হবে।

**২য় নির্দেশনা :** 34H কে R2 রেজিস্টারে জমা করা হবে।

**৩য় নির্দেশনা :** অ্যাকুমুলেটর (A) এর তথ্য (25H) এর সাথে R2 রেজিস্টারের তথ্য (34H) এর যোগ হয়ে যোগফল অ্যাকুমুলেটরে জমা হবে।

# অনুশীলনী-০৫

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **IDE কী?** *[বাকাশিবো : '১৪]*

**উত্তর :** IDE একটি সমন্বিত প্রোগ্রামিং পরিবেশ, যা একটি application program হিসেবে মোড়ককৃত (package)। এটি একটি source code editor, একটি compiler, একটি debugger এবং একটি graphical user interface (GUI) builder নিয়ে গঠিত। এটি প্রোগ্রামারকে সফটওয়্যার উন্নয়নে সমন্বিত সুবিধা প্রদান করে।

২। **অ্যাসেম্বলি প্রোগ্রামে list file এর কাজ কী?** *[বাকাশিবো : '১৪R]*

**উত্তর :** অ্যাসেম্বলি প্রোগ্রামে list file-কে প্রোগ্রামারগণ নির্দেশনার syntax error ও debugging করার জন্য ব্যবহার করে।

৩। **অ্যাসেম্বলি ভাষা কী?**

**উত্তর :** অ্যাসেম্বলি ভাষা হলো কম্পিউটার বা অন্যান্য প্রোগ্রামযোগ্য ডিভাইসের কেন্দ্রীয় প্রক্রিয়াকরণ অংশের জন্য মানুষের বোধগম্য নির্দেশনা ব্যবহার করে রচিত একটি নিম্নস্তরের প্রোগ্রামিং ভাষা (Low level language)। এটি মেশিনের ভাষা (বাইনারিতে রচিত) থেকে এক ধাপ উপরে কিন্তু মানুষের অত্যন্ত সহজ বোধগম্য ভাষা, (High level language) এর নিচে।

৪। **অ্যাসেম্বলার কাকে বলে?**

**উত্তর :** অ্যাসেম্বলার একটি প্রোগ্রাম যা অ্যাসেম্বলি ভাষায় রচিত প্রোগ্রামকে সম্পাদনযোগ্য মেশিনের ভাষায় রূপান্তর করে। রূপান্তরিত অ্যাসেম্বলি ফাইলকে Object file বলে। অবজেক্ট ফাইল বাইনারি ডিজিট 1 ও 0 নিয়ে গঠিত, যা মেশিন সরাসরি বুঝতে ও সম্পাদন করতে পারে। 8051 মাইক্রোকন্ট্রোলারের একটি জনপ্রিয় অ্যাসেম্বলার হলো ASM51।

৫। **অবজেক্ট ফাইল কাকে বলে?**

**উত্তর :** অ্যাসেম্বলার কর্তৃক রূপান্তরিত অ্যাসেম্বলি ফাইলকে Object file বলে। অবজেক্ট ফাইল বাইনারি ডিজিট 1 ও 0 নিয়ে গঠিত, যা মেশিন সরাসরি বুঝতে ও সম্পাদন করতে পারে।

৬। **লিংকার কাকে বলে? উদাহরণসহ লিখ।**

**উত্তর :** যে utility program কোন একটি বিশেষ পরিবেশে দুই বা ততোধিক অনুবাদিত মেশিন ফাইল বা অবজেক্ট ফাইলকে সংযুক্ত করে একটি standard library function এ রূপান্তরপূর্বক সম্পাদনযোগ্য প্রোগ্রামে পরিণত করে, তাকে লিংকার বলে। এর আউটপুট একটি load module, যা কম্পিউটারের সম্পাদনযোগ্য কোড। এটিকে Linker editor-ও বলে, কারণ এটি প্রোগ্রাম মডিউল ও প্রয়োজনীয় সাবরুটিন লাইব্রেরিসমূহকে একতাবদ্ধ করে একটিমাত্র সম্পাদনযোগ্য ফাইলে রূপান্তর করে। 8051 মাইক্রোকন্ট্রোলারের একটি জনপ্রিয় লিংকার প্রোগ্রাম Intel RL51.

৭। **অ্যাসেম্বলার ডিরেকটিভ কাকে বলে?**

**উত্তর :** অ্যাসেম্বলার ডিরেকটিভ হলো অ্যাসেম্বলার প্রোগ্রামের প্রতি কতকগুলো বিশেষ নির্দেশনা, যা কোন বিশেষ কাজের জন্য অ্যাসেম্বলারকে নির্দেশ প্রদান করে। এরা কোন প্রতীকের মান ও চলক জমা রাখার ক্ষেত্রকে (Storage area) সংরক্ষণ ও সূচনা করার জন্য ব্যবহৃত হয়। কোন বিশেষ অ্যাসেম্বলারের জন্য এটি নির্দিষ্ট।

**৮। বহুল ব্যবহৃত কয়েকটি অ্যাসেম্বলার ডিরেকটিভের নাম লিখ।**

**উত্তর :** শ্রেণিবিভাগসহ 8051 এর ডিরেকটিভসমূহ :

(ক) Assembler state control : ORG, END, USING

(খ) Symbol definition : SEGMENT, EQU, SET, DATA, IDATA, XDATA, BIT, CODE

(গ) Storage initialization/reservation : DS, DBIT, DB, DW

(ঘ) Program linkage : PUBLIC, EXTRN, NAME

(ঙ) Segment selection : RSEG, CSEG, DSEG, ISEG, BSEG, XSEG

**৯। 8051 মাইক্রোকন্ট্রোলারে অ্যাসেম্বলি প্রোগ্রামের absolute file কোথায় ব্যবহৃত হয়?**

**উত্তর :** Absolute ফাইল মনিটর প্রোগ্রাম সমৃদ্ধ 8051 ট্রেইনারে ব্যবহৃত হয়।

**১০। একটি অ্যাসেম্বলি ভাষার নির্দেশনা বা ভাবে গঠিত হয়?**

**উত্তর :** নিম্নবর্ণিত ক্ষেত্রসমূহ নিয়ে একটি অ্যাসেম্বলি ভাষার নির্দেশনা গঠিত :

[Label] [Opcode] [Operand] [Comments]

**১১। প্রথম পর্যায় প্রোগ্রাম লেখার জন্য কী ব্যবহার করা হতো?**

**উত্তর :** প্রথম পর্যায় প্রোগ্রাম লিখার জন্য একটি Editor ব্যবহার করতে হবে।

**১২। IDE এর পূর্ণরূপ লেখ।**

**উত্তর :** IDE এর পূর্নরূপ হলো : Integrated Development Enuironment

## সংক্ষিপ্ত প্রশ্নোত্তর

**১। CPU Instruction কাকে বলে?** *[বাকাশিবো : '১৪]*

**উত্তর :** কোন কম্পিউটার প্রোগ্রাম কর্তৃক কম্পিউটার প্রসেসরে প্রদেয় আদেশকে সিপিইউ নির্দেশনা বলে। নিম্নস্তরে প্রতিটি নির্দেশনা 0 ও 1 এর একটি ধারা, যা কম্পিউটারের করণীয় একটি ভৌত কাজের বর্ণনা করে। অ্যাসেম্বলি ভাষায় প্রত্যেক বিবরণ (statement) সাধারণত একটি সিপিইউ নির্দেশনার সঙ্গতিপূর্ণ, কিন্তু উচ্চস্তরের ভাষায় একটি বিবরণ সাধারণত বহু সিপিইউ নির্দেশনার সাথে সঙ্গতিপূর্ণ।

**২। অ্যাসেম্বলি ভাষা প্রোগ্রামের ফাইলগুলোর নাম লিখ।**

**উত্তর :** অ্যাসেম্বলি ভাষা প্রোগ্রামের নিম্নলিখিত ফাইলগুলো দেখা যায়–

১। Source File (File Extention-. asm বা src)

২। List File (File Extension-. 1st)

৩। Object File (File Extension-. obj)

৪। Absolute File (File Extension-. abs)

৫। Hex File (File Extension-. oh)।

**৩। অ্যাসেম্বলি ভাষার অপারেন্ড হিসেবে কী কী ব্যবহৃত হয়?**

**উত্তর :** অ্যাসেম্বলি ভাষার অপারেন্ড হিসেবে নিচের রাশিগুলো ব্যবহৃত হতে পারে :

(ক) ডাটার মান (Data value)

(খ) একটি সিপিউ রেজিস্টার (CPU Register)

(গ) একটি মেমোরি লোকেশন (Memory location)

(ঘ) একটি I/O port।

**৪। অ্যাসেম্বলি ভাষার ক্ষেত্রসমূহ বর্ণনা কর।**

**উত্তর :** নিম্নবর্ণিত ক্ষেত্রসমূহ নিয়ে একটি অ্যাসেম্বলি ভাষার নির্দেশনা গঠিত :

[Label] [Opcode] [Operand] [Comments]

নিচে ফিল্ডগুলোর সংক্ষিপ্ত বর্ণনা দেওয়া হলো :

**Label (ঐচ্ছিক) :** প্রোগ্রামের কোন নির্দেশনার লাইনকে নাম দ্বারা প্রকাশ করার জন্য এটা ব্যবহৃত হয়। এটা অনুসরণকারী নির্দেশনার ঠিকানাও প্রকাশ করে। যখন অনুসরণকারী নির্দেশনাকে শাখায়িত করা (Branching) হয়, তখন এই লেবেল Branch বা Jump Instruction-এ অপারেন্ড হিসেবে ব্যবহৃত হয়। লেবেল একটি কোলন চিহ্ন (:) দ্বারা সমাপ্ত হয়। লেবেল অবশ্যই অক্ষর, প্রশ্নবোধক চিহ্ন ("?") বা Underscore ("_") দ্বারা শুরু হয় এবং অক্ষর, সংখ্যা, "?" ও "_" চিহ্ন দিয়ে সর্বোচ্চ 31 অক্ষরের মধ্যে শেষ করতে হয়। Instruction mnemonic ও Assembler directive সমূহকে সংরক্ষিত শব্দ বিধায় লেবেল হিসেবে ব্যবহার করা যায় না।

**Opcode (আবশ্যিক) :** অ্যাসেম্বলি ভাষার Mnemonic বা assembler directive সমূহ Operation code (opcode) উৎপন্ন করে। এটি নির্দেশনার আবশ্যিক অংশ। এটি সিপিইউকে কী করতে হবে তা প্রকাশ করে।

**Operand (ঐচ্ছিক) :** যে তথ্যের উপর নির্দেশনা আরোপিত হয় তা হলো Operand। অপারেন্ড হিসেবে নিচের রাশিগুলো ব্যবহৃত হতে পারে :

(১) ডাটার মান (Data value)

(২) একটি সিপিইউ রেজিস্টার (CPU Register)

(৩) একটি মেমোরি লোকেশন (Memory location)

(৪) একটি I/O port।

অপারেন্ডের সম্ভাব্যতা সম্পূর্ণ নির্ভর করে অপারেশনের উপর। কোন কোন নির্দেশনায় এক বা একাধিক অপারেন্ড থাকতে পারে, আবার কোনটিতে অপারেন্ড থাকেই না (যেমন : RET)। যদি একাধিক অপারেন্ড থাকে, তবে এর গঠন নিম্নরূপ :

**Instruction Destination, Source**

**Comments (ঐচ্ছিক) :** এটি নির্দেশনার মন্তব্য। এটি নির্দেশনা লাইনের শেষে বা একটি স্বতন্ত্র লাইনেও হতে পারে। মন্তব্য লাইন সেমিকোলন (;) দিয়ে শুরু হয়। অ্যাসেম্বলার এটিকে উপেক্ষা করে। সাবরুটিন ও প্রোগ্রামের কোন বড় অংশ একটি Comment block (কয়েক লাইনের মন্তব্য) দ্বারা শুরু হয়।

**৫। একটি সরল অ্যাসেম্বলি ভাষার প্রোগ্রাম লিখ।**

**উত্তর :** নিচে 25H ও 34H দুটি সংখ্যার যোগ করে ফলাফল অ্যাকুমুলেটরে জমা রাখার একটি সরল অ্যাসেম্বলি প্রোগ্রাম উল্লেখ করা হলো :

```
MOV     A,  #25H      ; load 25H into A
MOV     R2, #34H      ; load 34H into R2
ADD     A, R2         ; add R2 to Accumulator
                      ; (A = A + R2)
```

**নির্দেশনার ব্যাখ্যা :**

**১ম নির্দেশনা :** 25H কে অ্যাকুমুলেটরে (A) জমা করা হবে।

**২য় নির্দেশনা :** 34H কে R2 রেজিস্টারে জমা করা হবে।

**৩য় নির্দেশনা :** অ্যাকুমুলেটর (A) এর তথ্য (25H) এর সাথে R2 রেজিস্টারের তথ্য (34H) এর যোগ হয়ে যোগফল অ্যাকুমুলেটরে জমা হবে।

৬। **অ্যাসিম্বলি ভাষা সম্পর্কে কার্যব্যাখ্যা কর।**

**উত্তর :** কম্পিউটার বা অন্যান্য প্রোগ্রামযোগ্য ডিভাইসের কেন্দ্রীয় প্রক্রিয়াকরণ অংশের জন্য মানুষের বোধগম্য নির্দেশনা ব্যবহার করে রচিত একটি নিম্নস্তরের প্রোগ্রামিং ভাষা (Low level language) হলো অ্যাসেম্বলি ভাষা। এটি মেশিনের ভাষা (বাইনারিতে রচিত) থেকে এক ধাপ উপরে, কিন্তু মানুষের অত্যন্ত সহজ বোধগম্য ভাষা (High level language)। অ্যাসেম্বলি ভাষার নির্দেশনাসমূহকে mnemonic বলে। এটি এমন একটি কম্পিউটার ভাষা, যার প্রকাশ (expression) কোন বিশেষ কম্পিউটারের মেশিনের ভাষার নির্দেশনার মানুষের বোধগম্য সাংকেতিক সমতুল্য। এটি এমন একটি Low level language, যা প্রোগ্রামকে মেশিন কোডের উপর সম্পূর্ণ নিয়ন্ত্রণের সুবিধা দেয়। প্রত্যেক অ্যাসেম্বলি ভাষা কোন বিশেষ কম্পিউটারের জন্য নির্দিষ্ট এবং এটিকে সম্পাদনযোগ্য মেশিনের ভাষায় রূপান্তরের জন্য অনুবাদক হিসেবে নির্দিষ্ট assembler প্রয়োজন।

## রচনামূলক প্রশ্নোত্তর

১। **উদাহরণসহকারে 8051 মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর ধাপে ধাপে এর কাজের বর্ণনা দাও।** *[বাকাশিবো : '১৫]*

**উত্তর :** অনুচ্ছেদ ৫.৬ নং অনুচ্ছেদ দ্রষ্টব্য।

২। **একটি সম্পাদনযোগ্য অ্যাসেম্বলি ভাষার প্রোগ্রাম রচনার ধাপগুলো বর্ণনা কর।** *[বাকাশিবো : '১৪R]*

**অথবা, অ্যাসেম্বলি ভাষার প্রোগ্রাম তৈরি ও সম্পাদনের ধাপ (Steps) লেখ।** *[বাকাশিবো : '১৪]*

**উত্তর :** অনুচ্ছেদ ৫.৫ নং অনুচ্ছেদ দ্রষ্টব্য।

৩। **অ্যাসেম্বলি ভাষার নির্দেশনার ক্ষেত্রসমূহ বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৫.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৪। **অ্যাসেম্বলি প্রোগ্রামিং সম্পর্কিত সংজ্ঞা বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৫.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৫। **সিপিইউ নির্দেশনা ও অ্যাসেম্বলার ডিরেকটিভ বর্ণনা**

**উত্তর :** অনুচ্ছেদ ৫.৩ নং অনুচ্ছেদ দ্রষ্টব্য।

# অধ্যায় ৬

# সাবরুটিন

# Understand the Subroutine

## ৬.০ ভূমিকা

## (Introduction)

প্রয়োজনীয় Function কিংবা Macro definition সম্বলিত একটি External file কে প্রোগ্রামের অংশ হিসেবে অন্তর্ভুক্ত করা যায়। এর ফলে একাধিক হেডারে রক্ষিত ঐ function বা macro definition সমূহকে নতুন করে লেখার প্রয়োজন হয় না। হেডার সংযোগের এই বিকল্প ধারাটি বেশি প্রাধান্য পায়। বিশেষ করে যখন প্রোগ্রামের বৈশিষ্ট্য ও চাহিদা মাফিক হেডারের সংখ্যা বৃদ্ধি পায়। সেক্ষেত্রে এক সঙ্গে অনেকগুলো হেডারের পরিবর্তে কেবলমাত্র একটি হেডার দিয়েই প্রোগ্রামের সঠিকতা ও বৈধতা রক্ষা করা যায়। পাশাপাশি অনেকগুলো হেডার লেখার বাড়তি কষ্ট ও অনেকটা কমে যায়।

## ৬.১ সাবরুটিন

## (Define Subroutine)

সাবরুটিন হলো কম্পিউটার প্রোগ্রামিং এ একটি বৃহৎ প্রোগ্রাম মডিউলের একটি ছোট অংশ যার নিজস্ব অভ্যন্তরীণ কার্যাবলি সার্বিক প্রোগ্রামের প্রয়োজন মেটানোর জন্য ডিজাইন করা হয়। এটি প্রোগ্রামের নির্দেশনাসমূহের একটি অনুক্রম যা একক ইউনিট হিসেবে একটি বিশেষ কাজ সম্পাদন করে। বিশেষ কাজের প্রয়োজন অনুসারে এই ইউনিট প্রোগ্রামের যে-কোন স্থানে সম্পন্ন হয়। এটি প্রোগ্রামে অথবা লাইব্রেরিসমূহে পৃথকভাবে সংজ্ঞায়িত হতে পারে। প্রসেডিউর, ফাংশন, রুটিন, মেথড বা সাবপ্রোগ্রাম ইত্যাদি বিভিন্ন প্রোগ্রামিং ভাষায় এটিকে বিভিন্নভাবে নামকরণ করা হয়।

## ৬.২ সাবরুটিনের সুবিধা উল্লেখ করণ

## (Mention the Advantages of Subroutine)

প্রোগ্রামের সাবরুটিনের সুবিধা নিচে দেওয়া হলো :

১। একাধিক প্রোগ্রামে এটিকে পুনঃব্যবহার করা যায়।

২। এটি কোন প্রোগ্রামে একই রকম কোড বার বার লিখার ঝামেলা কমায়। পরন্তু একই কোড বার বার লিখার পরিবর্তে শুধু একবার লিখে প্রোগ্রাম রচনা করা হয়, ফলে প্রোগ্রাম ছোট হয় এবং প্রোগ্রামের মেমরি ক্ষেত্র কম লাগে যা ছোট Embedded system এর জন্য খুবই উপযোগী।

৩। সাবরুটিনের সাহায্যে একটি জটিল প্রোগ্রামকে ছোট ছোট সরল ধাপে বিভক্ত করা যায়। এতে প্রোগ্রামের গঠন উন্নত হয়।

৪। ছোট ছোট অংশের (সাবরুটিনের) পরীক্ষা (Test) ও ত্রুটি সংশোধনের মাধ্যমে প্রোগ্রামের সার্বিক দক্ষতা বৃদ্ধি পায়।

৫। একটি বৃহৎ প্রোগ্রামকে ছোট ছোট মডিউলে বিভক্ত করে বিভিন্ন প্রোগ্রামারকে ভাগ করে দেওয়া যায়। ফলে প্রোগ্রাম উন্নয়ন সময় কম লাগে, প্রোগ্রামে ভুলত্রুটি কম হয় এবং ত্রুটি সংশোধন (Debugging) সহজ হয়।

## ৬.৩ সাবরুটিন আহ্বান ও প্রত্যাবর্তন বর্ণনা

## (Describe Calling and Returning from Subroutine)

**সাবরুটিন আহ্বান (Subroutine calling) :**

সাধারণত মূল প্রোগ্রামের সাবরুটিন আহ্বানের জন্য বা মূল প্রোগ্রাম থেকে প্রত্যাবর্তনের জন্য অ্যাসেম্বলি ভাষায় নির্দেশনা থাকে। 8051 মাইক্রোকন্ট্রোলারের সাবরুটিন আহ্বানের জন্য দুটি নির্দেশনা হলো : (১) ACALL ও (২) LCALL।

নিচে এদের কার্যক্রমে প্রোগ্রামের প্রবাহের নিয়ন্ত্রণ করা হলো :

**LCALL (Long CALL)**

কোন সাবরুটিনকে একটি নির্দিষ্ট ঠিকানায় এই নির্দেশনা আহ্বান করার জন্য ব্যবহৃত হয়। ঠিকানাটি 16 বিট দৈর্ঘ্যের হয় যাতে 64 কিলোবাইটের মেমরি ক্ষেত্রের যে-কোন স্থানে আহ্বান করা যায়। যখন LCALL নির্দেশনা সম্পাদিত হয় তখন বর্তমান PC এর ধারক স্বয়ংক্রিয়ভাবে PC এর Stack এর উপর PUSH হয়। যখন প্রোগ্রাম সাবরুটিন থেকে প্রত্যাবর্তন করে তখন PC এর ধারক Stack থেকে প্রত্যাবর্তন করে (POP)। ফলে LCALL যেখান থেকে প্রোগ্রাম সম্পাদিত হচ্ছিল প্রোগ্রাম ঠিক সেখানে থেকেই পুনরায় অপারেশন শুরু করে। সাবরুটিন থেকে প্রত্যাবর্তনের জন্য RET নির্দেশনা ব্যবহৃত হয়।

**ACALL (Absolute CALL)**

ACALL নির্দেশনা যৌক্তিকভাবে LCALL এর সদৃশ। কিন্তু এর ঠিকানার সীমা AJMP নির্দেশনার মতো সীমিত। CALL হলো অনেক 8051 অ্যাসেম্বলারের সমর্থিত একটি নির্দেশনা। কোন নির্দেশনা (LCALL বা ACALL) ব্যবহার হবে তা অ্যাসেম্বলার নির্ধারণ করে।

**সাবরুটিন প্রত্যাবর্তন (Subroutine returning) :**

সাবরুটিন একটি বিশেষ নির্দেশনা (RET) দ্বারা শেষ হয় যা প্রোগ্রাম সম্পাদনাকে CALL এর পরবর্তী নির্দেশনায় প্রত্যাবর্তন করে। এটি Stack থেকে শেষের দুই বাইটকে POP করে PC তে জমা করে।

RETI কোন Interrupt Service Routine (ISR) থেকে প্রত্যাবর্তন করার জন্য ব্যবহৃত হয়। RET ও RETI এর একমাত্র পার্থক্য হলো চলমান ইন্টারাপ্ট শেষ হলে RETI ইন্টারাপ্ট কন্ট্রোল সিস্টেমকে সিগন্যাল প্রদান করে। RETI সম্পাদনকালে কোন ইন্টারাপ্টের মুলতবি না হলে RETI ও RET এর কোন পার্থক্য থাকে না।

## ৬.৪ ডিলে উৎপন্ন করার জন্য একটি সরল সাবরুটিন প্রোগ্রাম

## (Write Subroutine for creating delay)

8051 মাইক্রোকন্ট্রোলারে প্রসেসরের ক্লক পালসের প্রতি12 ক্লক সাইকেলে একটি একক নির্দেশনা সম্পন্ন হয়। এতে 12 মে. হার্টস ক্লক গতিতে চালিত 8051 মাইক্রোকন্ট্রোলারের Instruction cycle time হলো এক মাইক্রোসেকেন্ড।

$$\text{Instruction cycle time} = \frac{12 \text{ clock cycles}}{12 \times 10^6 \text{ cycles/sec.}} = 10^{-6} \text{ seconds or } 1\mu\text{sec.}$$

ক্ষুদ্রতম নির্দেশনাসমূহ এক Instruction cycle এ সম্পাদিত হয় এবং অন্যান্য নির্দেশনাসমূহ দুই বা ততোধিক cycle time এ সম্পাদিত হয়। প্রতি নির্দেশনার cycle time 8051 মাইক্রোকন্ট্রোলার নির্মাতা কর্তৃক প্রকাশিত থাকে।

মনে করি 'ONE_MILLI_SUB' নামের একটি সাবরুটিন প্রোগ্রাম তৈরি করা হবে। যদি এতে 1000টি Instruction cycle সম্পাদন হয় তবে এর মোট সম্পাদন কাল হবে 1000 মাইক্রোসেকেন্ড বা 1

মিলিসেকেন্ড। এই সাবরুটিনে 4 Instruction cycles এর একটি Tight loop তৈরি করে এটিকে 250 বার সম্পাদন করা হলে প্রোগ্রামটি সম্পাদন হতে মোট সময় লাগবে 1000 মাইক্রোসেকেন্ড বা 1 মিলিসেকেন্ড। নিচে সাবরুটিন প্রোগ্রাম দেখানো হলো :

**Tight loop টি নিম্নরূপ :**

```
NOP                          ;  1 Instruction cycle
NOP                          ;  1 Instruction cycle
DJNZ R7, LOOP_1_MILLI        ;  2 Instruction cycles
```

এই ছোট লুপ প্রোগ্রামটি সম্পন্ন হতে মোট 4 Instruction cycle বা 4 microsecond লাগে। এটিকে ব্যবহার করে নিচে 1 মিলিসেকেন্ড ডিলে (delay) করে একটি সাবরুটিনের কোড দেখানো হলো। এই প্রোগ্রামে LOOP counter হিসেবে R7 রেজিস্টার ব্যবহার করা হয়েছে। কীভাবে এই সাবরুটিনকে মেইন প্রোগ্রামে call করা যায় তাও দেখানো হলো :

```
ONE-MILLI-SUB :
    PUSH 07h                 ;  save R7 to stack
    MOV R7, #250d            ;  250 decimal to R7 to count loops

    LOO-1-MILLI :            ;  Loops 250 times
    NOP                      ;  inserted NOPs to cause delay
    NOP                      ;
    DJNZ R7, LOOP-1-MILLI    ;  decrement R7, if not zero loop back
    POP 07h                  ;  restore R7 to original value
    RET                      ;  return from subroutine (to caller)
: How to call subroutine "ONE-MILLI-SUB"
    ORG 0h                   ;  program address starting at 0h
MAIN :
    LCALL ONE-MILLI-SUB      ;  ONE-MILLI-SUB subroutine call
    END                      ;  end of asm file
```

# অনুশীলনী-০৬

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **সাবরুটিন কাকে বলে?** *[বাকাশিবো : '১৪, '১৫]*

**উত্তর :** কম্পিউটার প্রোগ্রামিং এ সাবরুটিন হলো একটি বৃহৎ প্রোগ্রাম মডিউলের একটি ছোট অংশ যার নিজস্ব অভ্যন্তরীণ কার্যাবলি সার্বিক প্রোগ্রামের প্রয়োজন মেটানোর জন্য ডিজাইন করা হয়। এটি প্রোগ্রামের নির্দেশনাসমূহের একটি অনুক্রম যা একক ইউনিট হিসেবে একটি বিশেষ কাজ সম্পাদন করে।

২। **ACALL এর কাজ লেখ।** *[বাকাশিবো : '১৫]*

**উত্তর :** ACALL এর কাজ হলো একটি অ্যাসলিউট সাবরুটিনকে নির্দিষ্ট ঠিকনায় CALL করা। তবে এর সীমা সীমিত।

৩। **STMP এর কাজ কী?** *[বাকাশিবো : '১৫]*

**উত্তর :** STMP এর কাজ হলো নির্দেশনা সাপেক্ষে Short Jump দেওয়া।

৪। **8051-এ স্ট্যাক পয়েন্টার রেজিস্টার কীভাবে ব্যবহৃত হয়?** *[বাকাশিবো : '১৫]*

**উত্তর :** 8051-এ স্ট্যাক পয়েন্টার রেজিস্টার প্রোগ্রাম অ্যাকটিভেকিউট করার সময় অস্থায়ী মেমোরি হিসেবে ব্যবহৃত হয়।

৫। **8051 মাইক্রোকন্ট্রোলারে কী কী CALL নির্দেশনা আছে?** *[বাকাশিবো : '১৪]*

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারে LCALL ও ACALL নামে দুটি CALL নির্দেশনা আছে।

৬। **বিভিন্ন প্রোগ্রামিং ভাষায় সাবরুটিনকে কী কী নামে অভিহিত করা হয়?**

**উত্তর :** বিভিন্ন প্রোগ্রামিং ভাষায় সাবরুটিনকে প্রসিডিউর, ফাংশন, রুটিন, মেথড বা সাবপ্রোগ্রাম ইত্যাদি নামে অভিহিত করা হয়।

৭। **Call নির্দেশনার কাজ কী?**

**উত্তর :** সাবরুটিনকে মেইন প্রোগ্রামে আহ্বান করা।

৮। **সাবরুটিন সম্পর্কিত নির্দেশনাগুলোর নাম লিখ।**

**উত্তর :** সাবরুটিন সম্পর্কিত নির্দেশনা : LCALL, ACALL, RET, RETI

৯। **LCALL ও ACALL এর পার্থক্য লিখ।**

**উত্তর :** ACALL নির্দেশনা যৌক্তিকভাবে LCALL এর সদৃশ। কিন্তু এর ঠিকানার সীমা AJMP নির্দেশনার মতো সীমিত।

১০। **RET নির্দেশনার কাজ কী?**

**উত্তর :** সাবরুটিন একটি বিশেষ নির্দেশনা (RET) দ্বারা সমাপ্ত হয় যা প্রোগ্রাম সম্পাদনাকে CALL এর পরবর্তী নির্দেশনায় প্রত্যাবর্তন করে। এটি Stack থেকে শেষের দুই বাইটকে POP করে PC তে জমা করে।

১১। **LCALL এর কাজ লেখ।**

**উত্তর :** কোন সাবরুটিনকে একটি নির্দিষ্ট ঠিকানায় এই নির্দেশনা আহ্বান করার জন্য ব্যবহৃত হয়। ঠিকানাটি 16 বিট দৈর্ঘ্যের হয় যাতে 64 কিলোবাইটের মেমোরি ক্ষেত্রের যে-কোন স্থানে আহ্বান করা যায়।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **সাবরুটিনের সুবিধাগুলো লিখ।** *[বাকাশিবো : '১৪, '১৪R, '১৫]*

**উত্তর :** প্রোগ্রামের সাবরুটিনের সুবিধা নিচে দেওয়া হলো :

(ক) একাধিক প্রোগ্রামে এটিকে পুনঃব্যবহার করা যায়।

(খ) এটি কোন প্রোগ্রামে একই রকম কোড বার বার লিখার ঝামেলা কমায়। পরন্তু একই কোড বার বার লিখার পরিবর্তে শুধু একবার লিখে প্রোগ্রাম রচনা করা হয়, ফলে প্রোগ্রাম ছোট হয় এবং প্রোগ্রামের মেমরি ক্ষেত্র কম লাগে যা ছোট Embedded system এর জন্য খুবই উপযোগী।

(গ) সাবরুটিনের সাহায্যে একটি জটিল প্রোগ্রামকে ছোট ছোট সরল ধাপে বিভক্ত করা যায়। এতে প্রোগ্রামের গঠন উন্নত হয়।

(ঘ) ছোট ছোট অংশের (সাবরুটিনের) পরীক্ষা (Test) ও ত্রুটি সংশোধনের মাধ্যমে প্রোগ্রামের সার্বিক দক্ষতা বৃদ্ধি পায়।

(ঙ) একটি বৃহৎ প্রোগ্রামকে ছোট ছোট মডিউলে বিভক্ত করে বিভিন্ন প্রোগ্রামারকে ভাগ করে দেওয়া যায়। ফলে প্রোগ্রাম উন্নয়ন সময় কম লাগে, প্রোগ্রামে ভুলত্রুটি কম হয় এবং ত্রুটি সংশোধন (Debugging) সহজ হয়।

২। **12 মে. হার্টস ক্লক পাল চালিত 8051 মাইক্রোকন্ট্রোলারে একটি ক্ষুদ্রতম নির্দেশনা সম্পন্ন হতে কত সময় লাগে?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের প্রসেসরের ক্লক পালসের প্রতি 12 ক্লক সাইকেলে একটি একক নির্দেশনা সম্পন্ন হয়। এতে 12 মে. হার্টস ক্লক গতিতে চালিত 8051 মাইক্রোকন্ট্রোলারের Instruction cycle time হলো এক মাইক্রোসেকেন্ড।

$$\text{Instruction cycle time} = \frac{12 \text{ clock cycles}}{12 \times 10^6 \text{ cycles/sec.}}$$
$$= 10^{-6} \text{ seconds or } 1\mu\text{sec.}$$

ক্ষুদ্রতম নির্দেশনাসমূহ এক Instruction cycle এ সম্পাদিত হয় এবং অন্যান্য নির্দেশনাসমূহ দুই বা ততোধিক cycle time এ সম্পাদিত হয়। প্রতি নির্দেশনার cycle time 8051 মাইক্রোকন্ট্রোলার নির্মাতা কর্তৃক প্রকাশিত থাকে।

৩। **সাবরুটিন আহ্বান আলোচনা কর।**

সাধারণত মূল প্রোগ্রামের সাবরুটিন আহ্বানের জন্য বা মূল প্রোগ্রাম থেকে প্রত্যাবর্তনের জন্য অ্যাসেম্বলি ভাষায় নির্দেশনা থাকে। 8051 মাইক্রোকন্ট্রোলারের সাবরুটিন আহ্বানের জন্য দুটি নির্দেশনা হলো : (১) ACALL ও (২) LCALL।

৪। **LCALL আলোচনা কর।**

**উত্তর :** কোন সাবরুটিনকে একটি নির্দিষ্ট ঠিকানায় এই নির্দেশনা আহ্বান করার জন্য ব্যবহৃত হয়। ঠিকানাটি 16 বিট দৈর্ঘ্যের হয় যাতে 64 কিলোবাইটের মেমেরি ক্ষেত্রের যে-কোন স্থানে আহ্বান করা যায়। যখন LCALL নির্দেশনা সম্পাদিত হয় তখন বর্তমান PC এর ধারক স্বয়ংক্রিয়ভাবে PC এর Stack এর উপর PUSH হয়। যখন প্রোগ্রাম সাবরুটিন থেকে প্রত্যাবর্তন করে তখন PC এর ধারক Stack থেকে প্রত্যাবর্তন করে (POP)। ফলে LCALL যেখান থেকে প্রোগ্রাম সম্পাদিত হচ্ছিল প্রোগ্রাম ঠিক সেখানে থেকেই পুনরায় অপারেশন শুরু করে। সাবরুটিন থেকে প্রত্যাবর্তনের জন্য RET নির্দেশনা ব্যবহৃত হয়।

৫। **ACALL আলোচনা কর।**

**উত্তর :** ACALL নির্দেশনা যৌক্তিকভাবে LCALL এর সদৃশ। কিন্তু এর ঠিকানার সীমা AJMP নির্দেশনার মতো সীমিত। CALL হলো অনেক 8051 অ্যাসেম্বলারের সমর্থিত একটি নির্দেশনা। কোন নির্দেশনা (LCALL বা ACALL) ব্যবহার হবে তা অ্যাসেম্বলার নির্ধারণ করে।

৬। **সাবরুটিন প্রত্যাবর্তন আলোচনা কর।**

**উত্তর :** সাবরুটিন একটি বিশেষ নির্দেশনা (RET) দ্বারা সমাপ্ত হয় যা প্রোগ্রাম সম্পাদনাকে CALL এর পরবর্তী নির্দেশনায় প্রত্যাবর্তন করে। এটি Stack থেকে শেষের দুই বাইটকে POP করে PC তে জমা করে।

RETI কোন Interrupt Service Routine (ISR) থেকে প্রত্যাবর্তন করার জন্য ব্যবহৃত হয়। RET ও RETI এর একমাত্র পার্থক্য হলো চলমান ইন্টারাপ্ট শেষ হলে RETI ইন্টারাপ্ট কন্ট্রোল সিস্টেমকে সিগন্যাল প্রদান করে। RETI সম্পাদনকালে কোন ইন্টারাপ্টের মুলতবি না হলে RETI ও RET এর কোন পার্থক্য থাকে না।

## রচনামূলক প্রশ্নোত্তর

১। **উদাহরণসহ সাবরুটিন আহ্বান ও প্রত্যাবর্তন প্রক্রিয়া সংক্ষেপে বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৬.৩ নং দ্রষ্টব্য।

২। **ডিলে উৎপন্ন করার জন্য সরল সাবরুটিন প্রোগ্রাম বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ৬.৪ নং দ্রষ্টব্য।

# অধ্যায় ৭

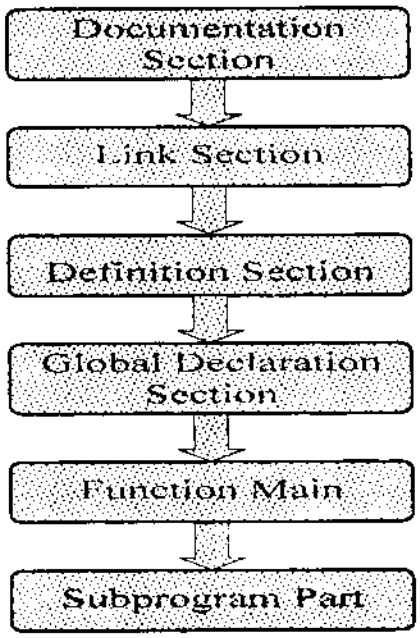
# সি ভাষায় 8051 মাইক্রোকন্ট্রোলার প্রোগ্রামিং

# Understand Programming 8051 in C

## ৭.০ ভূমিকা

**(Introduction) :**

সি-প্রোগ্রামের মৌলিক গঠন বা Basic Structure of C Program বলতে সি ল্যাংগুয়েজ ব্যবহার করে একটি স্ট্যান্ডার্ড প্রোগ্রাম তৈরি করতে হলে একজন প্রোগ্রামারকে প্রোগ্রামটি কোন গাঠনিক নিয়ম মেনে তৈরি করতে হবে তাকে বুঝানো হয়েছে। নিম্নে সি-প্রোগ্রামের মৌলিক গঠন ব্লক ডায়াগ্রামের মাধ্যমে উপস্থাপন করা হলো যার ছয়টি মৌলিক ধাপ রয়েছে :




চিত্র-৭.১ : সি-প্রোগ্রামের মৌলিক গঠন

**Documentation Section :** উপরের ব্লক ডায়াগ্রামে ডকুমেন্টেশন সেকশন বলতে যেকোনো প্রোগ্রামের শুরুতে অথবা এর বিভিন্ন অংশে প্রোগ্রামের উদ্দেশ্য, ডাটার বর্ণনা বা প্রোগ্রামের অংশ বিশেষের বর্ণনা comments (মন্তব্য) আকারে লেখা থাকে। সি-প্রোগ্রামে comments /* চিহ্ন দিয়ে লেখা শুরু হয় এবং */ চিহ্ন দিয়ে শেষ করতে হয়।

**Link Section :** Link Section এ প্রোগ্রামে ব্যবহৃত ফাংশনসমূহকে সি-প্রোগ্রামের হেডার ফাইলসমূহের সাথে সম্পৃক্ত করা হয়। এ কাজে সি-ল্যাংগুয়েজের প্রিপ্রোসেসর ডিরেকটিভস্ #include<> ব্যবহার করতে হয়, যার সিনটেক্স হচ্ছে #include < Header_file_name >

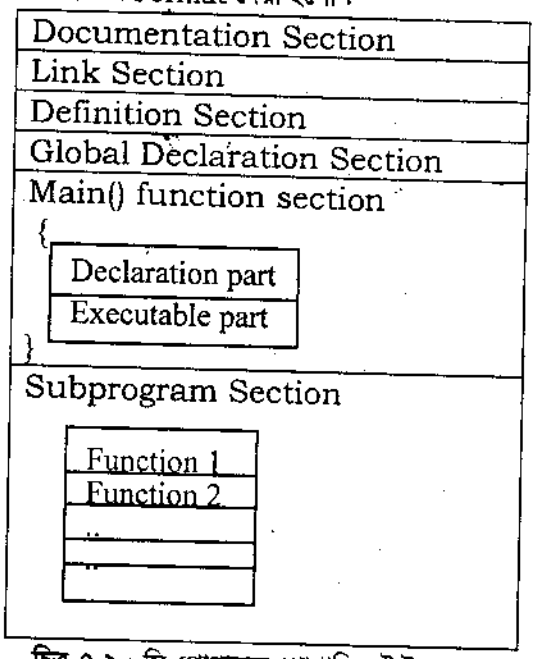
**Definition Section :** Definition Section-এ #define ডিরেকটিভস্ দ্বারা সিম্বলিক কনস্টেন্টকে সজ্ঞায়িত করা হয়। যেমন : #define PI 3.14 এখানে PI কনস্টেন্ট-এর মান 3.14 দ্বারা নির্ধারিত করা হয়েছে

**Global Declaration Section :** Global Declaration Section-এ গ্লোবাল ভেরিয়েবলের মান ঘোষণা করা হয় যে ভেরিয়েবলগুলোকে Main Function সহ অন্যান্য সকল Sub_function হতে ব্যবহার করা যায়।

**Function Main :** Function Main-এ মূল প্রোগ্রামটি লেখা হয়। এর দুটি অংশ থাকে। যথা : Local Declaration Part ও Executable Part. লোকাল ডিকলারেশন অংশে লোকাল ভেরিয়েবলের মান ঘোষণা করা হয় যে ভেরিয়েবলসমূহকে শুধুমাত্র ঐ ফাংশনই ব্যবহার করতে পারবে। আর executable বা নির্বাহ অংশে যে সমস্যা সমাধানের প্রোগ্রাম তৈরি হবে তার নির্দেশাবলি বা কোডিং লেখা হয়।

**Subprogram Section :** Subprogram Section Subprogram Part-এ Main Function-এর অধীনে একাধিক ইউজার ডিফাইন্ড ফাংশন বা প্রোগ্রামার কর্তৃক তৈরিকৃত ফাংশন থাকে, যে ফাংশনগুলোকে মেইন ফাংশন হতে প্রয়োজনে একাধিকবার ব্যবহার করা যায়।

নিম্নে সি প্রোগ্রামের প্রোগ্রামিং স্টাইল বা Format দেয়া হলো :




চিত্র-৭.২ : সি-প্রোগ্রামের প্রোগ্রামিং স্টাইল

**প্রোগ্রামিং স্টাইল (Programming Style) :** প্রোগ্রামিং স্টেটমেন্টসমূহকে যুক্তিগ্রাহ্যভাবে লেখার রীতিকে প্রোগ্রামিং স্টাইল (Programming Style) বলে। C-কে Free form ল্যাংগুয়েজ বলা হয়। এর অর্থ হচ্ছে কোথায় লাইন টাইপ শুরু হলো– কম্পাইলার তা গ্রাহ্য করে না। তবে যত্রতত্র প্রোগ্রাম স্টেটমেন্ট লেখার অভ্যাস পরিহার করা উচিত। তাতে বরং প্রোগ্রামের পাঠযোগ্যতা বাড়ে।

প্রোগ্রাম লেখার অনেক বিকল্প রীতি রয়েছে। তবে প্রোগ্রামের সাথে সংঙ্গতিপূর্ণ একটি স্টাইল বেছে নিলে ভালো হয়। প্রোগ্রামিং স্টাইল বার বার পরিবর্তন না করে একটির মধ্যে সীমাবদ্ধ থাকলে ভিন্ন মাত্রার পরিবেশে প্রোগ্রাম পরিচালনায় বেশি স্বাচ্ছন্দ অনুভব করা যায়।

C প্রোগ্রামিং এ Program লিখার জন্য যদিও একাধিক Style প্রচলিত তথাপি যে কোন একটি Style ব্যবহার করে প্রোগ্রাম লিখা খুবই গুরুত্বপূর্ণ বিষয়।

নিম্নে C Programming Style এর একটি নমুনা দেয়া হলো :

```
# include <stdio.h>
void main ()
{
clrscr ();
printf ("Dhaka Polytechnic Institute");
}
```

সি তে প্রোগ্রাম লিখার ক্ষেত্রে একাধিক Style থাকলেও প্রোগ্রাম কোডিং এর নিম্নবর্ণিত বিষয়সমূহ বিবেচ্য। যথা :

১. প্রতিটি সি প্রোগ্রামই ছোট হাতের অক্ষরে লিখতে হয় ব্যতিক্রম শুধু Symbolic Constant এর বেলায়।

২. সি প্রোগ্রামের বিশেষ কিছু Statement ছাড়া প্রতিটি Statement Semicolon দ্বারা শেষ হয়। সুতরাং Statement এর শেষে Semicolon ব্যবহার করে একই লাইনে একাধিক Statement লিখা যায়।

৩. Braces '{ }' বা দ্বিতীয় বন্ধনী দ্বারা ফাংশনের শুরু ও শেষ বুঝায়। একাধিক প্রোগ্রাম লাইনকে একত্রে সমন্বিত করার উদ্দেশ্যও Brace ব্যবহার করা হয়। যথোপযুক্ত স্থানে এই Brace ব্যবহার করলে প্রোগ্রামটি দেখতে আকর্ষণীয় হয়, রিডিং ও ডিবাগিং প্রক্রিয়া সহজতর হয়। অন্যথায় প্রোগ্রামে ত্রুটি (Error) পরিলক্ষিত হয়। তাই যথাযথভাবে Brace ব্যবহার করতে হবে।

৪. সামান্য কিছু ব্যতিক্রম ছাড়া কম্পাইলার ফাঁকা লাইন এবং ফাঁকা স্থানসহ প্রোগ্রাম কোড গণনা করে।

## ৭.১ সি ভাষায় প্রোগ্রাম লিখার কারণ উল্লেখ করণ

## (Mention the Reasons for writing program in C)

কোন বিশেষ ডিভাইসের সফটওয়্যার উন্নয়ন শুধুমাত্র ঐ ডিভাইসের বিশেষ অ্যাসেম্বলি ভাষা দিয়ে হয়। এই অ্যাসেম্বলি ভাষাতে সাংখ্যিক মেশিন ভাষা (যেমন– 0 × 12 0 × 07 0 × A4 0 × 8F) এর পরিবর্তে অক্ষরভিত্তিক mnemonic (যেমন : MOV 22 MYBUFFER + 7) ব্যবহৃত হয়। একটি অ্যাসেম্বলার অ্যাসেম্বলি ভাষাকে মেশিনের ভাষায় রূপান্তর করে। কিন্তু অ্যাসেম্বলি ভাষা কোন বিশেষ মাইক্রোকন্ট্রোলারের জন্য নির্দিষ্ট। ফলে মাইক্রোকন্ট্রোলার পরিবার পরিবর্তন করলে এর জন্য বিশেষ অ্যাসেম্বলি ভাষাও জানতে হয়। পক্ষান্তরে, সি (C) সাধারণ উদ্দেশ্যে ব্যবহৃত একটি উচ্চস্তরের ভাষা। কম্পাইলার থাকলে এটা যে কোন মাইক্রোকন্ট্রোলারে ব্যবহৃত হতে পারে। তবে এর প্রধান অসুবিধা হলো– প্রোগ্রাম বড় হয় এবং ফলে বড় প্রোগ্রাম মেমোরির প্রয়োজন হয়। তাই ছোট কোড মেমোরির মাইক্রোকন্ট্রোলারের ক্ষেত্রে এটি অনুপযোগী। নিচে মাইক্রোকন্ট্রোলারে সি প্রোগ্রাম ব্যবহারের সুবিধা উল্লেখ করা হলো :

১। ফাংশন লাইব্রেরির কোড ব্যবহার করা যায়।

২। সি প্রোগ্রাম সহজ বোধগম্য এবং একে সংরক্ষণও সহজ।

৩। অ্যাসেম্বলি ভাষার তুলনায় সি ভাষায় প্রোগ্রাম রচনা সহজ এবং সময় অনেক কম লাগে।

৪। একে পরিবর্তন ও হালনাগাদ করা সহজ।

৫। সব ধরনের মাইক্রোকন্ট্রোলারে প্রয়োগ করা যায়। ফলে ভিন্ন ভিন্ন মাইক্রোকন্ট্রোলারের জন্য ভিন্ন ভিন্ন এসেম্বলি ভাষা জানা প্রয়োজন নেই।

৬। এটা সুবহ (portable) অর্থাৎ একে সামান্য পরিবর্তন করে বা কোনরূপ পরিবর্তন না করেই সহজে অন্য মাইক্রোকন্ট্রোলারে ব্যবহার করা যায়।

## ৭.২ 8051 এর জন্য সি ডাটার ধরন ও অপারেটরের তালিকা

## (List C Data Types and Operators for 8051)

**8051 এর জন্য সি ডাটার ধরন (C data types for 8051) :**

8051 এর জন্য প্রোগ্রামিং "সি" এর ডাটার ধরনের উপর ভাল ধারণা প্রোগ্রামারকে ছোট hex file তৈরি করতে সাহায্য করে। নিচে সি প্রোগ্রামে বহুল ব্যবহৃত ডাটা শ্রেণির আকার ও রেঞ্জসহ তালিকা দেওয়া হলো :

| ডাটার শ্রেণি | আকার (bit) | ডাটা রেঞ্জ/ব্যবহার |
|---|---|---|
| Unsigned char | 8-bit | 0 – 255 |
| (Signed) char | 8-bit | (– 128) – (+ 127) |
| Unsigned int | 16-bit | 0 – 65535 |
| (signed) int | 16-bit | (– 32768) – (+ 32767) |
| Sbit (single bit) | 1-bit | শুধুমাত্র SFR bit-addressable |
| Bit | 1-bit | RAM bit-addressable |
| Sfr | 8-bit | শুধুমাত্র RAM addressable 80–FFH |

**8051 এর জন্য সি অপারেটর (C operator for 8051) :**

প্রোগ্রামিং "সি" এর রাশিমালায় ব্যবহৃত অপারেটরের তালিকা নিচে উল্লেখ করা হলো :

**(ক) Data access and size operator**

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| [ ] | Array element | x[6] | অ্যারে x এর ৭ম উপাদান |
| . | Member selection | portb.2 | পোর্ট D এর বিট 2 |
| -> | Member selection | pstruct->x | pstruct দ্বারা structure এর মেম্বার নির্ধারিত হয়। |
| * | Indirection | *p | অ্যাড্রেস p এ মেমোরি স্থানের ধারক। |
| & | Address of | &x | চলক x এর অ্যাড্রেস। |

**(খ) Relational and logical operator**

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| > | Greater than | x > y | যদি x এর মান y এর চেয়ে বড় হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| >= | Greater than or equal to | x > = y | যদি x এর মান y এর চেয়ে বড় বা সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| < | Less than | x < y | যদি x এর মান y এর চেয়ে ছোট হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| < = | Less than or equal | x < = y | যদি x এর মান y এর চেয়ে ছোট হয় বা সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| = | Equal to | x = y | যদি x এর মান y এর সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| != | Not equal to | x != y | যদি x এর মান y মান সমান না হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| ! | Logical NOT | !x | যদি x এর মান 0 হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| && | Logical AND | x&&y | যদি x বা y এর মান 0 হয়, তবে লজিক 0, অন্যথায় 1 হয়। |
| \|\| | Logical OR | x\|\|y | যদি x ও y এর মান উভয়ই 0 হয়, তবে লজিক 0, অন্যথায় 1 হয়। |

**(গ) Arithmetic operator**

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| * | Multiplication | x*y | x কে y দ্বারা গুণ করে। |
| / | Division | x/y | x কে y দ্বারা ভাগ করে। |
| % | Modulo | x%y | x কে y দ্বারা ভাগ করার পর ভাগশেষ নির্ণয় করে। |
| + | Addition | x + y | x এর সাথে y এর যোগ করে। |
| – | Subtraction | x – y | x থেকে y এর বিয়োগ করে। |
| ++ | Increment | x ++ | এটি ব্যবহারের পর x এর 1 বৃদ্ধি করে। |
| – – | Decrement | – – x | এটি ব্যবহারের পূর্বে x এর মান 1 হ্রাস করে। |
| + | Unary Plus | + x | x-কে ধনাত্মক প্রদর্শন করে। |
| – | Negation | – x | x কে – 1 দ্বারা গুণ করে। |

৩৮৭১- মাই: ২য়: সংখ্যা: – ৮ (খ)

**(ঘ) Bitwise operator**

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| ~ | Bitwise complement NOT | ~ x | বিট 1 কে 0 তে এবং বিট 0 কে 1 এ রূপান্তর করে। |
| & | Bitwise AND | x&y | x ও y এর Bitwise AND |
| \| | Bitwise OR | x\|y | x ও y এর Bitwise OR |
| ^ | Bitwise Exclusive OR | x ^ y | x ও y এর XOR |
| << | Left shift | x<<2 | x এর বিট বাম দিকে 2 বিট পজিশনে স্থানান্তরিত হয়। |
| >> | Rigth shift | x >> 2 | x এর বিট ডান দিকে 2 বিট পজিশনে স্থানান্তরিত হয়। |

**(ঙ) Miscellaneous operator**

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| ( ) | Function | wait (10) | 10 এর একটি আর্গুমেন্টের সাথে অপেক্ষা। |
| (type) | Type cast | (double)x | x একটি double-এ রূপান্তরিত হয়। |
| ?: | Conditional | x?y:z | যদি x শূন্য না হয়, তবে y মূল্যায়ন করে, অন্যথায় z মূল্যায়ন করে। |
| , | Sequential evaluation | x ++, y++ | প্রথমে x এর ক্রমবৃদ্ধি হয় এবং পরে y এর ক্রমবৃদ্ধি হয়। |

## ৭.৩ 8051 এর জন্য সি ভাষায় টাইম ডিলে তৈরির প্রক্রিয়া বর্ণনা (Describe Creating Time Delay in C)

8051 মাইক্রোকন্ট্রোলারের জন্য সি ভাষায় দুই ভাবে টাইম ডিলে করা যায় যথা :

১। একটি সরল লুপ ব্যবহার করে।

২। 8051 টাইমার ব্যবহার করে।

উভয়ক্ষেত্রেই ডিলে একটি অসিলোস্কোপে বা একটি সিমুলেটর ব্যবহার করে দেখা যায়। এখানে একটি সরল লুপ ব্যবহার করে কীভাবে টাইম ডিলে তৈরি করা হয় তা আলোচনা করা হলো। লুপ ব্যবহার করে টাইম ডিলে তৈরির জন্য তিনটি ফেক্টর বিবেচনা করতে হয়, যা ডিলে এর সঠিকতায় প্রভাব বিস্তার করতে পারে :

১। মেশিন সাইকেলের সংখ্যা এবং প্রতি মেশিন সাইকেলে ক্লকের সংখ্যা।

২। 8051 এর সিস্টেম ফ্রিকুয়েন্সি। এটি XTAL1 ও XTAL2 এর মধ্যে সংযুক্ত ক্রিস্টালের ফ্রিকুয়েন্সি। মেশিন সাইকেলের জন্য ক্লক পিরিয়ডের সময়কাল ক্রিস্টাল ফ্রিকুয়েন্সির একটি ফাংশন।

৩। 8051 এর সি কম্পাইলার বাছাই।

অ্যাসেম্বলি ভাষা প্রোগ্রামিং এ যেহেতু নির্দেশনা সংখ্যা এবং প্রতি নির্দেশনায় সাইকেল সংখ্যা জানা, তাই উৎপন্ন ডিলে জানা ও নিয়ন্ত্রণ করা যায়। সি প্রোগ্রামের ক্ষেত্রে সি কম্পাইলার সি স্টেটমেন্টসমূহকে অনুবাদ করে এবং অ্যাসেম্বলি ভাষা প্রোগ্রামের ফাংশনে রূপান্তর করে। তাই ভিন্ন ভিন্ন কম্পাইলার ভিন্ন ভিন্ন কোড উৎপন্ন করে। নিচে লুপ ব্যবহার করে 8051 এর P1 পোর্টের বিটগুলোকে সামান্য ডিলেসহ toggle করার জন্য একটি সি প্রোগ্রাম দেখানো হলো।

```
#include < reg51.h>
Void main (void)
    {
        Unsighned int x;
        For (;;)
            {
                P1 = 0x55;
                For (x =0; x < 4000; x ++);
                P1 = 0 x AA;
                For (x = 0 x AA;
                For (x = 0; x < 4000; x ++)
            }
    }
```

## ৭.৪ 8051 এর জন্য সি প্রোগ্রাম

**(Write Program in C for Sending Data to Port, Accessing Code ROM, Data Serialization and Interrupt Operation)**

**প্রোগ্রাম- ১ : 8051 এর পোর্ট P1 এ – 4 থেকে + 4 মানগুলো পাঠানোর জন্য সি প্রোগ্রাম লিখ।**

**(Write an 8051 C program to send values of – 4 to + 4 port P1)**

**সমাধান :**

```
// Signed numbers
# include <reg51.h>
void main (void)
{
char mynum [] = { + 1, – 1, + 2, – 2, + 3, – 3, + 4, – 4};
unsigned char z;
for (z = 0; z < = 8; z ++}
pl = mynum [z];
}
```

**প্রোগ্রাম- ২ : 8051 এর পোর্ট P1 এ ডাটা পাঠানোর জন্য সি প্রোগ্রাম লিখ।**

**(Write a 8051 C program to send values 00–FF to port P1)**

**সমাধান :**

```
# include <reg51.h>
void main (void)
{
unsigned char z;
for (z = 0; z < = 255; z ++)
P1 = z;
}
```

**প্রোগ্রাম- ৩ : 8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে প্রেরণের জন্য সি প্রোগ্রাম লিখ। LSB (Least Significant Bit) প্রথমে বের হতে হবে। (Write a C program to send out the value 44H serially one bit at a time via P1.0. The LSB should go out first)**

**সমাধান :**

```
# include < reg51.h>
sbit p1b0 = p1^0;
sbit regALSB = ACC^0.
void main (void)
{
unsigned char conbyte = 0 x 44;
unsigned char x;
ACC = conbyte;
for (x = 0; x < 8; x ++)
    {
    P1b0 = regALSB;
    ACC = ACC >>1;
    }
}
```

```
unsigned char x;
for (x = 0; x < 8; x ++)
    {
    membit = P1b0;
    ACC = ACC <<1;
    regALSB = membit;
    }
P2 = ACC;
}
```

**প্রোগ্রাম- ৭ : 8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে প্রেরণের জন্য সি প্রোগ্রাম লিখ। MSB (Most Significant Bit) প্রথমে বের হতে হবে।**

**(Write a C program to send out the value 44H serially one bit at a time via P1.0. The MSB should go out first)**

**সমাধান :**

```
# include < reg51.h>
sbit P1b0 = P1^0;
sbit regAMSB = ACC^7;
void main (void)
{
unsigned char conbyte = 0 × 44;
unsigned char x;
ACC = conbyte;
for (x = 0; x < 8; x ++)
    {
    P1b0 = regAMSB;
    ACC = ACC <<1;
    }
}
```

**প্রোগ্রাম- ৮ : P0 পোর্ট থেকে ডাটার একটি বাইট গ্রহণ করে যদি এটি 100 এর চেয়ে কম হয়, তবে একে P1 পোর্টে পাঠাও অন্যথায় P2 পোর্টে পাঠানোর 8051 সি প্রোগ্রাম লিখ।**

**(Write an 8051 C program to get a byte of data from P0. If it is less than 100, send it to P1; otherwise, send it to P2)**

**সমাধান :**

```
#include <reg51.h>
void main (void)
{
Unsigned char mybyte;
P0 = 0 × FF                    // make P0 input port
while (1)
        {
        Mybyte = P0            // get a byte from P0
        If (mybyte < 100)
            P1 = mybyte;       //send it to P1
        Else
            P2 = mybyte;       //send it to P2
        }
}
```

**প্রোগ্রাম- ৯ : P1 পোর্ট থেকে একটি বাইট গ্রহণ করে এবং $\frac{1}{2}$ সেকেন্ড অপেক্ষা করে এবং পরে এটিকে P2 পোর্ট এ পাঠানোর 8051 সি প্রোগ্রাম লিখ। (Write an 8051 C program to get a byte of data from P1, wait $\frac{1}{2}$ second and then send it to P2)**

**সমাধান :**

```
#include <reg51.h>
void MSDelay (unsigned int);
void main (void)
{
        unsigned char mybyte;
        P1 = 0 × FF;                    //make P1 input port
        While (1)
            {
                Mybyte = P1;            //get a byte from P1
                MSDelay (500);
                P2 = mybyte;            //send it to P2
            }
    }
```

**প্রোগ্রাম- ১০ : RAM এ ডাটা জমা রাখার প্রোগ্রাম লিখ (Write a program to store data into RAM)**

**সমাধান :**

```
#include < reg51.h>
Void main (void)
{
        unsinged char mydata [ ] = "hello";
        unsigned char z;
        for (z = 0; z <5; z ++)
            P1 = mydata [z];
}
```

**প্রোগ্রাম- ১১ : ROM এ ডাটা জমা রাখার প্রোগ্রাম লিখ (Write a program to store data into ROM)**

**সমাধান :**

```
#include < reg51.h>
Void main (void)
{
        code unsinged char mydata [ ] = "hello";
        unsigned char z;
        for (z = 0; z <5; z ++)
            P1 = mydata [z];
}
```

**প্রোগ্রাম- ১২ : কোন 8051 মাইক্রোকন্ট্রোলারের ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মে. হার্টস এবং baud rate 9600 হলে নিচের কর্মকাণ্ডসমূহের জন্য সৃষ্ট ইন্টারাপ্ট ব্যবহার করে একটি সি প্রোগ্রাম লিখ।**

**(ক) T0 8-bit auto-reload টাইমার ব্যবহার করে P2.1-এ 10 কিলোহার্টস ফ্রিকুয়েন্সি উৎপাদন।**

**(খ) একটি 1 হার্টস পালস গণনার জন্য টাইমার 1-কে ইভেন্ট কাউন্টার হিসেবে ব্যবহার এবং এটিকে P0 তে প্রদর্শন।**

**সমাধান :**

```
#include <reg51. h>
        Sbit WAVE = P2^1;
        Unsigned char cnt;
Void timer 0 ( ) interrupt 1
{
        WAVE = ~ WAVE;                  //toggle pin
}
Void timer 1 ( ) interrupt 3
{
        cnt ++;                         //increment counter
        P0 = cnt;                       //display value on pins
        }
void main ( )
{
        cnt = 0;                        // set counter to 0
        TMOD = 0 × 42;
        TH0 = 0 × – 46;                 //10 kHz
        IE = 0 × 86;                    //enable interrupts
        TR0 = 1;                        /start timer 0
while (1);                              //wait until interrupted
}
```

**প্রোগ্রাম- ১৩ : কোন 8051 মাইক্রোকন্ট্রোলারের ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মে. হার্টস এবং baud rate 4800 হলে নিচের কর্মকাণ্ডসমূহের জন্য সৃষ্ট ইন্টারাপ্ট ব্যবহার করে একটি সি প্রোগ্রাম লিখ।**

**(ক) সিরিয়াল ডাটা গ্রহণ করে তাকে P0 port এ প্রেরণ**

**(খ) পোর্ট P1 পঠন, ডাটা সিরিয়ালি সম্প্রচার এবং P2 পোর্টে কপি করা**

**(গ) টাইমার 0 কে P0.1 এ 5 কিলোহার্টসের একটি স্কয়ার ওয়েব উৎপাদন।**

**সমাধান :**

```
#include < reg51.h>
Sbit WAVE = P0^1;
Void timer 0 ( ) interrupt 1
{
        WAVE = ~ WAVE;                  //toggle pin
}
Void serial0 ( ) interrupt 4
{
        If (TI = = 1)
{
```

```
            TI = 0                  //clear interrupt
        }
        else
    {
            P0 = SBUF;              //put value on pins
            RI = 0;                 // clear interrupt
        }
    }
    void main ( )
    {
    unsigned char x ;               //make P1 an input
    P1 = 0 × FF;
    TMOD = 0 × 22;                  //4800 baud rate
    TH1 = 0 × F6;
    SCON = 0 × 50;
    TH0 = 0 × A4;                   //5 KHz has T = 2000 us
    IE = 0 × 92;                    //enable interrupts
    TR1 = 0 × 92;                   //start timer1
    TR0 = 1;                        //start timer 0
    while (1)
    {
        x = P1;                     //read value from pins
        SBUF = x;                   //put value in buffer
        P2 = x;                     //write value to pins
        }
    }
```

# অনুশীলনী-০৭

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **অ্যাসেম্বলি ভাষায় মাইক্রোকন্ট্রোলার প্রোগ্রাম রচনায় অসুবিধা কী?**

**উত্তর :** অ্যাসেম্বলি ভাষা কোন বিশেষ মাইক্রোকন্ট্রোলারের জন্য নির্দিষ্ট। ফলে মাইক্রোকন্ট্রোলার পরিবার পরিবর্তন করলে এর জন্য বিশেষ অ্যাসেম্বলি ভাষাও জানতে হয়।

২। **মাইক্রোকন্ট্রোলার প্রোগ্রামিং-এ সি ভাষার অসুবিধা কী?**

**উত্তর :** সি ভাষার প্রধান অসুবিধা হলো– প্রোগ্রাম বড় হয় এবং ফলে বড় প্রোগ্রাম মেমোরির প্রয়োজন। তাই ছোট কোড মেমোরির মাইক্রোকন্ট্রোলারের ক্ষেত্রে এটি উপযোগী নয়।

৩। **সি ভাষায় x++ দ্বারা কী বুঝায়?**

**উত্তর :** কোন স্টেটমেন্টে x++ ব্যবহারের পূর্বে x-এর মান 1 হ্রাস করে।

৪। **সি ভাষায় x?y : z দ্বারা কী বুঝায়?**

**উত্তর :** যদি x শূন্য না হয় তবে y মূল্যায়ন করে, অন্যথায় z মূল্যায়ন করে।

৫। **সি ভাষায় x++, y++ দ্বারা কী বুঝায়?**

**উত্তর :** প্রথমে x এর ক্রমবৃদ্ধি হয় এবং পরে y এর ক্রমবৃদ্ধি হয়।

৬। **সি ভাষায় ~x দ্বারা কী বুঝায়?**

**উত্তর :** এটি সি ভাষায় Bitwise complement NOT। অর্থাৎ এটি বিট 1 কে 0 তে এবং বিট 0 কে 1 এ রূপান্তর করে।

৭। **8051 মাইক্রোকন্ট্রোলারের জন্য সি ভাষায় কয়ভাবে টাইম ডিলে তৈরি করা যায়?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের জন্য সি ভাষায় দুই ভাবে টাইম ডিলে তৈরি করা যায় :

১। 8051 টাইমার ব্যবহার করে।

২। একটি সরল লুপ ব্যবহার করে।

৮। **8051 মাইক্রোকন্ট্রোলারের জন্য সি ভাষায় লুপ ব্যবহার করে টাইম ডিলে তৈরির জন্য এর সঠিকতায় প্রভাব বিস্তারকারী বিবেচ্য বিষয়গুলো কী কী?**

**উত্তর :** লুপ ব্যবহার করে টাইম ডিলে তৈরির জন্য তিনটি ফ্যাক্টর বিবেচনা করতে হয়, যা ডিলে এর সঠিকতায় প্রভাব বিস্তার করতে পারে :

১। মেশিন সাইকেলের সংখ্যা এবং প্রতি মেশিন সাইকেলে ব্লকের সংখ্যা।

২। XTAL1 ও XTAL2 এর মধ্যে সংযুক্ত ক্রিস্টালের ফ্রিকুয়েন্সি।

৩। কম্পাইলার বাছাই।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **8051 এর জন্য সি ভাষায় bitwise operator গুলোর নাম লিখ।** *[বাকাশিবো : '১৫]*

**উত্তর :** 8051 এর জন্য সি ভাষার bitwise operator গুলোর নাম নিচে দেওয়া হলো :

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| ~ | Bitwise complement NOT | ~ x | বিট 1 কে 0 তে এবং বিট 0 কে 1 এ রূপান্তর করে। |
| & | Bitwise AND | x&y | x ও y এর Bitwise AND |
| \| | Bitwise OR | x\|y | x ও y এর Bitwise OR |
| ^ | Bitwise Exclusive OR | x ^ y | x ও y এর XOR |
| << | Left shift | x<<2 | x এর বিট বাম দিকে 2 বিট পজিশনে স্থানান্তরিত হয়। |
| >> | Rigth shift | x >> 2 | x এর বিট ডান দিকে 2 বিট পজিশনে স্থানান্তরিত হয়। |

২। **C কে Mid level language বলা হয় কেন?**

*[বাকাশিবো : '০২, '০৩, '০৪, '০৫, '০৬, '০৭, '০৮, '০৯, '১০R, '১১, '১১R, '১২, '১২R, '১৫, '১৫R]*

**উত্তর :** C একটি mid level (মধ্যম পর্যায়ের) language কারণ এতে যেমন Assemdly language এর মতো bit, byte, memory address নিয়ে কাজ করা যায়, অন্যদিকে তেমনি high level language-এর মতো বিভিন্ন data type নিয়ে কাজ করা যায়। তাই এটাকে মিড লেভেল ল্যাংগুয়েজ বলা হয়।

৩। **C কে Structured language বলা হয় কেন?** *[বাকাশিবো : '০৪, '০৫, '০৬, '০৮, '০৯, '১০, '১৩, '১৩R, '১৪]*

**উত্তর :** C তে একটি বড় প্রোগ্রামকে ভেঙে ছোট ছোট অংশে বা মডিউলে ভাগ করে কার্য সম্পাদন করা যায়। তাই C-কে স্ট্রাকচার্ড প্রোগ্রামিং ল্যাংগুয়েজ বলা হয়।

৪। **C কে পোর্টেবল ল্যাংগুয়েজ বলা হয় কেন?** *[বাকাশিবো : '০৮, '১২, '১৩R, '১৪, '১৪R]*

**উত্তর :** C একটি Portable ল্যাংগুয়েজ। কারণ এক Computer -এ (যেমন IBM PC) লিখিত C Program অন্য কোম্পানির তৈরি Computer- এও (যেমান EDC VAX) কম্পাইল এবং Runকরা যায়।

৫। **কয়েকটি High level language এর নাম লেখ।** *[বাকাশিবো : '০৯, '১৩R]*

**অথবা, কয়েকটি OOP Language এর নাম লেখ।**

**উত্তর :** BASIC = Beginers All purpose Symbolic Instruction Code

COBOL = Common Business Oriented Language

Prolog = Programming in Logic

C, C++, visual BASIC

৬। **C কে Case sensitive language বলা হয় কেন?** *[বাকাশিবো : '১২]*

**উত্তর :** সি প্রোগ্রামিং ল্যাংগুয়েজে প্রোগ্রাম লিখার ক্ষেত্রে বড় হাতের অক্ষর ও ছোট হাতের অক্ষরের মধ্যে পার্থক্য বিদ্যামান। অর্থাৎ কোথায় বড় হাতের অক্ষর করা যাবে আর কোথায় কোথায় ছোট হাতের অক্ষরে লিখতে হতে তার সুনির্দিষ্ট নিয়ম বিদ্যামান। তাই 'সি' Case sensitive language কে বলা হয়।

৭। **C-প্রোগ্রাম কম্পাইলেশন ও নির্বাহ প্রক্রিয়ার ধাপগুলো লিখ।** *[বাকাশিবো : '০৫]*

**উত্তর :** C-প্রোগ্রাম কম্পাইলেশন ও নির্বাহ প্রক্রিয়ার ধাপগুলো হচ্ছে : (i) প্রোগ্রাম তৈরিকরণ, (ii) প্রোগ্রাম কম্পাইলকরণ, (iii) প্রোগ্রামের সাথে লাইব্রেরি সংযুক্তকরণ এবং (iv) প্রোগ্রাম নির্বাহকরণ।

৮। **C-প্রোগ্রাম কখন, কোথায় এবং কে তৈরি করেন?** *[বাকাশিবো : '০১, '০২]*

**উত্তর :** 1972 সালে আমেরিকার Bell ল্যাবরেটরিতে Dennis Ritchie প্রথম C-প্রোগ্রাম তৈরি করেন।

৯। **মাইক্রোকন্ট্রোলারের জন্য সি ভাষায় প্রোগ্রাম লিখার সুবিধা কী?**

**উত্তর :** নিচে মাইক্রোকন্ট্রোলারে সি প্রোগ্রাম ব্যবহারের সুবিধ্য উল্লেখ করা হলো :

১। ফাংশন লাইব্রেরির কোড ব্যবহার করা যায়।

২। সি প্রোগ্রাম সহজ বোধগম্য এবং একে সংরক্ষণও সহজ।

৩। অ্যাসেম্বলি ভাষার তুলনায় সি ভাষায় প্রোগ্রাম রচনা সহজ এবং সময় অনেক কম লাগে।

৪। একে পরিবর্তন ও হালনাগাদ করা সহজ।

৫। সব ধরনের মাইক্রোকন্ট্রোলারে প্রয়োগ করা যায়। ফলে ভিন্ন ভিন্ন মাইক্রোকন্ট্রোলারের জন্য ভিন্ন ভিন্ন এসেম্বলি ভাষা জানা প্রয়োজন নেই।

৬। এটা সুবহ (portable) অর্থাৎ একে সামান্য পরিবর্তন করে বা কোনরূপ পরিবর্তন না করেই সহজে অন্য মাইক্রোকন্ট্রোলারে ব্যবহার করা যায়।

**১০। 8051 এর জন্য সি ভাষার ডাটার ধরন উল্লেখ কর।**

**উত্তর :** 8051 এর জন্য প্রোগ্রামিং "সি" এর ডাটার ধরনের উপর ভাল ধারণা প্রোগ্রামারকে ছোট hex file তৈরি করতে সাহায্য করে। নিচে সি প্রোগ্রামে বহুল ব্যবহৃত ডাটা শ্রেণির আকার ও রেঞ্জসহ তালিকা দেওয়া হলো :

| ডাটার শ্রেণি | আকার (bit) | ডাটা রেঞ্জ/ব্যবহার |
|---|---|---|
| Unsigned char | 8-bit | 0 – 255 |
| (signed) char | 8-bit | (–128) – (+ 127) |
| Unsigned int | 16-bit | 0 – 65535 |
| (signed) int | 16-bit | (– 32768) – (+ 32767) |
| Sbit (single bit) | 1-bit | শুধুমাত্র SFR bit-addressable |
| Bit | 1-bit | RAM bit-addressable |
| Sfr | 8-bit | শুধুমাত্র RAM addressable 80–FFH |

**১১। 8051 এর জন্য সি ভাষার relational ও logical operator গুলোর নাম লিখ।**

**উত্তর :** 8051 এর জন্য সি ভাষার relational ও logical operator গুলোর নাম নিম্নরূপ :

| অপারেটর | নাম | উদাহরণ | বর্ণনা |
|---|---|---|---|
| > | Greater than | x > y | যদি x এর মান y এর চেয়ে বড় হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| >= | Greater than or equal to | x > = y | যদি x এর মান y এর চেয়ে বড় বা সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| < | Less than | x < y | যদি x এর মান y এর চেয়ে ছোট হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| < = | Less than or equal | x < = y | যদি x এর মান y এর চেয়ে ছোট হয় বা সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| = = | Equal to | x = y | যদি x এর মান y এর সমান হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| != | Not equal to. | x != y | যদি x এর মান y মান সমান না হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| ! | Logical NOT | !x | যদি x এর মান 0 হয়, তবে লজিক 1, অন্যথায় 0 হয়। |
| && | Logical AND | x&&y | যদি x বা y এর মান 0 হয়, তবে লজিক 0, অন্যথায় 1 হয়। |
| \|\| | Logical OR | x\|\|y | যদি x ও y এর মান উভয়ই 0 হয়, তবে লজিক 0, অন্যথায় 1 হয়। |

**১২। 8051 এর পোর্ট P1 এ – 4 থেকে + 4 মানগুলো পাঠানোর জন্য সি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ১ নং দ্রষ্টব্য।

**১৩। 8051 এর পোর্ট P1 এ ডাটা পাঠানোর জন্য সি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ২ নং দ্রষ্টব্য।

**১৪। 8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে প্রেরণের জন্য সি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৩ নং দ্রষ্টব্য।

**১৫। 8051 এর পোর্ট P1 এ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C ও D এর ASCII অক্ষর পাঠানোর জন্য সি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৪ নং দ্রষ্টব্য।

**১৬। 8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে আনয়নের সি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৫ নং দ্রষ্টব্য।

১৭। **8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে আনয়নের জন্য সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৬ নং দ্রষ্টব্য।

১৮। **8051 এর পোর্ট P1 এর মাধ্যমে সিরিয়ালি এক বিট করে 44H মানকে প্রেরণের জন্য সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৭ নং দ্রষ্টব্য।

১৯। **P0 পোর্ট থেকে ডাটার একটি বাইট গ্রহণ করে যদি এটি 100 এর চেয়ে কম হয়, তবে একে P1 পোর্টে পাঠাও অন্যথায় P2 পোর্টে পাঠানোর 8051 সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৮ নং দ্রষ্টব্য।

২০। **P1 পোর্ট থেকে একটি বাইট গ্রহণ করে এবং $\frac{1}{2}$ সেকেন্ড অপেক্ষা করে এবং পরে এটিকে P2 পোর্ট এ পাঠানোর 8051 সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ৯ নং দ্রষ্টব্য।

২১। **প্রোগ্রাম- ১০ : RAM এ ডাটা জমা রাখার প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ১০ নং দ্রষ্টব্য।

২২। **প্রোগ্রাম- ১১ : ROM এ ডাটা জমা রাখার প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ১১ নং দ্রষ্টব্য।

২৩। **কোন 8051 মাইক্রোকন্ট্রোলারের ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মে. হার্টস এবং baud rate 9600 হলে নিচের কর্মকাণ্ডসমূহের জন্য সৃষ্ট ইন্টারাপ্ট ব্যবহার করে একটি সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ১২ নং দ্রষ্টব্য।

২৪। **কোন 8051 মাইক্রোকন্ট্রোলারের ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মে. হার্টস এবং baud rate 4800 হলে নিচের কর্মকাণ্ডসমূহের জন্য সৃষ্ট ইন্টারাপ্ট ব্যবহার করে একটি সি প্রোগ্রাম লিখ।**
**উত্তর :** অনুচ্ছেদ ৭.৪ প্রোগ্রাম- ১৩ নং দ্রষ্টব্য।

২৫। **কয়েকটি C Compiler এর নাম লেখ।**
**উত্তর :** UNIX C/ Traditional C, ANSI C, Borland C & Turbo C, Microsoft C, Zortech C.

২৬। **হাই লেভেল ল্যাংগুয়েজ কাকে বলে?**
**উত্তর :** প্রোগ্রাম রচনার জন্য সহজে বোধগম্য সার্বজনীন ভাষা যেখানে স্বাভাবিগ ভাষায় (ইংরেজি) অনেক শব্দ ব্যবহার করা হয় তাকে হাই লেভেল ল্যাংগুয়েজ (High level language) বলে।

২৭। **কয়েকটি Mid level language এর নাম লিখ।**
**উত্তর :** কয়েকটি Mid level language হলো 'C', 'Forth', 'Macro Assembler' ইত্যাদি।

২৮। **মেশিন ল্যাংগুয়েজ প্রোগ্রাম কী?**
**উত্তর :** কম্পিউটার মেশিনের ভাষা হচ্ছে 1 এবং 0. এই 1 এবং 0 কে ব্যবহার করে যখন কম্পিউটারে ইন্সট্রাকশন প্রদান করে প্রোগ্রাম তৈরি করা হয়, তখন এ ধরনের প্রোগ্রামকে মেশিন ল্যাংগুয়েজ প্রোগ্রাম বলে।

২৯। **C প্রোগ্রামিং কে আবিষ্কার করেন?**
**উত্তর :** Getchar ( ), scanf ( ), fscanf ( ), printf ( ) ইত্যাদি।

## রচনামূলক প্রশ্নোত্তর

১। **8051 এর জন্য সি ভাষায় টাইম ডিলে তৈরির প্রক্রিয়া উদাহরণসহ বর্ণনা কর।**
**উত্তর :** অনুচ্ছেদ ৭.৩ নং দ্রষ্টব্য।

২। **8051 এর জন্য সি ডাটার ধরন ও অপারেটরের তালিকা লেখ।**
**উত্তর :** অনুচ্ছেদ ৭.২ নং দ্রষ্টব্য।

৩। **সি ভাষায় প্রোগ্রাম লেখার কারণ বর্ণনা কর।**
**উত্তর :** অনুচ্ছেদ ৭.১ নং দ্রষ্টব্য।

# অধ্যায় ৮ ইনপুট/আউটপুট পোর্ট প্রোগ্রামিং
# Understand I/O Port Programming

## ৮.১ 8051 এর ইনপুট/আউটপুট পোর্টের তালিকা
## (List the I/O Ports of 8051)

**8051 মাইক্রোকন্ট্রোলারে 4 বিটের এবং 8 লাইনের চারটি ইনপুট/আউটপুট পোর্ট আছে। এরা হলো–**

1. P0-Port 0, Address 80h, Bit-Addressable,
2. P1-Port 1, Address 90h, Bit-Addressable,
3. P2-Port 2, Address A0h, Bit-Addressable ও
4. P3-Port 3, Address B0h, Bit-Addressable।

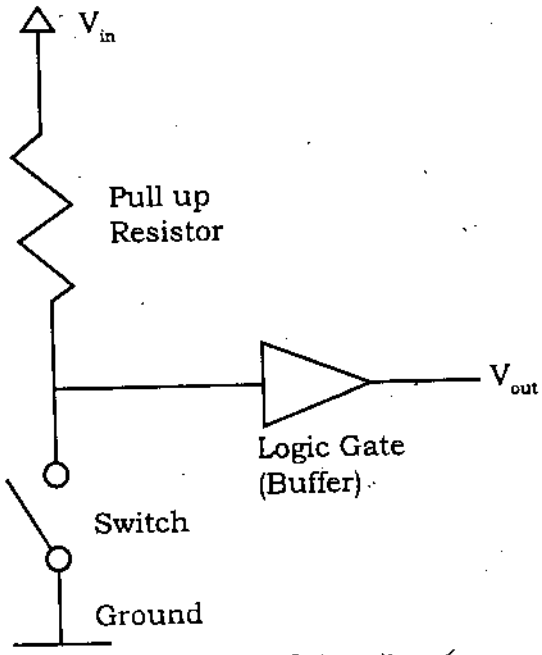
আউটপুট তখনই হয় যখন কোন পোর্টে প্রথম 0 লিখা হয়। ইনপুট হিসেবে এটিকে আবার পুনঃসজ্জা করতে চাইলে পোর্টে অবশ্যই একটি 1 পাঠাতে হবে। এই পোর্টসমূহের যে কোনটিকে একটি ইনপুট পোর্ট হিসেবে ব্যবহার করতে চাইলে এটিকে প্রোগ্রাম করতে হয়।

## ৮.২ পুল-আপ রেজিস্টরের উদ্দেশ্য উল্লেখ করণ
## (Mention the Purpose of Pull-up Resistor)

আমরা জানি, ডিজিটাল লজিক সার্কিটে তিনটি অবস্থা (State) থাকে, যথা– High, Low ও Floating (High Impedance)। যদি কোন ইলেকট্রনিক লজিক সার্কিটের সাথে সংযুক্ত বাহ্যিক ডিভাইসসমূহে High impedance সূচিত হয় তবে লজিক সিস্টেমের ইনপুটসমূহকে কোন আকাঙ্ক্ষিত লজিক লেবেল নিশ্চিত করার জন্য ইলেকট্রনিক লজিক সার্কিটে pull-up রেজিস্টর ব্যবহৃত হয়। যখন কোন সিগন্যালের উৎস সংযোগ না থাকে তখন এটি ইনপুটের অবস্থা নিরূপণ করে। এটি সিগন্যালের নয়েজের স্তর কমায়। এরা বিভিন্ন ধরনের লজিক ডিভাইসের ভিন্ন পাওয়ার সাপ্লাই সিস্টেমে চালিত হলেও মধ্যে ইন্টারফেস হিসেবেও ব্যবহৃত হয়। নিচে চিত্র ৮.১ এ একটি pull-up রেজিস্টর নেটওয়ার্ক দেখানো হলো। এখানে যখন সুইচ খোলা থাকে তখন গেইটের ইনপুট ভোল্টেজ ($V_{in}$) লেবেলে pulled-up হয় এবং সুইচ বন্ধ হলে গেইট ইনপুট ভোল্টেজ গ্রাউন্ড হয়।

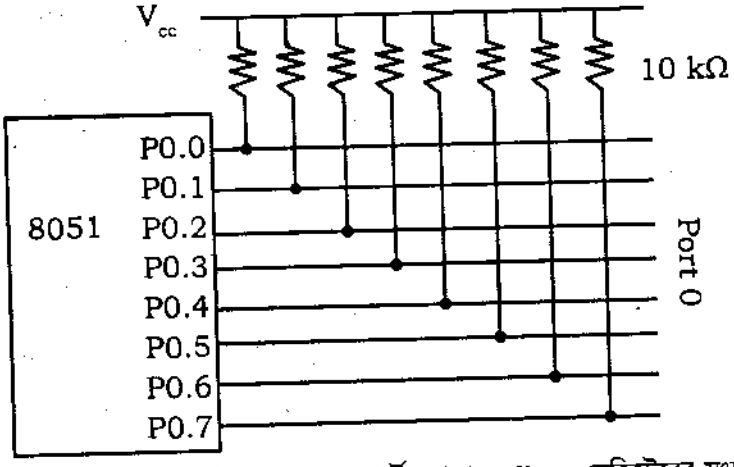
ইনপুট সিগন্যালের অনুপস্থিতিতে কোন তারকে একটি উচ্চ লজিক্যাল লেবেলে pull করার নিশ্চিতকরণের জন্য যে রেজিস্টর ব্যবহৃত হয় তাকে pull-up রেজিস্টর বলে। এরা যে কোন অবস্থায় কোন পিনে একটি সুস্পষ্ট লজিক্যাল লেবেল নিশ্চিতকরণের জন্য ডিজিটাল লজিক সার্কিটে ব্যবহৃত হয়।

একটি পিনে যখন High বা Low লজিক লেবেল Pull না হয় তখন ঐ পিনে High impedance state হয় এবং state floating অবস্থায় থাকে। এর একটি ভাল উদাহরণ হলো : কোন মাইক্রোকন্ট্রোলারের একটি সংযোগবিহীন ইনপুট পিনের অবস্থা High বা Low কোনটিই নয়। ফলে মাইক্রোকন্ট্রোলার এর ইনপুট মানকে নিশ্চিতভাবে ব্যাখ্যা করতে পারে না এবং একে হয় High অথবা Low-তে অনিশ্চিতভাবে ব্যাখ্যা (unpredictable interpret) করে। pull-up রেজিস্টর ইনপুটকে একটি লজিক্যাল High state-এ pull করে মাইক্রোকন্ট্রোলারকে এই সংকট থেকে উত্তরণ করে। যদি pull-up রেজিস্টর না থাকে তবে সুইচ খোলা থাকলে মাইক্রোকন্ট্রোলারের ইনপুট float করে।




চিত্র-৮.১ : pull-up রেজিস্টর নেটওয়ার্ক

pull-up রেজিস্টর কোন বিশেষ ধরনের রেজিস্টর নয়। এটি একটি স্থির মানের সরল রেজিস্টর যা পাওয়ার সাপ্লাই (সাধারণত + 5 ভোল্ট) ও যথাযথ পিনের মাঝখানে যুক্ত থাকে। এর মান নির্ণয়ের জন্য যে নিয়মটি অনুসরণ করা হয়, তা হলো pull-up রেজিস্টরের মান সিগন্যাল সোর্স ইম্পিডেন্সের তুলনায় অন্তত 10 গুণের চাইতে বড় হয়। এটি সাধারণত 10 কিলোওহম বা 100 কিলোওহমের হয়। 8051 এ পোর্ট P1, P2 ও P3 এর অভ্যন্তরীণ pull-up রেজিস্টর আছে। কিন্তু পোর্ট P0 এর অভ্যন্তরীণ pull-up রেজিস্টর নেই, তাই বাহ্যিকভাবে সংযোগ করতে হয়। নিচের চিত্রে পোর্ট P0 এর সাথে রেজিস্টরের সংযোগ দেখানো হলো :




চিত্র-৮.২ : 8051 মাইক্রোকন্ট্রোলারের P0 পোর্টের সাথে pull-up রেজিস্টরের সংযোগ

"Pull-down" রেজিস্টর Pull-up রেজিস্টরের মতো সবকিছু একই তবে এটি পাওয়ার সাপ্লাইয়ের গ্রাউন্ডের সাথে যুক্ত থাকে এবং এটি লজিক লেবেল প্রায় জিরোই ধারণ করে।

## ৮.৩ 8051 এর পোর্টকে ইনপুট হিসেবে সাজানোর প্রোগ্রাম (Write Code for Configuring Port for Input)

**প্রোগ্রাম ১ :** পোর্ট P0- কে ইনপুট পোর্ট হিসেবে ব্যবহারের জন্য প্রোগ্রাম লিখ।

**সমাধান :** পোর্ট P0-কে ইনপুট করার জন্য এর প্রত্যেক বিটে 1 লিখে প্রোগ্রাম করতে হবে। নিচে প্রোগ্রামটি দেখানো হলো :

```
        MOV     A, #0FFH    ; A = FF hex
        MOV     P0, A       ; make P0 an i/p port by writing it all 1's
Back :  MOV     A, P0       ; get data from P0
        MOV     P1, A       ; send it port 1
        SJMP    BACK        ; keep doing it
```

**প্রোগ্রাম ২ :** পোর্ট P1-কে ইনপুট পোর্ট হিসেবে ব্যবহারের জন্য প্রোগ্রাম লিখ।

**সমাধান :** পোর্ট P1-কে ইনপুট করার জন্য এর প্রত্যেক বিটে 1 লিখে প্রোগ্রাম করতে হবে। নিচে Port-1-কে ইনপুট করে এটি থেকে ডাটা গ্রহণপূর্বক ডাটাকে R7 ও R5 রেজিস্টরে জমা করার প্রোগ্রাম দেখানো হলো :

```
MOV     A, #0FFH    ; A = FF hex
MOV     P, A        ; make P1 an input port
                    ; by writing it all 1's
MOV     A, P1       ; get data from P1
MOV     R7, A       ; save it to in register R7
ACALL   DELAY       ; wait
MOV     A, P1       ; another data from P1
MOV     R5, A       ; save it to in register R5
```

প্রোগ্রাম ২ ও ৩ কে অনুসরণ করে : পোর্ট P2 ও P3-কে ইনপুট পোর্ট হিসেবে ব্যবহার করা যায়।

## ৮.৪ ইনপুট/আউটপুট পোর্টের মাধ্যমে ডাটা গ্রহণ ও প্রেরণের প্রোগ্রাম (Write Program for Receiving and Sending data Through I/O Port)

**প্রোগ্রাম ১ :** P1.0 পিনের সাথে একটি সুইচ এবং P2.7 পিনের সাথে একটি LED সংযুক্ত আছে। সুইচের অবস্থা জানা এবং একে LED-এ পাঠানোর প্রোগ্রাম লিখ।

**সমাধান :**

```
        SETB P1.7       ; make P1.7 an input
AGAIN : MOV C, P1.0     ; read SW status into CF
        MOV P2.7, C     ; send SW status to LED
        SJMP AGAIN      ; keep repeating
```

**প্রোগ্রাম ২ :** P1.7 পিনে একটি সুইচ সংযুক্ত আছে। সুইচের অবস্থা জানার ও নিচের শর্তগুলোর জন্য একটি প্রোগ্রাম লিখ :

(ক) যদি সুইচ অফ থাকে তবে P2 তে N পাঠাও

(খ) যদি সুইচ অন থাকে তবে P2 তে Y পাঠাও।

**সমাধান :**

```
        SETB    P1.7        ; make P1.7 an input
AGAIN : MOV     C, P1.2     ; read SW status into CF
        JC      OVER        ; jump if SW = 1
        MOV     P2, # 'N'   ; SW = 0, issue 'N' to P2
        SJMP    AGAIN       ; keep monitoring
OVER :  MOV     P2 # 'Y'    ; SW = 1, issue 'Y' to P2
        SJMP    AGAIN       ; keep monitoring
```

## ৮.৫ পোর্ট 3 এর বিকল্প কাজ বর্ণনা

### (Describe the Alternate Functions of Port-3)

একটি দ্বৈত কাজের পোর্ট (পিন 10 – পিন 17) হলো PORT 3 (P0)। তবে এটি সাধারণ কাজের ইনপুট/আউটপুট। পাশাপাশি এর পিনগুলোর ইনপুট/আউটপুট কাজ ছাড়াও প্রত্যেকের একটি বিকল্প কাজ আছে। নিচে PORT 3 এর বিকল্প কাজের ছক দেখানো হলো :

| BIT | নাম | Bit address | বিকল্প কাজ |
|---|---|---|---|
| P3.0 | RXD | B0H | সিরিয়াল পোর্টের জন্য ডাটা গ্রহণ করে। |
| P3.1 | TXD | B1H | সিরিয়াল পোর্টের ডাটা প্রেরণ করে। |
| P3.2 | INT0 | B2H | বাহ্যিক ইন্টারাপ্ট 0 |
| P3.3 | INT1 | B3H | বাহ্যিক ইন্টারাপ্ট 1 |
| P3.4 | T0 | B4H | টাইমার/কাউন্টার 0 বাহ্যিক ইনপুট |
| P3.5 | T1 | B5H | টাইমার/কাউন্টার 1 বাহ্যিক ইনপুট |
| P3.6 | WR | B6H | বাহ্যিক ডাটা মেমোরি write strobe |
| P3.7 | RD | B7H | বাহ্যিক ডাটা মেমোরি read strobe |

# অনুশীলনী–০৮

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **Pull-up রেজিস্টরের মান নির্ণয়ের নিয়ম কী?**

**উত্তর :** Pull-up রেজিস্টরের মান নির্ণয়ের জন্য যে নিয়মটি অনুসরণ করা হয় তাহলো : Pull-up রেজিস্টরের মান সিগন্যাল সোর্স ইম্পিডেন্সের তুলনায় অন্তত 10 গুণের চাইতে বড় হয়। এটি সাধারণত 10 কিলোওহম বা 100 কিলোওহমের হয়।

২। **8051 এ কোন পোর্টে অভ্যন্তরীণ Pull-up রেজিস্টার নেই।**

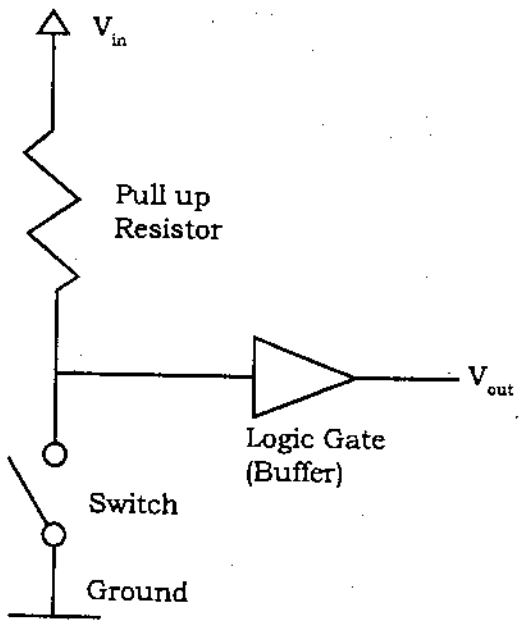
**উত্তর :** 8051 এ P0 পোর্টে অভ্যন্তরীণ Pull-up রেজিস্টার নেই।

৩। **Pull-up রেজিস্টর কাকে বলে?**

**উত্তর :** ইনপুট সিগন্যালের অনুপস্থিতিতে কোন তারকে একটি উচ্চ লজিক্যাল লেবেলে Pull করার নিশ্চিতকরণের জন্য যে রেজিস্টর ব্যবহৃত হয় তাকে Pull-up রেজিস্টর বলে।

৪। **একটি Pull-up রেজিস্টর নেটওয়ার্ক অঙ্কন কর।**

**উত্তর :** নিচে একটি Pull up রেজিস্টর নেটওয়ার্ক দেখানো হলো :




চিত্র : pull-up রেজিস্টর নেটওয়ার্ক

৫। **8051 এ কোন পোর্টে অভ্যন্তরীণ Pull-up রেজিস্টর থাকে?**

**উত্তর :** 8051 এ P1, P2 ও P3 পোর্টে অভ্যন্তরীণ Pull-up রেজিস্টর থাকে।

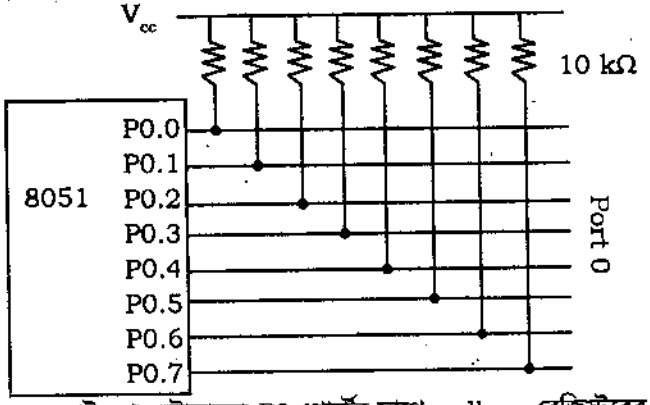
৬। **8051 এর ইনপুট/আউটপুট পোর্টের নাম লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারে 4 বিটের এবং 8 লাইনের চারটি ইনপুট/আউটপুট পোর্ট আছে। এগুলো হলো–

1. P0–Port 0, Address 80h, Bit-Addressable,
2. P1–Port 1, Address 90h, Bit-Addressable,
3. P2–Port 2, Address A0h, Bit-Addressable ও
4. P3–Port 3, Address B0h, Bit-Addressable।

**৭। 8051 মাইক্রোকন্ট্রোলারের P0 পোর্টের সাথে Pull-up রেজিস্টরের সংযোগ চিত্র অঙ্কন কর।**

**উত্তর :** যেহেতু 8051 মাইক্রোকন্ট্রোলারের P0 পোর্টের সাথে অভ্যন্তরীণ Pull-up রেজিস্টর নেই, তাই এর সাথে নিচের চিত্র অনুযায়ী P0 পোর্টের সাথে Pull-up রেজিস্টর সংযোগ করা হয়।




চিত্র : 8051 মাইক্রোকন্ট্রোলারের P0 পোর্টের সাথে pull-up রেজিস্টরের সংযোগ

## সংক্ষিপ্ত প্রশ্নোত্তর

**১। Pull-up রেজিস্টরের উদ্দেশ্য লিখ।**

**উত্তর :** নিচে Pull-up রেজিস্টরের উদ্দেশ্য উল্লেখ করা হলো :

(ক) ইনপুট সিগন্যালের অনুপস্থিতিতে কোন তারকে একটি উচ্চ লজিক্যাল লেবেলে Pull করার নিশ্চিত করে।

(খ) এরা যে কোন অবস্থায় কোন পিনে একটি সুস্পষ্ট লজিক্যাল লেবেল নিশ্চিত করে।

(গ) যদি কোন ইলেকট্রনিক লজিক সার্কিটের সাথে সংযুক্ত বাহ্যিক ডিভাইসসমূহে high impedance সূচিত হয় তবে লজিক সিস্টেমের ইনপুটসমূহকে কোন আকাঙ্ক্ষিত লজিক লেবেলে নিশ্চিত করার জন্য ইলেকট্রনিক লজিক সার্কিটে Pull-up রেজিস্টর ব্যবহৃত হয়।

(ঘ) যখন কোন সিগন্যালের উৎস সংযোগ না থাকে তখন এটি ইনপুটের অবস্থা নিরূপণ করে।

(ঙ) এটি সিগন্যালের নয়েজের স্তর কমায়।

(চ) এরা বিভিন্ন ধরনের লজিক ডিভাইসের (ভিন্ন পাওয়ার সাপ্লাই সিস্টেমে চালিত হলেও) মধ্যে ইন্টারফেস হিসেবেও ব্যবহৃত হয়।

**২। পোর্ট P3 এর বিকল্প কাজগুলো উল্লেখ কর।**

**উত্তর :** একটি দ্বৈত কাজের পোর্ট (পিন 10 – পিন 17) হলো PORT 3 (P0)। তবে এটি সাধারণ কাজের ইনপুট/আউটপুট। পাশাপাশি এর পিনগুলোর ইনপুট/আউটপুট কাজ ছাড়াও প্রত্যেকের একটি বিকল্প কাজ আছে। নিচে PORT 3 এর বিকল্প কাজের ছক দেখানো হলো :

| BIT | নাম | Bit address | বিকল্প কাজ |
|---|---|---|---|
| P3.0 | RXD | B0H | সিরিয়াল পোর্টের জন্য ডাটা গ্রহণ করে। |
| P3.1 | TXD | B1H | সিরিয়াল পোর্টের ডাটা প্রেরণ করে। |
| P3.2 | INT0 | B2H | বাহ্যিক ইন্টারাপ্ট 0 |
| P3.3 | INT1 | B3H | বাহ্যিক ইন্টারাপ্ট 1 |
| P3.4 | T0 | B4H | টাইমার/কাউন্টার 0 বাহ্যিক ইনপুট |
| P3.5 | T1 | B5H | টাইমার/কাউন্টার 1 বাহ্যিক ইনপুট |
| P3.6 | WR | B6H | বাহ্যিক ডাটা মেমোরি write strobe |
| P3.7 | RD | B7H | বাহ্যিক ডাটা মেমোরি read strobe |

৩। **পোর্ট P0- কে ইনপুট পোর্ট হিসেবে ব্যবহারের জন্য প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৮.৩ প্রোগ্রাম ১ নং দ্রষ্টব্য।

৪। **P1.0 পিনের সাথে একটি সুইচ এবং P2.7 পিনের সাথে একটি LED সংযুক্ত আছে। সুইচের অবস্থা জানা এবং একে LED-এ পাঠানোর প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৮.৪ প্রোগ্রাম ১ নং দ্রষ্টব্য।

## রচনামূলক প্রশ্নোত্তর

১। **মাইক্রোকন্ট্রোলারে Pull-up রেজিস্টরের উদ্দেশ্য আলোচনা কর।**

**উত্তর :** অনুচ্ছেদ ৮.২ নং দ্রষ্টব্য।

২। **কে ইনপুট পোর্ট হিসেবে ব্যবহারের জন্য প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৮.৩ প্রোগ্রাম ২ নং দ্রষ্টব্য।

৩। **P1.7 পিনে একটি সুইচ সংযুক্ত আছে। সুইচের অবস্থা জানার ও নিচের শর্তগুলোর জন্য একটি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ৮.৪ প্রোগ্রাম ২ নং দ্রষ্টব্য।

# অধ্যায় ৯ — 8051 এর টাইমার/কাউন্টার

# Understand the 8051 Timer/counter

## ৯.১ টাইমারের কাজের তালিকা

**(List the Function of a Timer)**

8051 মাইক্রোকন্ট্রোলারের ভিতরে দুটি টাইমার আছে। উভয়কেই আলাদাভাবে নিয়ন্ত্রণ, সেট গঠন ও সাজানো যায়। 8051 টাইমারের সাধারণ কাজ নিচে দেওয়া হলো :

(১) ফ্রিকুয়েন্সি পরিমাপ (Frequency measurements)

(২) পালসের প্রশস্ততা পরিমাপ (Pulse width measurements)

(৩) ঘটনা গণনা (Event counting)

(৪) সিরিয়াল কমিউনিকেশনের জন্য baud rate উৎপাদন।

(৫) বিরতির সময় নির্ধারণ (Interval timing)

(ক) পর্যায়বৃত্ত ঘটনার সময় নির্ধারণ (Periodic event timing)

(খ) পরিমাপের জন্য সময়ের ভিত্তি নির্ধারণ (Time base for measurements)

## ৯.২ 8051 এ টাইমার/কাউন্টার অপারেশনের সহযোগী রেজিস্টারসমূহ আলোচনা

**(Discuss the Mode of Operation of Timer)**

8051 মাইক্রোকন্ট্রোলারের টাইমার/কাউন্টার অপারেশনে সহযোগী রেজিস্টার হিসেবে নিচের রেজিস্টারসমূহ ব্যবহার করা হয় :

| | | | |
|---|---|---|---|
| 1. | TCON | (8 BIT) | টাইমার নিয়ন্ত্রণ রেজিস্টার। |
| 2. | TMOD | (8 BIT) | টাইমার মুড নির্ধারণ রেজিস্টার। |
| 3. | TL0 | (8 BIT) | TIMER 0 এর Lower byte ধারণ করে। |
| 4. | TH0 | (8 BIT) | TIMER 0 এর Upper byte ধারণ করে। |
| 5. | TL1 | (8 BIT) | TIMER 1 এর Lower byte ধারণ করে। |
| 6. | TH1 | (8 BIT) | TIMER 1 এর Upper byte ধারণ করে। |

### ৯.২.১ TMOD Register (Timer MODe Register) :

TMOD register টাইমার/কাউন্টার (0/1) এর অপারেটিং মুড নির্ধারণ করে। এটি bit-addressable রেজিস্টার নয়। এটির বিট সজ্জা নিম্নরূপ :

| (MSB) | | | | | | | (LSB) |
|---|---|---|---|---|---|---|---|
| Gate | $C/\bar{T}$ | M1 | M0 | Gate | $C/\bar{T}$ | M1 | M0 |
| ---------- Timer 1 ---------- | | | | ---------- Timer 0 ---------- | | | |

**বিটগুলোর কাজ :**

$Gate_x$ : $Timer_x$ কে $INT_x$ এর সিগন্যালের মাধ্যমে কার্যকর বা অকার্যকর করা হয়।

1 = $INT_x$ বিট সেট হলে $Timer_x$ কার্যকর হয়।

0 = $Timer_x$ কার্যকর হওয়ার জন্য $INT_x$ বিটের স্টেটের উপর নির্ভর করে না।

$C/\bar{T}$ : টাইমার/কাউন্টার নির্বাচন বিট

1 = $Counter_x$

0 = $Timer_x$

M0, M1- টাইমার মুড নির্বাচন বিট। টাইমার মুড নির্বাচনের জন্য নিচে এদের সমন্বয় দেখানো হলো :

| $T_xM_1$ | $T_xM_0$ | Timer Mode |
|---|---|---|
| 0 | 0 | Timer Mode 0 : 13-bit timer |
| 0 | 1 | Timer Mode 1 : 16-bit timer |
| 1 | 0 | Timer Mode 2 : 8-bit auto reload |
| 1 | 1 | Timer Mode 3 : Split timer mode |

বি: দ্র: Subscript x, – 0 বা 1 নির্দেশ করে, যা Timer 0 এবং Timer 1 প্রকাশ করে।

### ৯.২.২ TCON Register (Timer Control Register) :

TMOD রেজিস্টার কর্তৃক নির্ধারিত টাইমার বা কাউন্টারকে পরিচালনার জন্য এটি প্রয়োজনীয় কন্ট্রোল বিট ধারণ করে এবং বিভিন্ন ফ্ল্যাগ বিটগুলো (Status flag) তৈরি করে। এটি একটি bit-addressable রেজিস্টার। এর বিট-সজ্জা নিম্নরূপ :

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |

TF1/TF0 – Timer overflow flag bit. টাইমার overflow হলে এটি হার্ডওয়্যার দ্বারা সেট হয় এবং হার্ডওয়্যার বা সফটওয়্যার দ্বারা রিসেট হয়।

TR1/TR0 – টাইমার চালনার জন্য TR1/TR0 নিয়ন্ত্রণ বিট হিসেবে কাজ করে। এটি সেট হলে টাইমার অন হয় এবং রিসেট হলে টাইমার অফ হয়। এটি সফটওয়্যার দ্বারা সেট/রিসেট করা হয়।

IE1/IE0 – এটি বাহ্যিক ইন্টারাপ্টের Edge flag bit, যখন $\overline{INT_x}$ এ একটি falling edge শনাক্ত হয়, তখন হার্ডওয়্যার দ্বারা এটি সেট হয় এবং সফটওয়্যার দ্বারা রিসেট হয়।

IT1/IT0– এটি বাহ্যিক ইন্টারাপ্ট type flag। এটি falling edge/low level এ সক্রিয় বাহ্যিক ইন্টারাপ্ট এর জন্য সফটওয়্যার কর্তৃক সেট/রিসেট হয়।

## ৯.৩ টাইমার অপারেশনের মুড আলোচনা

**(Describe the Function of Each Bit of TMOD & TCON Register)**

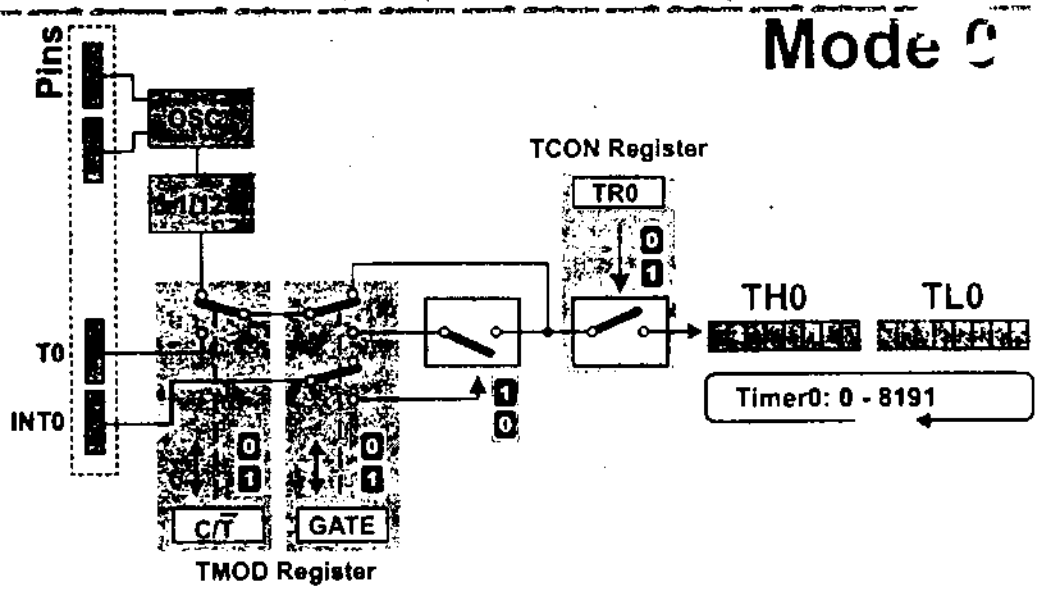
8051 মাইক্রোকন্ট্রোলারের দুটি 16 বিটের টাইমার T0 ও T1 আছে। টাইমার T0-এর সংশ্লিষ্ট দুটি 8 বিটের রেজিস্টার TH0 ও TL0 এবং T1 এর TH0 ও TH1 আছে। উভয় টাইমারই চারটি অপারেটিং মুডে কাজ করে। মুডগুলো হলো :

(i) Timer Mode 0 : 13 bit timer
(ii) Timer Mode 1 : 16-bit timer
(iii) Timer Mode 2 : 8-Bit auto reload
(iv) Timer Mode 3 : Split timer mode.

নিচে মুডগুলোর সংক্ষিপ্ত বিবরণ দেয়া হলো :

**Timer Mode-0**

(1) এটি 13 বিটের টাইমার।
(2) এর 13 বিটের উপরের 8 বিট $TH_x$ রেজিস্টার এবং নিচের 5 বিট $TL_x$ রেজিস্টার ধারণ করে। (subscript x, 0 বা 1 নির্দেশ করে।)
(3) এটি উভয় টাইমারের ক্ষেত্রে প্রযোজ্য।
(4) TCON register এর TR বিট সেট করে এটি চালু করা যায়।
(5) এর সর্বোচ্চ গণনার সীমা 0 – 8192।
(6) টাইমার Overflow হলে TF flag bit উৎপন্ন হয়।
(7) এটি কখনও কখনও ব্যবহৃত হয়।




চিত্র-৯.১ : Timer mode 0

**Timer Mode-1**

(1) এটি 16 বিটের টাইমার।

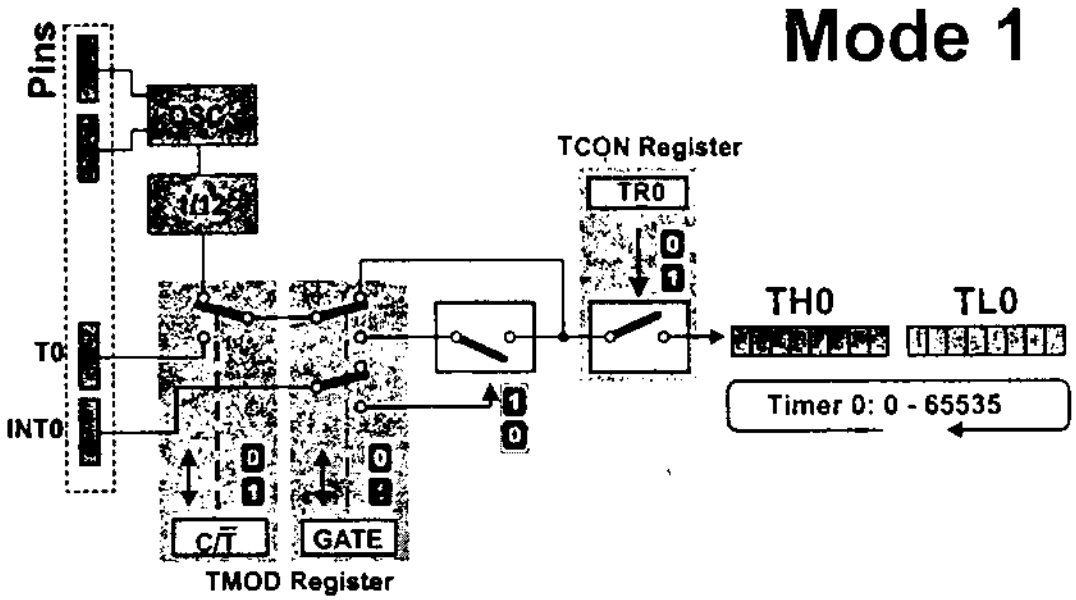
(2) এতে উভয় রেজিস্টারই (16 বিট) ব্যবহৃত হয়।

(3) এটি উভয় টাইমারের ক্ষেত্রে ব্যবহার করা হয়।

(4) TCON register এর TR বিট সেট করে একে চালু করা হয়।

(5) এর সর্বোচ্চ গণনার সীমা 0–8192।

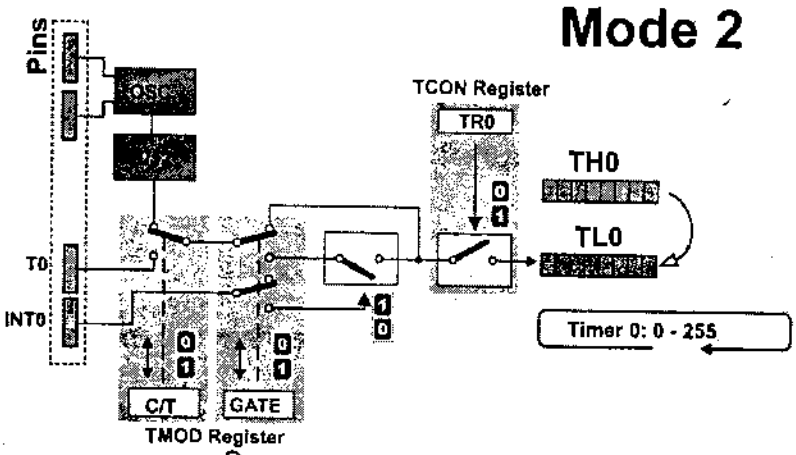
(6) টাইমার Overflow হলে TF flag bit উৎপন্ন হয়।




চিত্র-৯.২ : Timer mode 1

**Timer Mode-2**

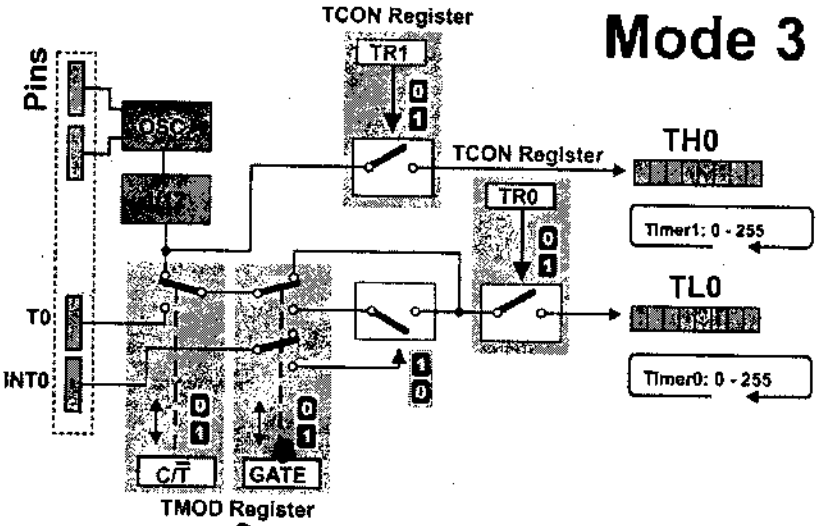
(1) এটি একটি 8 বিটের টাইমার।
(2) এটি একটি auto-reload mode.
(3) $TH_x$ reload value ধারণ করে এবং $TL_x$ হলো টাইমার।
(4) এটি উভয় টাইমারের ক্ষেত্রে প্রযোজ্য।
(5) TCON register এর TR বিট সেট করে এটাকে চালু করা হয়।
(6) এর সর্বোচ্চ গণনার সীমা 0–255
(7) টাইমার overflow হলে TF flag bit উৎপন্ন হয়।
(8) $TH_x$ এর জমা মানই এর রিসেট মান।




চিত্র-৯.৩ : Timer mode 2

**Timer Mode-3**

(1) এটি একটি split-timer mode.
(2) এটি Timer 0 ও Timer 1 এর ক্ষেত্রে ভিন্ন।
(3) যদি Timer 0 কে Timer mode 3 তে সাজানো হয়, তবে এটি 8 বিটের দুটি টাইমারের বিভক্ত হয়।
(4) এতে টাইমার রান বিট হলো TR0 ও TR1.
(5) এর সর্বোচ্চ গণনার সীমা 0–255।




চিত্র-৯.৪ : Timer mode 3

## ৯.৪ টাইমারকে বিভিন্ন মুডে সেটিং এর কোড

### (Write Code for Setting Timer in Different Mode)

টাইমারকে সাধারণত সঠিক অপারেটিং মুডে সেট করার জন্য প্রোগ্রামের শুরুতে সূচিত করতে হয়। টাইমারকে বিভিন্ন মুডে সেট করার জন্য একটি বিশেষ রেজিস্টার TMOD ব্যবহৃত হয়। টাইমার মুড সেট করার জন্য নিচে একটি নির্দেশনা উল্লেখ করা হলো :

MOV TMOD, #00010000B

এই নির্দেশনার সম্পাদনের ফলে TMOD রেজিস্টারের বিটগুলোর অবস্থা নিম্নরূপ হয় :

| Timer 1 | | | | Timer 0 | | | |
|---|---|---|---|---|---|---|---|
| Gate | $C/\bar{T}$ | M1 | M0 | Gate | $C/\bar{T}$ | M1 | M0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

এটি Timer 0 এর মুড বিটগুলোকে ক্লিয়ার করে এবং Timer 1 এর মুড বিটকে সাজায়। এতে M1 = 0, M0 = 1 যা Timer 1 এর mode 1 (16 বিট টাইমার) নির্দেশ করে, $C/\bar{T}$ = 0 যা টাইমার অপারেশন বুঝায় এবং Gate = 0 অর্থাৎ টাইমার অভ্যন্তরীণ ক্লকে পরিচালিত। এই অবস্থায় TCON রেজিস্টারে TR1 বিট সেট করলে টাইমার চলবে।

অনুরূপভাবে Timer 0 কে mode 1 চালিত করার জন্য টাইমার সূচনা কোড নিম্নরূপ :

```
MOV     TMOD    ; #00000001B    ; Timer 0 mode 1
SETB    TR0                     ; Start Timer 0
```

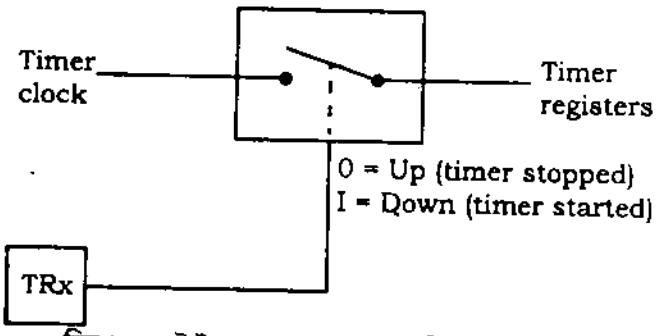
এভাবে অন্যান্য মুডে Timer 0 ও Timer 1 চালিত করার কোড উল্লেখ করা হলো :

```
MOV     TMOD, #00100000b    ; Timer 1 mode 2
MOV     TMOD, #0000010b     ; Timer 1 mode 2
MOV     TMOD, #00110000b    ; Timer 1 mode 2
MOV     TMOD, #00000011b    ; Timer 1 mode 2
MOV     TMOD, #01100000b    ; Timer 1 mode 2
```

## ৯.৫ টাইমার আরম্ভ, বন্ধ ও নিয়ন্ত্রণ প্রক্রিয়া ব্যাখ্যা

### (Explain the Procedure of Starting, Stopping and Controlling Timer)

TCON রেজিস্টারের Run control bit, $TR_x$ দ্বারা টাইমারকে আরম্ভ বা বন্ধ করার সরলতম পদ্ধতি। সিস্টেম রিসেটে $TR_x$ ও রিসেট হয় এবং টাইমারও অকার্যকর বা বন্ধ হয় (নিচের চিত্র)। টাইমারকে চালু করতে সফটওয়্যারের সাহায্যে $TR_x$-কে সেট করতে হয়।




চিত্র-৯.৫ : টাইমার আরম্ভ ও বন্ধের লজিক সার্কিট

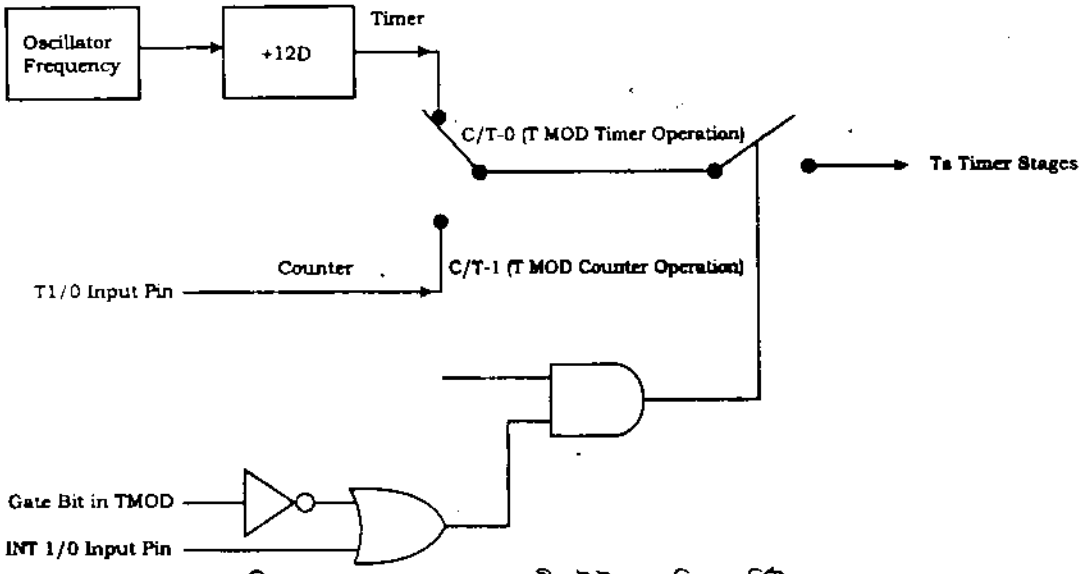
যেহেতু TCON একটি bit addressable রেজিস্টার, তাই $TR_x$ সেট/রিসেট করে প্রোগ্রামে টাইমারকে সহজেই Start/stop করা যায়। যেমন– Timer 0 কে start করার জন্য কোড :

```
SETB    TR0
```

এবং Timer 0 কে stop করার জন্য কোড :

```
CLRB    TR0
```

টাইমারকে নিয়ন্ত্রন করতে হয় TMOD রেজিস্টারের Gate বিট এবং বাহ্যিক ইনপুট $\overline{INT_x}$ সিগন্যাল ব্যবহার করে। Gate কে High করে $INT_x$ সিগন্যাল দ্বারা টাইমারকে নিয়ন্ত্রণ করা হয়। এটি পালসের প্রশস্ততা নির্ণয়ের জন্য খুবই উপযোগী। মনে করি, $\overline{INT0}$ এর অবস্থা Low কিন্তু পরিমাপকৃত সময়ের জন্য পালস এর অবস্থা High। যদি Timer 0 কে mode 2 (16 বিট) তে চালনা করা হয়, তবে TL0/TH0 = 0000H, Gate = 1 এবং TR0 = 1। যখন $\overline{INT0}$ এর অবস্থা High হয়, তখন টাইমারের Gate ON হয় এবং ক্লকের হার 1 MHz হয়। আবার যখন $\overline{INT0}$ এর অবস্থা Low হয়, তখন টাইমারের Gate OFF হয় এবং TL0/TH0 এর গণনাই হলো মাইক্রোসেকেন্ডে পালসের সময়কাল।




চিত্র-৯.৬ : 8051 এর অভ্যন্তরীণ টাইমার লজিক সার্কিট

## ৯.৬ একটি টাইম ডিলে তৈরির জন্য টাইমারের প্রাথমিক মানের হিসাব (Calculate the Initial Value of Timer for Creating a Certain Delay)

একটি নির্দিষ্ট মানের টাইম ডিলে তৈরি করার জন্য টাইমারের প্রাথমিক মান নির্ণয় করা প্রয়োজন। উদাহরণস্বরূপ ধরা যাক সিস্টেম অসিলেটর ফ্রিকুয়েন্সি 11.0592 মেগাহার্টস এবং 5 মিলিসেকেন্ড টাইম ডিলে তৈরির জন্য Timer 0 এর প্রাথমিক মানের হিসাব নিচে দেখানো হলো :

আমরা জানি, 8051-এ একটি একক মেশিন সাইকেল 12টি ক্রিস্টাল পালস নিয়ে গঠিত। যেহেতু ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মেগাহার্টস, সুতরাং টাইমারের ক্লক সাইকেল সময়কাল

$$= \frac{12}{11.0592}$$

= 1.085 মাইক্রোসেকেন্ড

টাইম ডিলে = ডিলে মান × টাইমারের ক্লক সাইকেল সময়কাল

$$\text{ডিলে টাইম} = \frac{\text{টাইম ডিলে}}{\text{টাইমারের ক্লক সাইকেল সময়কাল}}$$

$$= \frac{5 \text{ মিলিসেকেন্ড}}{1.085 \text{ মাইক্রোসেকেন্ড}}$$

= 4608 ক্লক

অর্থাৎ টাইমার roll over করার পূর্বে 4608টি ক্লক গণনা করে। যদি টাইমারকে mode 1 এ অপারেট করা হয়, তবে এর সর্বোচ্চ কাউন্ট হয় 65536।

সুতরাং টাইমারের Reload মান = টাইমারের সর্বোচ্চ গণনা – ডিলে মান

= 65536 – 4608

= 0 × EE00H

সুতরাং TH0 ও TL0 রেজিস্টারের প্রাথমিক মান যথাক্রমে

TH = 0 × EE H এবং

TL0 = 0 × 00H

## ৯.৭ টাইমার ব্যবহার করে 8051 প্রোগ্রাম

**(Write Subroutine for Creating Delay of Certain Amount of Time Using Timer)**

**প্রোগ্রাম- ১ : টাইমার ব্যবহার করে টাইম ডিলে তৈরির সাবরুটিন লেখ।**

**সমাধান :**

নিচে Timer 0 ব্যবহার করে 100 মাইক্রোসেকেন্ড টাইম ডিলে তৈরির একটি সাবরুটিন প্রোগ্রাম উল্লেখ করা হলো :

```
: ..................... Delay using Timer 0
DELAY :   MOV TMOD,#01H     ; initialize TMOD, Timer 0, mode 1
          OV TL0,#47H       ; initialize TL0
          OV TL0,#FFH       ; initialize TH0
          SETB TR0          ; start timer
WAIT :    JNB TF0, Wait     ; wait for TF0
          CLR TR0           ; stop timer
          CLR TF0           ; clear TF0
          RET
```

## ৯.৮ জেনারেটিং স্কয়ার ওয়েভ ও PWM এর জন্য প্রোগ্রাম উন্নয়ন

**(Develop Program for Generating Square Wave and PWM)**

**প্রোগ্রাম- ১ : টাইমার ব্যবহার করে স্কয়ার ওয়েভ তৈরির 8051 প্রোগ্রাম লেখ।**

**সমাধান :**

নিচে P1.0 পিনে 10 কিলোহার্টসের একটি স্কয়ার ওয়েভ তৈরির প্রোগ্রাম উল্লেখ করা হলো :

```
        MOV   TMOD   ,#02h    ; 8 bit auto-reload mode
        MOV   TH0    ,# – 50  ; – 50 reload value in TH0
        SETB  TR0             ; start timer
LOOP :  JNB   TF0,   LOOP     ; wait for overflow
        CLR   TF0             ; clear timer overflow flag
        CPL   P1.0            ; toggle port bit
        SJMP  Loop            ; repeat
        END
```

**প্রোগ্রাম- ২ : টাইমার ব্যবহার করে PWM উৎপন্ন করার 8051 প্রোগ্রাম লেখ।**

**সমাধান :**

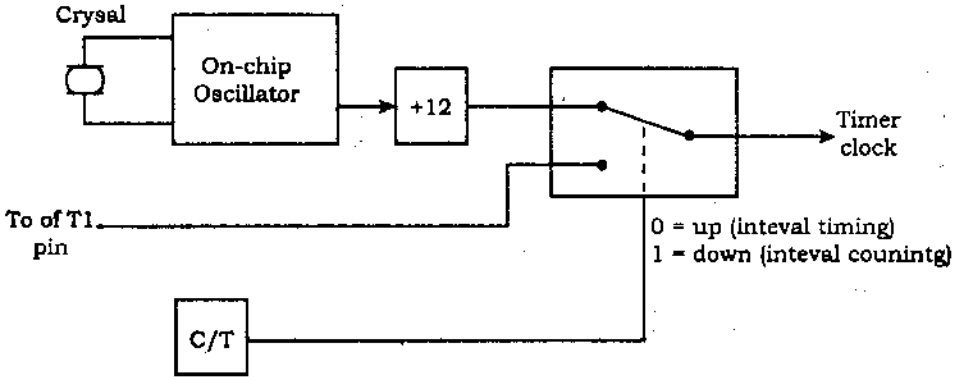
```
; Timer setup for PWM
CODE :
                PWMPIN EQU P1.0      ; PWM output pin
PWM_SETUP :
                MOV TMOD,# OOH       ; Timer 0 in Mode 0
                                     ; Set pulse width control the value
                                       loaded in R7 is value X as; discussed
                                       above.
                SETB EA              ; Enable Interrupts
                SETB ET0             ; Enable Timer 0 Interrupt
                SETB TR0             ; Start TimerRET â

; Interrupt Service Routine
CODE :
            TIMER_0_INTERRUT
            JB F0, HIGH_DONE         ; If F0 flag is set then we just finished the
                                       high section of the
                                     ; cycle so Jump to HIGH_DONESETB F0
                                       Make F0 = 1 to indicate start of high
                                       section
LOW_DONE :
            SETB PWMPIN              ; Make PWM output pin HIgh
            MOV TH0, R7              ; Load high byte of timer with R7 (pulse
            CLR TF0                    width control value)
            RETI                     ; Clear the Timer 0 interrupt flag
                                     ; Return from Interrupt to where the
                                       program came from
HIGH_DONE :
            CLR F0                   ; Make F0 = 0 to indicate start of low
            CLR PWMPIN                 section
            MOV A,#0FFH              ; Make PWM output pin low
            CLR C                    ; Move FFH (255) to A
            SUBB A, R7               ; Clear (the carry bit) so it does not affect
            MOV TH0, A                 the subtraction
            CLR TF0                  ; Subtract R7 from A.a = 255 – R7
            RETI                     ; So the value loaded into TH0 + R7 = 255
                                     ; Clear the Timer 0 interrupt flag
                                     ; Return from Interrupt to where the
                                       program came from.
```

## ৯.৯ ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার

### (Describe the Timer as an Event Counter)

টাইমার শুধুমাত্র সময় পরিমাপ নয় বরং এটি ঘটনা গণনায়ও ব্যবহৃত হয়। যদি Clock source সার্কিট (নিচের চিত্রে) $C/\overline{T} = 1$ হয়, তবে টাইমার ক্লক বাহ্যিক উৎস থেকে হয়। অধিকাংশ ব্যবহারিক ক্ষেত্রে এই বাহ্যিক উৎস কোন ঘটনা সংঘটনে একটি পালস টাইমারকে সরবরাহ করে। এই অবস্থায় টাইমারকে ইভেন্ট কাউন্টার বলে। যেহেতু টাইমার রেজিস্টার প্রত্যেক ঘটনায় ক্রমবৃদ্ধি হয়, তাই এর $TL_x$ ও $TH_x$ এর মান পড়ে সফটওয়্যারে ইভেন্টের সংখ্যা গণনা করা হয়। বাহ্যিক ক্লক উৎস পোর্ট P3 এর বিকল্প ফাংশন থেকে পাওয়া যায়। P3.4 বিটটি টাইমার T0 এর জন্য বাহ্যিক ক্লক ইনপুট এবং P3.5 বিট টাইমার T1 এর বাহ্যিক ক্লক ইনপুট হিসেবে ব্যবহৃত হয়। কাউন্টার প্রয়োগে বাহ্যিক ইনপুট (Tx) এর 1 থেকে 0 পরিবর্তনে (Transition) এ টাইমার রেজিস্টারের ক্রমবৃদ্ধি হয়। অর্থাৎ যখন বাহ্যিক ইনপুট এক সাইকেলে একটি High এবং পরের সাইকেলে Low প্রদর্শন করে, তখন টাইমার রেজিস্টারের ক্রমবৃদ্ধি ঘটে। এই ঘটনার জন্য দুই মেশিন সাইকেল প্রয়োজন, তাই 1 থেকে 0 পরিবর্তনের জন্য 2 মাইক্রোসেকেন্ড সময় লাগে। অপারেশন ফ্রিকুয়েন্সি 12 মে. হার্টস হলে বাহ্যিক ক্লক ফ্রিকুয়েন্সি সর্বোচ্চ 500 কিলোহার্টস।




চিত্র-৯.৭ : ক্লক উৎস

**প্রোগ্রাম- ১ : ইভেন্ট কাউন্টার হিসেবে টাইমার ব্যবহার করে ৮০৫১ প্রোগ্রাম লেখ।**

**সমাধান :**

```
MOV     TMOD,   #01100000B              ; COUNTER1, MODE 2, C/T = 1
                                        ; EXTERNAL PULSE
                MOV     TH1, #0         ; CLEAR TH1
                SETB    P3.5            ; MAKE T1 INPUT
AGAIN :         SETB    TR1             ; START THE COUNTER
BACK :  MOV     A, TL1                  ; GET COPY OF COUNT TL1
                MOV     P2, A           ; DISPLAY IT ON PROT 2
                JNB     TF1, BACK       ; KEEP DOING IT IF TF = 0
                CLR     TR1             ; STOP THE COUNTEWR
                CLR     TF1             ; MAKE TF = 0
                SJMP    AGAIN           ; KEEP DOING IT
```

# অনুশীলনী-০৯

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **DB ডাইরেক্টিভের ব্যবহার বর্ণনা কর।** *[বাকাশিবো : '১৫]*

**উত্তর :** DB ডাইরেক্টিভের একটি 8 বিট ডাটা সংজ্ঞায়িত (Difine) করার জন্য ব্যবহার করা হয়।

২। **ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার লেখ।** *[বাকাশিবো : '১৪]*

**উত্তর :** টাইমার রেজিস্টার প্রত্যেক ঘটনায় ক্রমবৃদ্ধি হয়, তাই এর $TL_x$ ও $TH_x$ এর মান পড়ে সফটওয়্যারে ইভেন্টের সংখ্যা গণনা করা হয়।

৩। **8051 মাইক্রোকন্ট্রোলারে কয়টি অভ্যন্তরীণ টাইমার আছে?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারে দুইটি অভ্যন্তরীণ টাইমার আছে। যথা : T0 ও T1

৪। **8051 এর সহযোগী টাইমার রেজিস্টারগুলোর নাম লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের টাইমার/কাউন্টার অপারেশনে সহযোগী রেজিস্টার হিসেবে নিচের রেজিস্টারসমূহ ব্যবহার হয় :

| | | | |
|---|---|---|---|
| 1. | TCON | (8 BIT) | টাইমার নিয়ন্ত্রণ রেজিস্টার। |
| 2. | TMOD | (8 BIT) | টাইমার মুড নির্ধারণ রেজিস্টার। |
| 3. | TL0 | (8 BIT) | TIMER 0 এর Lower byte ধারণ করে। |
| 4. | TH0 | (8 BIT) | TIMER 0 এর Upper byte ধারণ করে। |
| 5. | TL1 | (8 BIT) | TIMER 1 এর Lower byte ধারণ করে। |
| 6. | TH1 | (8 BIT) | TIMER 1 এর Upper byte ধারণ করে। |

৫। **TMOD রেজিস্টারের কাজ কী?**

**উত্তর :** TMOD register টাইমার/কাউন্টার (0/1) এর অপারেটিং মুড নির্ধারণ করে।

৬। **TCON রেজিস্টারের কাজ কী?**

**উত্তর :** TMOD রেজিস্টার দ্বারা নির্ধারিত টাইমার বা কাউন্টারকে পরিচালনার জন্য এটি প্রয়োজনীয় কন্ট্রোল বিট ধারণ করে এবং বিভিন্ন ফ্ল্যাগ বিটগুলো (status flag) উৎপন্ন করে।

৭। **TMOD register এ Gate সিগন্যালের কাজ কী?**

**উত্তর :** $Gate_x$ সিগন্যাল $Timer_x$ কে $INT_x$ এর সিগন্যালের মাধ্যমে কার্যকর বা অকার্যকর করা হয়। এর মান

1 = $INT_x$ বিট সেট হলে $Timer_x$ কার্যকর হয়।

0 = $Timer_x$ কার্যকর হওয়ার জন্য $INT_x$ বিটের স্টেটের উপর নির্ভর করে না।

৮। **TMOD register এ $C/\bar{T}$ সিগন্যালের কাজ কী?**

**উত্তর :** TMOD register এ $C/\bar{T}$ সিগন্যালের কাজ হলো টাইমার বা কাউন্টার নির্বাচন করা। এর মান

1 = $Counter_x$

0 = $Timer_x$

**৯। TMOD register এ টাইমার মুড নির্ধারী বিটগুলোর নাম লিখ এবং মুড নির্ধারণে বিটগুলোর সজ্জা দেখাও।**

**উত্তর :** TMOD registewr এ টাইমার মুড নির্ধারণী বিটগুলোর নাম M0, M1। টাইমার মুড নির্বাচনের জন্য নিচে এদের সমন্বয় দেখানো হলো :

| $T_xM_1$ | $T_xM_0$ | Timer Mode |
|---|---|---|
| 0 | 0 | Timer Mode 0 : 13-bit timer |
| 0 | 1 | Timer Mode 1 : 16-bit timer |
| 1 | 0 | Timer Mode 2 : 8-Bit auto-reload |
| 1 | 1 | Timer Mode 3 : Split timer mode |

**১০। TCON রেজিস্টারের TF বিটের কাজ কী?**

**উত্তর :** এটি একটি টাইমার overflow নির্দেশক। টাইমার overflow হলে এটি হার্ডওয়্যার দ্বারা সেট হয় এবং হার্ডওয়্যার বা সফটওয়্যার দ্বারা রিসেট হয়।

**১১। TCON রেজিস্টারের TR বিটের কাজ কী?**

**উত্তর :** এটি টাইমার চালনার জন্য নিয়ন্ত্রণ বিট। এটি সেট হলে টাইমার অন হয় এবং রিসেট হলে টাইমার অফ হয়। এটি সফটওয়্যার দ্বারা সেট/রিসেট হয়।

**১২। 8051 এর টাইমার মুড কয়টি ও কী কী?**

**উত্তর :** 8051 এর মোট 4টি টাইমার মুড আছে, এগুলো হলো :

1. Timer Mode 0 : 13-bit timer
2. Timer Mode 1 : 16-bit timer
3. Timer Mode 2 : 8-Bit auto reload
4. Timer Mode 3 : Split timer mode.

**১৩। Timer 0 কে টাইমার হিসেবে mode 1 সেটিং এর জন্য TMOD রেজিস্টার সাজানোর কোড লিখ।**

**উত্তর :** MOV TMOD, #00000001B

**১৪। Timer 0 কে চালু ও বন্ধ করার নির্দেশনা লিখ।**

**উত্তর :**
```
SETB TR0        ; start timer 0
CLEAR T0        ; stop timer 0
```

**১৫। ইভেন্ট কাউন্টার বলতে কী বোঝায়?**

**উত্তর :** অধিকাংশ ব্যবহারিক ক্ষেত্রে এই বাহ্যিক উৎস কোন ঘটনা সংঘটনে একটি পালস টাইমারকে সরবরাহ করে। এই অবস্থায় টাইমারকে ইভেন্ট কাউন্টার বলে।

## সংক্ষিপ্ত প্রশ্নোত্তর

**১। টাইমারের কাজ উল্লেখ কর।** *[বাকাশিবো : '১৪, '১৪R]*

**উত্তর :** 8051 টাইমারের সাধারণ কাজ নিচে দেওয়া হলো :

(১) ফ্রিকুয়েন্সি পরিমাপ (Frequency measurements)
(২) পালসের প্রশস্ততা পরিমাপ (Pulse width measurements)
(৩) ঘটনা গণনা (Event counting)
(৪) সিরিয়াল কমিউনিকেশনের জন্য baud rate উৎপাদন।
(৫) বিরতির সময় নির্ধারণ (Interval timing)
  (ক) পর্যায়বৃত্ত ঘটনার সময় নির্ধারণ (Periodic event timing)
  (খ) পরিমাপের জন্য সময়ের ভিত্তি নির্ধারণ (Time base for measurements)

২। **TMOD রেজিস্টারের বিটগুলো বর্ণনা কর।**

**উত্তর :** TMOD register টাইমার/কাউন্টার (0/1) এর অপারেটিং মুড নির্ধারণ করে। এটি bit-addressable রেজিস্টার নয়। এটির বিট সজ্জা নিম্নরূপ :

(MSB) (LSB)

| Gate | $C/\bar{T}$ | M1 | M0 | Gate | $C/\bar{T}$ | M1 | M0 |
|---|---|---|---|---|---|---|---|

---------- Timer 1 ---------- ---------- Timer 0 ----------

**বিটগুলোর কাজ :**

$Gate_x$ : $Timer_x$ কে $INT_x$ এর সিগন্যালের মাধ্যমে কার্যকর বা অকার্যকর করা হয়।

1 = $INT_x$ বিট সেট হলে $Timer_x$ কার্যকর হয়।

0 = $Timer_x$ কার্যকর হওয়ার জন্য $INT_x$ বিটের স্টেটের উপর নির্ভর করে না।

$C/\bar{T}$ : টাইমার/কাউন্টার নির্বাচন বিট

1 = $Counter_x$

0 = $Timer_x$

M0, M1- টাইমার মুড নির্বাচন বিট। টাইমার মুড নির্বাচনের জন্য নিচে এদের সমন্বয় দেখানো হলো :

| $T_xM_1$ | $T_xM_0$ | Timer Mode |
|---|---|---|
| 0 | 0 | Timer Mode 0 : 13-bit timer |
| 0 | 1 | Timer Mode 1 : 16-bit timer |
| 1 | 0 | Timer Mode 2 : 8-bit auto reload |
| 1 | 1 | Timer Mode 3 : Split timer mode |

বি: দ্র: Subscript x, 0 বা 1 নির্দেশ করে, যা Timer 1 প্রকাশ করে।

৩। **TCON রেজিস্টারের বিটগুলো বর্ণনা কর।**

**উত্তর :** TMOD রেজিস্টার কর্তৃক নির্ধারিত টাইমার বা কাউন্টারকে পরিচালনার জন্য এটি প্রয়োজনীয় কন্ট্রোল বিট ধারণ করে এবং বিভিন্ন ফ্ল্যাগ বিটগুলো (Status flag) তৈরি করে। এটি একটি bit-addressable রেজিস্টার। এর বিট-সজ্জা নিম্নরূপ :

| TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
|---|---|---|---|---|---|---|---|

TF1/TF0 – Timer overflow flag bit. টাইমার overflow হলে এটি হার্ডওয়্যার দ্বারা সেট হয় এবং হার্ডওয়্যার বা সফটওয়্যার দ্বারা রিসেট হয়।

TR1/TR0 – টাইমার চালনার জন্য TRI/TRO নিয়ন্ত্রণ বিট হিসেবে কাজ করে। এটি সেট হলে টাইমার অন হয় এবং রিসেট হলে টাইমার অফ হয়। এটি সফটওয়্যার দ্বারা সেট/রিসেট করা হয়।

IE1/IE0 – এটি বাহ্যিক ইন্টারাপ্টের Edge flag bit, যখন $\overline{INT_x}$ এ একটি falling edge শনাক্ত হয়, তখন হার্ডওয়্যার দ্বারা এটি সেট হয় এবং সফটওয়্যার দ্বারা রিসেট হয়।

IT1/IT0– এটি বাহ্যিক ইন্টারাপ্ট type flag। এটি falling edge/low level এ সক্রিয় বাহ্যিক ইন্টারাপ্ট এর জন্য সফটওয়্যার কর্তৃক সেট/রিসেট হয়।

**৪। টাইমারকে বিভিন্ন মুডে সেটিং করার কোড লিখ।**

**উত্তর :** টাইমারকে সাধারণত সঠিক অপারেটিং মুডে সেট করার জন্য প্রোগ্রামের শুরুতে সূচিত করতে হয়। টাইমারকে বিভিন্ন মুডে সেট করার জন্য একটি বিশেষ রেজিস্টার TMOD ব্যবহৃত হয়। টাইমার মুড সেট করার জন্য নিচে একটি নির্দেশনা উল্লেখ করা হলো :

MOV TMOD, #00010000B

এই নির্দেশনার সম্পাদনের ফলে TMOD রেজিস্টারের বিটগুলোর অবস্থা নিম্নরূপ হবে :

| Timer 1 | | | | Timer 0 | | | |
|---|---|---|---|---|---|---|---|
| Gate | $C/\bar{T}$ | M1 | M0 | Gate | $C/\bar{T}$ | M1 | M0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

এটি Timer 0 এর মুড বিটগুলোকে ক্লিয়ার করে এবং Timer 1 এর মুড বিটকে সাজায়। এতে M1 = 0, M0 = 1 যা Timer 1 এর mode 1 (16 বিট টাইমার) নির্দেশ করে, $C/\bar{T}$ = 0 যা টাইমার অপারেশন বুঝায় এবং Gate = 0 অর্থাৎ টাইমার অভ্যন্তরীণ ক্লকে পরিচালিত। এই অবস্থায় TCON রেজিস্টারে TR1 বিট সেট করলে টাইমার চলবে।

অনুরূপভাবে Timer 0 কে mode 1 চালিত করার জন্য টাইমার সূচনা কোড নিম্নরূপ :

```
MOV     TMOD     ; #00000001B     ; Timer 0 mode 1
SETB    TR0                       ; Start Timer 0
```

এভাবে অন্যান্য মুডে Timer 0 ও Timer 1 চালিত করার কোড উল্লেখ করা হলো :

```
MOV     TMOD, #00100000b     ; Timer 1 mode 2
MOV     TMOD, #0000010b      ; Timer 1 mode 2
MOV     TMOD, #00110000b     ; Timer 1 mode 2
MOV     TMOD, #00000011b     ; Timer 1 mode 2
MOV     TMOD, #01100000b     ; Timer 1 mode 2
```

**৫। একটি টাইম ডিলে তৈরির জন্য টাইমারের প্রাথমিক মান নির্ণয় কর।**

**উত্তর :** একটি নির্দিষ্ট মানের টাইম ডিলে তৈরি করার জন্য টাইমারের প্রাথমিক মান নির্ণয় করা প্রয়োজন। উদাহরণস্বরূপ ধরা যাক সিস্টেম অসিলেটর ফ্রিকুয়েন্সি 11.0592 মেগাহার্টস এবং 5 মিলিসেকেন্ড টাইম ডিলে তৈরির জন্য Timer 0 এর প্রাথমিক মানের হিসাব নিচে দেখানো হলো :

আমরা জানি, 8051-এ একটি একক মেশিন সাইকেল 12টি ক্রিস্টাল পালস নিয়ে গঠিত। যেহেতু ক্রিস্টাল ফ্রিকুয়েন্সি 11.0592 মেগাহার্টস, সুতরাং টাইমারের ক্লক সাইকেল সময়কাল

$$= \frac{12}{11.0592}$$

= 1.085 মাইক্রোসেকেন্ড

টাইম ডিলে = ডিলে মান × টাইমারের ক্লক সাইকেল সময়কাল

$$\text{ডিলে টাইম} = \frac{\text{টাইম ডিলে}}{\text{টাইমারের ক্লক সাইকেল সময়কাল}}$$

$$= \frac{\text{5 মিলিসেকেন্ড}}{\text{1.085 মাইক্রোসেকেন্ড}}$$

= 4608 ক্লক

অর্থাৎ টাইমার roll over করার পূর্বে 4608টি ক্লক গণনা করে। যদি টাইমারকে mode 1 এ অপারেট করা হয়, তবে এর সর্বোচ্চ কাউন্ট হয় 65536।

সুতরাং টাইমারের Reload মান = টাইমারের সর্বোচ্চ গণনা – ডিলে মান
= 65536 – 4608
= 0 – EE00H

সুতরাং TH0 ও TL0 রেজিস্টারের প্রাথমিক মান যথাক্রমে

TH = 0 × EE H এবং

TL0 = 0 × 00H

৬। **টাইমার ব্যবহার করে টাইম ডিলে তৈরির সাবরুটিন প্রোগ্রাম লিখ।**

**উত্তর :** নিচে Timer 0 ব্যবহার করে 100 মাইক্রোসেকেন্ড টাইম ডিলে তৈরির একটি সাবরুটিন প্রোগ্রাম উল্লেখ করা হলো :

```
: ..................... Delay using Timer 0
DELAY :   MOV TMOD,#01H     ; initialize TMOD, Timer 0, mode 1
          OV TL0,#47H       ; initialize TL0
          OV TL0,#FFH       ; initialize TH0
          SETB TR0          ; start timer
WAIT :    JNB TF0, Wait     ; wait for TF0
          CLR TR0           ; stop timer
          CLR TF0           ; clear TF0
          RET
```

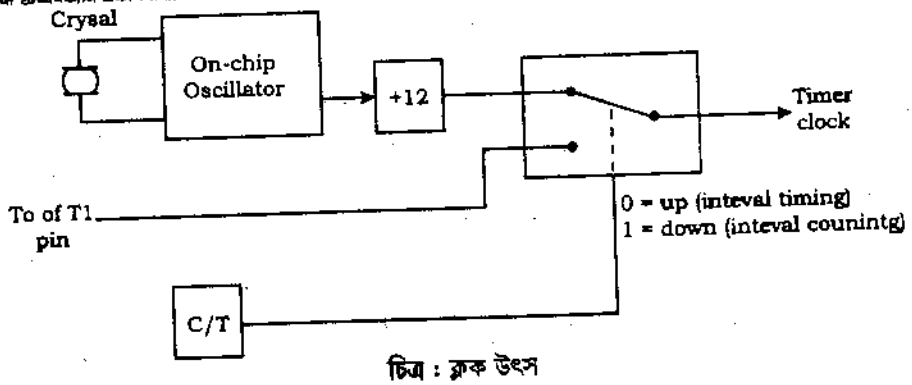
৭। **টাইমার ব্যবহার করে স্কয়ার ওয়েভ তৈরির 8051 প্রোগ্রাম লিখ।**

**উত্তর :** নিচে P1.0 পিনে 10 কিলোহার্টসের একটি স্কয়ার ওয়েভ তৈরির প্রোগ্রাম উল্লেখ করা হলো :

```
        MOV     TMOD    #02h    ; 8 bit auto-reload mode
        MOV     TH0     # – 50  ; – 50 reload value in TH0
        SETB    TR0             ; start timer
LOOP :  JNB     TF0,    LOOP    ; wait for overflow
        CLR     TF0             ; clear timer overflow flag
        CPL     P1.0            ; toggle port bit
        SJMP    Loop            ; repeat
        END
```

৮। **ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার বর্ণনা কর।**

**উত্তর :** টাইমার শুধুমাত্র সময় পরিমাপ নয় বরং এটি ঘটনা গণনায়ও ব্যবহৃত হয়। যদি Clock source সার্কিট এ $C/\overline{T} = 1$ হয়, তবে টাইমার ক্লক বাহ্যিক উৎস থেকে হয়। অধিকাংশ ব্যবহারিক ক্ষেত্রে এই বাহ্যিক উৎস কোন ঘটনা সংঘটনে একটি পালস টাইমারকে সরবরাহ করে। এই অবস্থায় টাইমারকে ইভেন্ট কাউন্টার বলে। যেহেতু টাইমার রেজিস্টার প্রত্যেক ঘটনায় ক্রমবৃদ্ধি হয়, তাই এর TLx ও THx এর মান পড়ে সফটওয়্যারে ইভেন্টের সংখ্যা গণনা করা হয়। বাহ্যিক ক্লক উৎস পোর্ট P3 এর বিকল্প ফাংশন থেকে পাওয়া যায়। P3.4 বিটটি টাইমার T0 এর জন্য বাহ্যিক ক্লক ইনপুট এবং P3.5 বিট টাইমার T1 এর বাহ্যিক ক্লক ইনপুট হিসেবে ব্যবহৃত হয়। কাউন্টার প্রয়োগে বাহ্যিক ইনপুট (Tx) এর 1 থেকে 0 পরিবর্তনে (Transition) এ টাইমার রেজিস্টারের ক্রমবৃদ্ধি হয়। অর্থাৎ যখন বাহ্যিক ইনপুট এক সাইকেলে একটি High এবং পরের সাইকেলে Low প্রদর্শন করে, তখন টাইমার রেজিস্টারের ক্রমবৃদ্ধি ঘটে। এই ঘটনার জন্য দুই মেশিন সাইকেল প্রয়োজন, তাই 1 থেকে 0 পরিবর্তনের জন্য 2 মাইক্রোসেকেন্ড সময় লাগে। অপারেশন ফ্রিকুয়েন্সি 12 মে. হার্টস হলে বাহ্যিক ক্লক ফ্রিকুয়েন্সি সর্বোচ্চ 500 কিলোহার্টস।




চিত্র : ক্লক উৎস

**৯। টাইমার ব্যবহার করে টাইম ডিলে তৈরির সাবরুটিন লেখ।**

**উত্তর :** নিচে Timer 0 ব্যবহার করে 100 মাইক্রোসেকেন্ড টাইম ডিলে তৈরির একটি সাবরুটিন প্রোগ্রাম উল্লেখ করা হলো :

```
: ..................... Delay using Timer 0
DELAY :   MOV TMOD,#01H     ; initialize TMOD, Timer 0, mode 1
          OV TL0,#47H       ; initialize TL0
          OV TL0,#FFH       ; initialize TH0
          SETB TR0          ; start timer
WAIT :    JNB TF0, Wait     ; wait for TF0
          CLR TR0           ; stop timer
          CLR TF0           ; clear TF0
          RET
```

**১০। টাইমার ব্যবহার করে স্কয়ার ওয়েভ তৈরির 8051 প্রোগ্রাম লেখ।**

**উত্তর :** নিচে P1.0 পিনে 10 কিলোহার্টসের একটি স্কয়ার ওয়েভ তৈরির প্রোগ্রাম উল্লেখ করা হলো :

```
          MOV     TMOD    ,#02h     ; 8 bit auto-reload mode
          MOV     TH0     ,# – 50   ; – 50 reload value in TH0
          SETB    TR0               ; start timer
LOOP :    JNB     TF0,    LOOP      ; wait for overflow
          CLR     TF0               ; clear timer overflow flag
          CPL     P1.0              ; toggle port bit
          SJMP    Loop              ; repeat
          END
```

**১১। টাইমার ব্যবহার করে PWM উৎপন্ন করার 8051 প্রোগ্রাম লেখ।**

**উত্তর :**

```
; Timer setup for PWM
CODE :
                PWMPIN EQU P1.0     ; PWM output pin
PWM_SETUP :
                MOV TMOD,# OOH      ; Timer 0 in Mode 0
                                    ; Set pulse width control the valu
                                    loaded in R7 is value X as; discuss
                                    above.
```

```
                SETB EA          ; Enable Interrupts
                SETB ET0         ; Enable Timer 0 Interrupt
                SETB TR0         ; Start TimerRET â
; Interrupt Service Routine
CODE :
                TIMER_0_INTERRUT
                JB F0, HIGH_DONE ; If F0 flag is set then we just finished the
                                 high section of the
                                 ; cycle so Jump to HIGH_DONESETB F0
                                 Make F0 = 1 to indicate start of high
                                 section
LOW_DONE :
                SETB PWMPIN      ; Make PWM output pin HIgh
                MOV TH0, R7      ; Load high byte of timer with R7 (pulse
                                 width control value)
                CLR TF0          ; Clear the Timer 0 interrupt flag
                RETI             ; Return from Interrupt to where the
                                 program came from
HIGH_DONE :
                CLR F0           ; Make F0 = 0 to indicate start of low
                                 section
                CLR PWMPIN       ; Make PWM output pin low
                MOV A,#0FFH      ; Move FFH (255) to A
                CLR C            ; Clear (the carry bit) so it does not affect
                                 the subtraction
                SUBB A, R7       ; Subtract R7 from A.a = 255 – R7
                MOV TH0, A       ; So the value loaded into TH0 + R7 = 255
                CLR TF0          ; Clear the Timer 0 interrupt flag
                RETI             ; Return from Interrupt to where the
                                 program came from.
```

১২। ইভেন্ট কাউন্টার হিসেবে টাইমার ব্যবহার করে ৮০৫১ প্রোগ্রাম লেখ।

উত্তর :

```
MOV     TMOD,   #01100000B              ; COUNTER1, MODE 2, C/T = 1
                                        ; EXTERNAL PULSE
                MOV     TH1, #0         ; CLEAR TH1
                SETB    P3.5            ; MAKE T1 INPUT
AGAIN :  SETB   TR1                     ; START THE COUNTER
BACK :   MOV    A, TL1                  ; GET COPY OF COUNT TL1
                MOV     P2, A           ; DISPLAY IT ON PROT 2
                JNB     TF1, BACK       ; KEEP DOING IT IF TF = 0
                CLR     TR1             ; STOP THE COUNTEWR
                CLR     TF1             ; MAKE TF = 0
                SJMP    AGAIN           ; KEEP DOING IT
```

## রচনামূলক প্রশ্নোত্তর

[বাকাশিবো : '১৫]

১। 8051 এর টাইমার অপারেশনের মুডগুলো বর্ণনা কর।

উত্তর : ৯.৩ নং অনুচ্ছেদ দ্রষ্টব্য।

২। 8051 এর TMOD ও TCON রেজিস্টারের বিটগুলোর কাজ লিখ।

উত্তর : ৯.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৩। 8051 টাইমারের অভ্যন্তরীণ লজিক সার্কিট অংকন করে টাইমার আরম্ভ, বন্ধ ও নিয়ন্ত্রণ পদ্ধতি বর্ণনা কর।

উত্তর : ৯.৫ নং অনুচ্ছেদ দ্রষ্টব্য।

৪। টাইমার ব্যবহার করে PWM তৈরির 8051 প্রোগ্রাম লিখ।

উত্তর : ৯.৮ নং অনুচ্ছেদ দ্রষ্টব্য।

৫। প্রোগ্রামসহ ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার বর্ণনা কর।

উত্তর : ৯.৯ নং অনুচ্ছেদ দ্রষ্টব্য।

৬। টাইমারের কাজের তালিকা লিখ।

উত্তর : ৯.১ নং অনুচ্ছেদ দ্রষ্টব্য।

৭। TCON Register লিখ।

উত্তর : ৯.২.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৮। টাইমারকে বিভিন্ন মুডে সেটিং এর কোড বর্ণনা কর।

উত্তর : ৯.৪ নং অনুচ্ছেদ দ্রষ্টব্য।

৯। একটি টাইম ডিলে তৈরির জন্য টাইমারের প্রাথমিক মানের হিসাব বর্ণনা কর।

উত্তর : ৯.৬ নং অনুচ্ছেদ দ্রষ্টব্য।

# অধ্যায় ১০ 8051 সিরিয়াল কমিউনিকেশন
# Understand 8051 Serial Communication

## ১০.০ ভূমিকা
## (Introduction)

**কমিউনিকেশন নেটওয়ার্ক (Communication Network) :** যে নেটওয়ার্কের মাধ্যমে সংবাদ, তথ্য বা ম্যাসেজ আদান-প্রদান করা হয় তাকে কমিউনিকেশন নেটওয়ার্ক বা যোগাযোগ নেটওয়ার্ক বলে।

অর্থাৎ, বিভিন্ন প্রকার তথ্যকে যে সকল মাধ্যমের সাহায্যে এক স্থান থেকে অন্য স্থানে পাঠানো বা স্থানান্তর করা হয় তাকে কমিউনিকেশন নেটওয়ার্ক বলে।

বিভিন্ন মাধ্যমের সাহায্যে কমিউনিকেশন নেটওয়ার্ক সম্পাদন করা হয়। কমিউনিকেশন নেটওয়ার্কগুলো হলো- রেডিও, টেলিভিশন, টেলিফোন, টেলিগ্রাফ বা টেলি প্রিন্টার, ভিশন ফোন, ফ্যাক্সিমাইল, রাডার, অপটিক্যাল ফাইবার, স্যাটেলাইট, ইন্টারনেট ইত্যাদি।

বর্তমান সময়ে রেডিও যোগাযোগ ব্যবস্থা অত্যন্ত গুরুত্বপূর্ণ বিষয় হয়ে উঠেছে। কারণ হাজার হাজার মাইল দূরত্বের মধ্যে কোনো একটি স্থান হতে যেকোনো একটি স্থানে যোগাযোগ করা সম্ভব হয়েছে আধুনিক রেডিও যোগাযোগ ব্যবস্থার উন্নয়নের কারণেই। তার বিহীন ইলেকট্রিক্যাল কমিউনিকেশন যেখানে রেডিও ওয়েভকে ট্রান্সমিশনের জন্য ব্যবহার করা হয় তাকেই রেডিও কমিউনিকেশন বলে।

রেডিও ওয়েভ হলো মূলত ইলেকট্রোম্যাগনেটিক ওয়েভ যা শূন্য মাধ্যমের মধ্য দিয়ে ভ্রমণ করতে পারে।

রেডিও ওয়েভের গতি আলোর গতির সমান। এটি ইলেকট্রিক ও ম্যাগনেটিক ফিল্ড নিয়ে গঠিত। ইলেকট্রিক ও ম্যাগনেটিক ফিল্ড দুটি পরস্পর 90° কোণে অবস্থান করে। নিম্নলিখিতভাবে রেডিও কমিউনিকেশনের সংজ্ঞা দেওয়া যায়।

রেডিও কমিউনিকেশন হলো দুই বা ততোধিক অংশের মধ্যে সিম্বল, সিগনাল অথবা ইন্টেলিজেন্সের (Intelligence) আদান-প্রদান যেখানে ট্রান্সমিশনের মাধ্যম হিসেবে ইলেকট্রোম্যাগনেটিক ওয়েভকে ব্যবহার করা হয়।

## ১০.১ সিরিয়াল ও প্যারালাল কমিউনিকেশন
## (Define Serial and Parallel Communication)

প্রতি নিয়ত দুটি ইলেকট্রনিক্স ডিভাইসের মধ্যে ডাটা আদান প্রদান চলে। দুটি ইলেকট্রনিক্স ডিভাইসের মধ্যকার ডাটা আদান-প্রদান (যেমন- কম্পিউটার ও মাইক্রোকন্ট্রোলার বা পেরিফেরাল ডিভাইস) সাধারণত দুইভাবে হয়, যথা-

১। সিরিয়াল ডাটা কমিউনিকেশন (Serial Communication),

২। প্যারালাল ডাটা কমিউনিকেশন (Parallel Communication)।

**সিরিয়াল কমিউনিকেশন (Serial Communication) :** সিরিয়াল কমিউনিকেশনে একটি একক তারের ভিতর দিয়ে একটি সময় অনুক্রমে (Timer sequence) অর্থাৎ একই সময়ে এক বিট আকারে ডাটা সম্প্রচারিত হয়। অর্থাৎ যে কমিউনিকেশনে সিগন্যাল প্রপাগেশন একটি একক পথ (single path) দিয়ে হয় এবং ডাটা বিটগুলো একটির পর একটি স্থানান্তরিত হয়, তাকে সিরিয়াল কমিউনিকেশন বলা হয়। এটা দুই ধরনের (১) synchronous ও (২) Asynchronous হয়। সিরিয়াল কমিউনিকেশন এক বা দুটি ডাটা লাইন ব্যবহার করে এবং সাধারণত দূর পাল্লার কমিউনিকেশনে ব্যবহৃত হয়।

**প্যারালাল কমিউনিকেশন (Parallel Communication) :** সিরিয়াল কমিউনিকেশনের তুলনায় প্যারালাল কমিউনিকেশন ডাটা সম্প্রচার গতি অনেক বেশি। তাই কম্পিউটারে সব ডাটা স্থানান্তর প্যারালাল কমিউনিকেশনের উপর ভিত্তি করে হয়। কিন্তু উচ্চ খরচের কারণে স্বল্প পাল্লার কমিউনিকেশনে এটি সীমাবদ্ধ। তাই সংজ্ঞায় বলা যায়, যে কমিউনিকেশনে একাধিক তার (বাস) একে অপরের সাথে সমান্তরালে অবস্থান করে এবং সব তারে একই সাথে ডাটা স্থানান্তরিত হয়, অর্থাৎ একই সময়ে কোন বাইটের সব বিট স্থানান্তরিত হয়, তাকে প্যারালাল কমিউনিকেশন বলে।

## ১০.২ আরএস 232 ডিবি 9 পিন কানেক্টর আলোচনা

## (Describe the RS232 9-pin D Type Connector)

সিরিয়াল পোর্ট অন্যান্য PC পোর্টের মতো একটি বাস্তব ইন্টারফেস, যা কম্পিউটার ও একটি বাহ্যিক হার্ডওয়্যার বা ডিভাইসের মধ্যে ডাটা স্থানান্তর করে। সিরিয়াল পোর্টের মাধ্যমে ডাটা স্থানান্তর bit by bit হয়। IBM কর্তৃক প্রবর্তিত সিরিয়াল ইনপুট/আউটপুট (DB-9 RS-232) কম্পিউটার ও অন্যান্য ডিভাইসে বহুল ব্যবহৃত ইন্টারফেস। RS-232 এ High ও Low ভোল্টেজ নিচের voltage range দ্বারা প্রকাশ করা হয়।

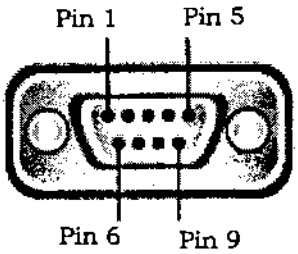
| Bit | Voltage Range (Volts) | |
|---|---|---|
| 0 | + 3 | + 25 |
| 1 | – 25 | – 3 |

– 3V থেকে + 3V range অনুরূপিত। TTL standard আসে RS-232 standard এর অনেক পরে, RS-232 তাই TTL লজিকের সাথে সুসংগত (compatible) নয়। তাই RS-232 কে মাইক্রোকন্ট্রোলারের সাথে সংযুক্তিতে একটি ভোল্টেজ কনভার্টার প্রয়োজন। কনভার্টার মাইক্রোকন্ট্রোলার আউটপুট লেভেলকে RS-232 ভোল্টেজ লেভেলে রূপান্তর করে। ভোল্টেজ ড্রাইভার হিসেবে সাধারণত বহুল ব্যবহৃত হয় IC MAX 232, যা লাইন ড্রাইভার হিসেবে সমাধিক পরিচিত।

কম্পিউটার সিরিয়াল পোর্টের ন্যূনতম তিনটি পিন দরকার, কম্পিউটার ও মাইক্রোকন্ট্রোলারের মাঝে সংযোগ স্থাপনের জন্য যথা : R × D (Receiver, pin2), T × D (Transmitter, pin3) এবং Ground (pin5)। নিচে DB-9 RS-232 এর পিন ডায়াগ্রাম দেখানো হলো :

| | |
|---|---|
| Pin 1 | DCD |
| Pin 2 | RXD |
| Pin 3 | TXD |
| Pin 4 | DTR |
| Pin 5 | GND |
| Pin 6 | DSR |
| Pin 7 | RTS |
| Pin 8 | CTS |
| Pin 9 | RI |

RS232 Pinout (9 Pin Male)




চিত্র-১০.১ : RS-232 এর পিন ডায়াগ্রাম

**পিনের বর্ণনা :**

| PIN | আদ্যাক্ষরা (Acronym) | পূর্ণ নাম | দিক | বর্ণনা |
|---|---|---|---|---|
| 1. | – DCD | Data carrier detect | ← | মডেম সংযুক্ত। |
| 2. | R × D | Receive data | ← | কম্পিউটার বা ডিভাইস ডাটা বাইট গ্রহণ করে। |
| 3. | T × D | Transmit Data | → | কম্পিউটার বা ডিভাইস থেকে ডাটা বাইট বের হয়। |

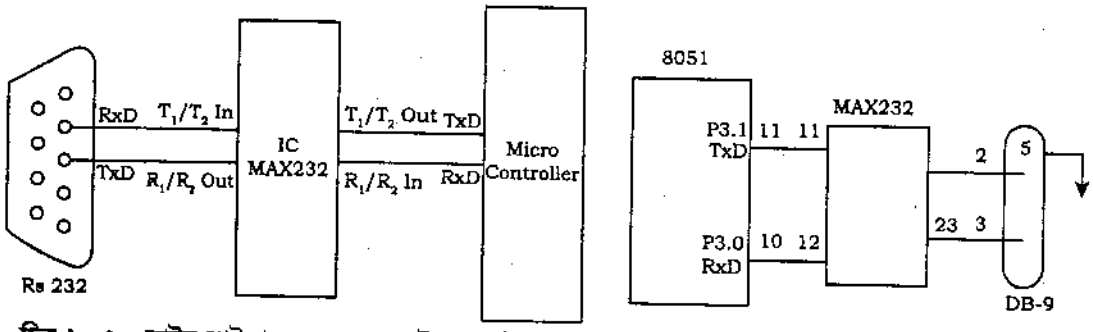
| | | | | |
|---|---|---|---|---|
| 4. | DTR | Data terminal ready | → | এই ডিভাইস যোগাযোগের জন্য প্রস্তুত। |
| 5. | GND | Signal ground | | গ্রাউন্ড। |
| 6. | – DSR | Data set ready | ← | এই ডিভাইস যোগাযোগের জন্য প্রস্তুত। |
| 7. | – RTS | Request to send | → | RTS/CTS প্রবাহ নিয়ন্ত্রণ। |
| 8. | – CTS | Clear to send | ← | RTS/CTS প্রবাহ নিয়ন্ত্রণ। |
| 9. | RI | Ring indicator | ← | টেলিফোন লাইনে রিংগিং। |

## ১০.৩ লাইন ড্রাইভারের পিন ও সংযোগ আলোচনা

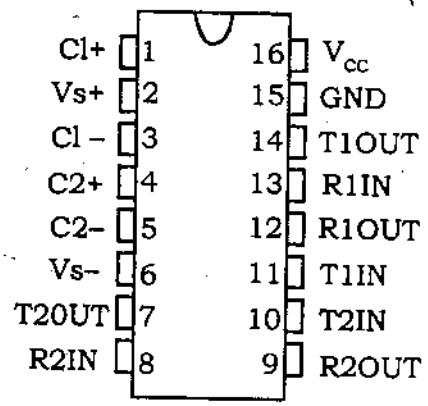
**(Describe the Pins and Connections of Line Driver (Such as MAX232))**

TTL standard ও RS-232 standard এর মধ্যে Compatibility সমস্যা আছে। তাই RS-232 কে মাইক্রোকন্ট্রোলারের সাথে সংযুক্তিতে ভোল্টেজ কনভার্টারে একটি ভোল্টেজ ড্রাইভার দরকার। কনভার্টার মাইক্রোকন্ট্রোলার আউটপুট লেভেলকে RS-232 ভোল্টেজ লেভেলে রূপান্তর করে। ভোল্টেজ ড্রাইভার হিসেবে সাধারণত বেশি ব্যবহৃত হয় IC MAX 232, যা লাইন ড্রাইভার হিসেবে সমাধিক পরিচিত।

সিরিয়াল পোর্টের T × D পিনের সাথে কন্ট্রোলারের R × D পিনের সংযোগ করা হয় MAX232 এর মাধ্যমে এবং একইভাবে সিরিয়াল পোর্টের R × D পিনের সাথে কন্ট্রোলারের T × D পিনের সংযোগ MAX232 এর মাধ্যমে হয়। ডাটা প্রেরণ ও গ্রহণের জন্য MAX232 এর দুই সেট লাইন ড্রাইভার থাকে। T1 ও T2 হলো ট্রান্সমিশন লাইনের জন্য ব্যবহৃত লাইন ড্রাইভার এবং R1 ও R2 হলো রিসিভারের লাইন ড্রাইভার।




চিত্র-১০.২ : লাইন ড্রাইভারের সাহায্যে মাইক্রোকন্ট্রোলার ও কম্পিউটারের (T × D ও R × D পিনের) সংযুক্তি




চিত্র-১০.৩ : MAX 232 এর পিন ডায়াগ্রাম

অন্য লাইন ড্রাইভার হলো Maxim এর MAX233। এটি MA232 এর মতো একই কাজ করলেও এতে পিনের মধ্যেও অসামঞ্জস্য বিদ্যমান এবং ক্যাপাসিটরের প্রয়োজন নেই।

## ১০.৪ SCON রেজিস্টারের বিটগুলোর কাজ আলোচনা

### (Describe the Function of Each Bit of SCON Register)

সিরিয়াল কন্ট্রোলের (Serial control) সংক্ষিপ্ত রূপ হলো SCON. এটি একটি 8 বিটের রেজিস্টার, যা সিরিয়াল কমিউনিকেশনে ডাটা ফ্রেমিং এর start bit, stop bit ও data bit এর প্রোগ্রামের জন্য ব্যবহৃত হয়। এটি একটি bit-addressable রেজিস্টার। নিচে এর বিটগুলো বর্ণনা করা হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Value after reset |
| SCON | SM0 | SM1 | SM2 | REN | TB8 | RB8 | T1 | R1 | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

| বিটের নাম | বিটের ঠিকানা | বর্ণনা |
|---|---|---|
| SMO | SCON.7 | (Serial port mode specifier) এরা start ও stop বিট এবং প্রতি character এ বিট সংখ্যা নির্দিষ্টকরণের দ্বারা ডাটার ফ্রেম নির্ণয় করে। |
| SM1 | SCON.6 | এদের বিভিন্ন মিশ্রণ বিভিন্ন সিরিয়াল কমিউনিকেশন মুড নির্ধারণ করে। |
| SM2 | SCON.5 | এটি 8051 এর multiprocessing ক্ষমতা কার্যকর করে। মাইক্রোপ্রসেসর কমিউনিকেশনে ব্যবহৃত হয়। |
| REN | SCON.4 | (Receive Enable) এটি HIGH হলে 8051 R × D পিনের মাধ্যমে ডাটা গ্রহণ করে এবং LOW হলে রিসিভার অকার্যকর হয়। |
| TB8 | SCON.3 | (Transmitted Bit) এটি Mode 2 এবং 3 তে গৃহীত ডাটার 9th বিট, কম ব্যবহৃত হয়। |
| TI | SCON.1 | (Transmit Interrupt flag) 8051 যখন 8 বিট ক্যারেকটার সম্প্রচার শেষ করে, তখন TI flag কে সেট করে এটি নির্দেশ করে যে পরবর্তী বাইট সম্প্রচারের জন্য প্রস্তুত। Stop bit এর আগেও TI bit সেট হয়। |
| RI | SCON.0 | (Receive Interrupt flag) 8051 যখন R × D এর মাধ্যমে সিরিয়ালি ডাটা গ্রহণ করে, তখন এটি start ও stop থেকে মুক্ত হয় এবং বাইটকে SBUF রেজিস্টারে রাখে। এতে RI বিট সেট হয় এবং এটি নির্দেশ করে যে, একটি বাইট গৃহীত হয়েছে এবং হারানোর পূর্বে এটিকে সংগ্রহ করা উচিত। |

## ১০.৫ 8051 এর সিরিয়াল কমিউনিকেশনের মুডসমূহের বর্ণনা

### (Describe the Modes of Serial Communication of the 8051)

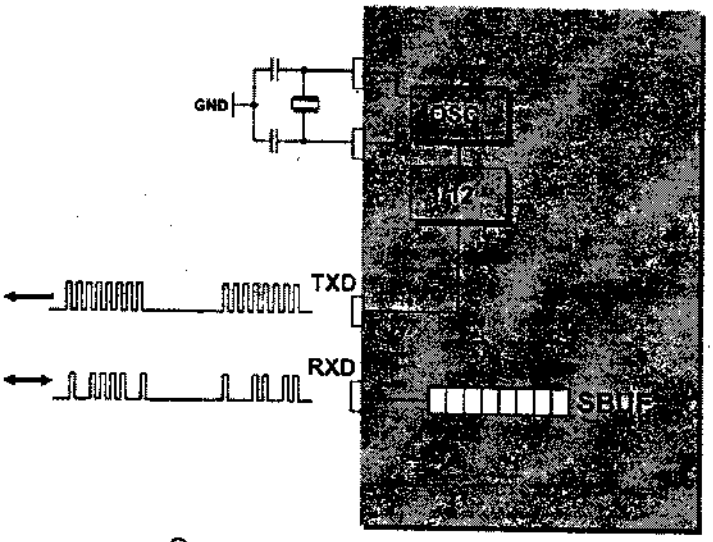
8051 এর 4টি অপারেশন মুড আছে, যারা SCON রেজিস্টারের SM0 ও SM1 দ্বারা নির্বাচিত হয়। এদের মধ্যে তিনটি অ্যাসিনক্রোনাস কমিউনিকেশন এবং চতুর্থটি একটি সরল শিফট রেজিস্টার। সিরিয়াল কমিউনিকেশনের মুড নির্ধারণে SM0 ও SM1 মিশ্রণ নিম্নে দেখানো হলো :

| SM0 | SM1 | Mode | Descritption | Baud Rate |
|---|---|---|---|---|
| 0 | 0 | 0 | 8-bit Shift Register | Quartz frequency এর 1/12 |
| 0 | 1 | 1 | 8-bit UART | Timer 1 দ্বারা নির্ণীত হয় |
| 1 | 0 | 2 | 9-bit UART | Quartz frequency এর 1/32 বা 1/64 |
| 1 | 1 | 3 | 9-bit UART | Timer 1 দ্বারা নির্ণীত হয় |

নিচে মুডগুলোর সংক্ষিপ্ত বিবরণ দেয়া হলো :

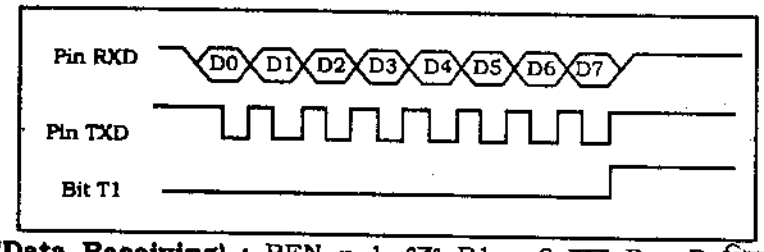
**Serial Communication mode-0 :**

এটি একটি সরল শিফট রেজিস্টার। এতে সিরিয়াল ডাটার প্রবেশ ও নির্গমন হয় R × D পিনের মাধ্যমে এবং T × D পিন শিফট ক্লককে আউটপুট করে। LSB (Least significant bit) কে প্রথম ধরেই 8 বিটের ডাটা সম্প্রচার ও গ্রহণ হয়। এর Baud rate স্থির এবং এটি অভ্যন্তরীণ অসিলেটর ফ্রিকুয়েন্সির 1/12 ভাগ।

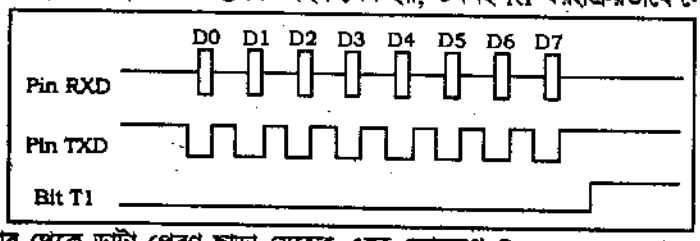



চিত্র-১০.৪ : Serial Mode 0

**ডাটা প্রেরণ (Data Sending) :** SBUF রেজিস্টারে লিখনের মাধ্যমে ডাটা প্রেরণ সূচনা করা হয়। প্রকৃতপক্ষে এই রেজিস্টারে সংঘটিত যে কোন নির্দেশনার (Instruction) পর এই প্রক্রিয়া শুরু হয়। 8 বিট প্রেরণ শেষ হলে TCON রেজিস্টারের TI বিট স্বয়ংক্রিয়ভাবে সেট হয়।




**ডাটা গ্রহণ (Data Receiving) :** REN = 1 এবং R1 = 0 হলে R × D পিনের মাধ্যমে SCON রেজিস্টারের ডাটা গ্রহণ শুরু হয়। যখন 8 বিট গ্রহণ শেষ হয়, তখনই RI স্বয়ংক্রিয়ভাবে সেট পালস ক্রমানুসারে

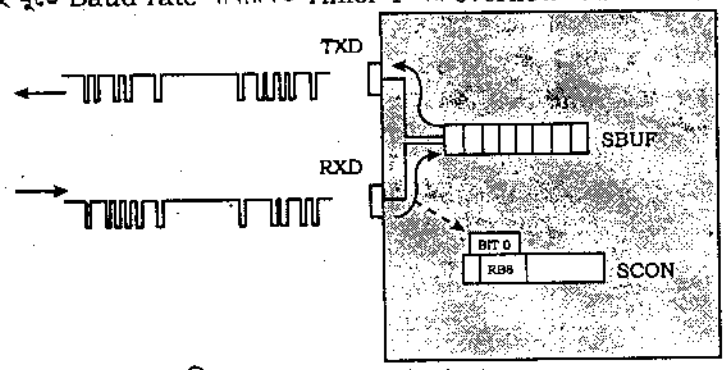



SBUF রেজিস্টার থেকে ডাটা প্রেরণ ছাড়া যেহেতু এতে কোনরূপ Start, stop বা অন্য কোন বিট নেই তাই এই মুড প্রধানত ব্যবহৃত হয় যেখানে ডিভাইসের মধ্যকার দূরত্ব কম, নয়েজ কম এবং অপারেটিং গতি গুরুত্বপূর্ণ।

**Serial Communication Mode- 1**

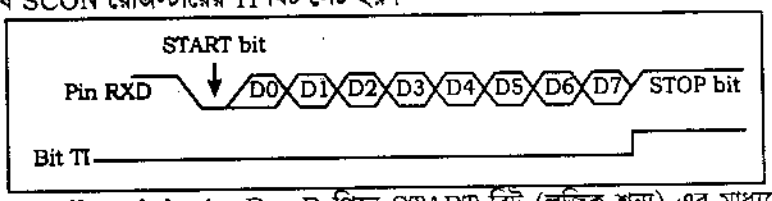
এই মুডে নিচের পদ্ধতি অনুসারে 10 বিট T × D পিনের মাধ্যমে প্রেরণ হয় এবং R × D পিনের মাধ্যমে গৃহীত হয়। প্রথম একটি START বিট (সবসময় 0), 8 বিটের ডাটা এবং একটি STOP বিট (সবসময় 1) START

বিট ডাটা গ্রহণ সূচনা করার জন্য ব্যবহৃত হয় এবং STOP বিট স্বয়ংক্রিয়ভাবে SCON রেজিস্টারের RB8 বিটে লিখিত হয়। এই মুডে Baud rate সাধারণত Timer 1 এর overflow দ্বারা নির্ণিত হয়।

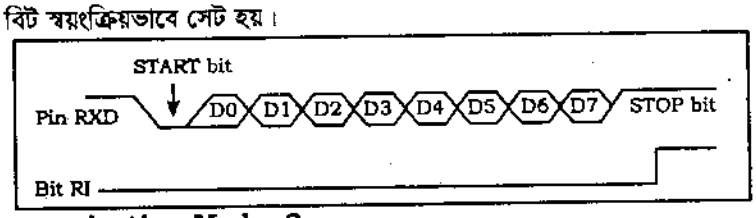



চিত্র-১০.৫ : Serial mode 1

**ডাটা প্রেরণ (Data Sending) :** SBUF রেজিস্টারের লিখনের মাধ্যমে ডাটা প্রেরণ শুরু সূচিত হয় এবং ডাটা সম্প্রচার শেষে SCON রেজিস্টারের TI বিট সেট হয়।

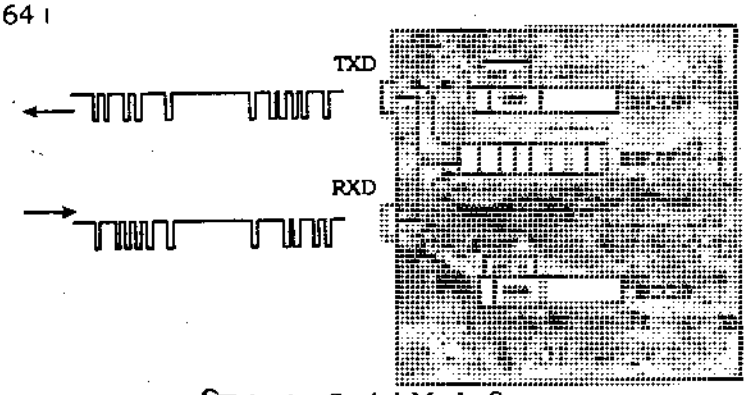



**ডাটা গ্রহণ (Data Receiving) :** R × D পিনে START বিট (লজিক শূন্য) এর মাধ্যমে ডাটা গ্রহণ সূচিত হয়। তবে তজ্জন্য নিম্নবর্ণিত দুটি শর্ত হলো– SCON রেজিস্টারের বিট REN = 1 ও RI = 0 হতে হয়। গ্রহণ শেষ হলে RI বিট স্বয়ংক্রিয়ভাবে সেট হয়।



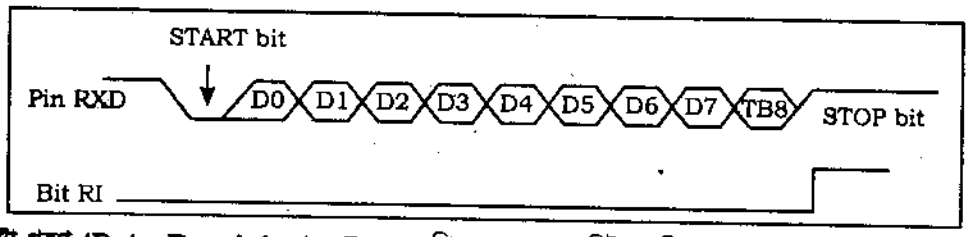

**Serial Communication Mode- 2**

এই মুড, 11টি বিট T × D পিনের মাধ্যমে প্রেরণ অথবা R × D এর মাধ্যমে গ্রহণ হয়। একটি START বিট (সর্বদা শূন্য), 8 বিটের ডাটা (LSB) একটি প্রোগ্রামেবল ৯ম ডাটা বিট এবং একটি STOP বিট (সর্বদা 1)। ডাটা প্রেরণের সময় ৯ম বিট প্রকৃতপক্ষে SCON রেজিস্টারের TB8 বিট। এই বিট সচরাচর পেরিটি বিটের একটি ফাংশন। গ্রহণের সময় ৯ম বিট SCON রেজিস্টারের RB8 বিটে যায়। এর Baud rate অসিলেটর ফ্রিকুয়েন্সির 1/32 বা 1/64।

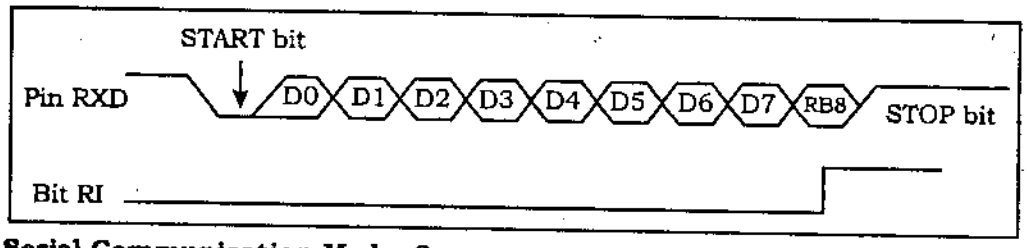



চিত্র-১০.৬ : Serial Mode 2

**ডাটা প্রেরণ (Data Sending) :** SBUF রেজিস্টারের লিখনের মাধ্যমে ডাটা প্রেরণ শুরু সূচিত হয় এবং ডাটা প্রেরণ শেষে SCON রেজিস্টারের TI বিট সেট হয়।




**ডাটা গ্রহণ (Data Receiving) :** R × D পিনে START বিট (লজিক শূন্য) এর মাধ্যমে ডাটা গ্রহণ সূচিত হয়। তবে তজ্জন্য নিম্নবর্ণিত দুটি শর্ত হলো– SCON রেজিস্টারের বিট REN = 1 ও R1 = 0 হতে হয়। গ্রহণ শেষ হলে R1 বিট স্বয়ংক্রিয়ভাবে সেট হয়।




**Serial Communication Mode- 3**

সিরিয়াল কমিউনিকেশন মুড 3 এর সবকিছু মুড 2 এর মতো, শুধু Baud rate ভিন্ন। এর Baud rate পরিবর্তনশীল।

## ১০.৬ সিরিয়াল পোর্টের Baud rate সেটিং ও সজ্জা বর্ণনা

## (Describe Configuring and Setting Baud Rate of Serial Port)

প্রোগ্রামকে অবশ্যই সিরিয়াল পোর্টের Baud rate- সাজাতে হয় যদি সিরিয়াল পোর্ট মুড একবার সাজানো হয়ে যায়। এটি শুধুমাত্র সিরিয়াল পোর্ট মুড 1 ও 3 এর ক্ষেত্রে প্রযোজ্য। 8051 যখন মুড 0 ও 2 থাকে তখন অসিলেট ফ্রিকুয়েন্সির উপর ভিত্তি করে Baud rate নির্ণয় করা হয়। মুড 0 তে Baud rate সর্বদা অসিলেটর ফ্রিকুয়েন্সির 1/12 ভাগ অর্থাৎ 921,583 Baud এবং মুড 2 তে Baud rate নির্ভর করে PCON রেজিস্টারের PCON.7 (SMOD) এর মানের উপর। যদি SMOD = 0 হয়, তবে এটি অসিলেটর ফ্রিকুয়েন্সির 1/64 বা 172,797 Baud (অসিলেটর ফ্রিকুয়েন্সি : 11.059 MHz) এবং যদি SMOD = 1 হয় তবে Baud rate হলো অসিলেটর ফ্রিকুয়েন্সির 1/32 ভাগ (12 ক্লক মুড বিবেচনায়)। মুড 1 ও 3 এ Baud rate নির্ধারিত হয় টাইমার 1 কত ঘন ঘন overflow হয় তার উপর। টাইমার 1 যত ঘন ঘন overflow হয় Baud rate তত বেশি হয়। টাইমার 1 কে একটি নির্দিষ্ট হারে (যা Baud rate নির্ণয় করে) overflow করার অনেক পদ্ধতির মধ্যে অতি সাধারণ পদ্ধতি হলো টাইমার 1 কে Timer mode 2 (auto reload mode) এ সাজানো এবং একটি ফ্রিকুয়েন্সিতে (যা একটি Baud rate উৎপন্ন করে) টাইমার 1 কে overflow করার জন্য একটি Reload value (TH1) সেট করা।

একটি নির্দিষ্ট Baud rate উৎপাদনের নিমিত্তে TH1 এর মান নির্ণয়ের জন্য নিচের সমীকরণ

$$\text{Baud rate} = \frac{2\text{SMOD}}{33} \times \frac{\text{Crystal frequency}}{12 \times [256 - (\text{TH1})]}$$

যদি PCON রেজিস্টারের বিট PCON.7 = 0 হয়, তবে উপরের সমীকরণ থেকে

TH1 = 256 – ((Crystal frequency/384)/Baud) ..... (i)

এবং যদি PCON.7 = 1 হয়, তবে Baud rate কার্যকরীভাবে দ্বিগুণ হয় এবং তজ্জন্য সমীকরণ :

TH1 = 256 – ((Crystal frequency/192)/Baud) ..... ii)

উদাহরণস্বরূপ, যদি ক্রিস্টাল ফ্রিকুয়েন্সি 11.059 মে. হার্টস হয়, তবে সিরিয়াল পোর্টকে 9,200 baud এ সেট করার জন্য সমীকরণ (i) অনুসারে–.

TH1 = 256 – ((11059000/384)/19200) = 254.5

এখন দেখা যাচ্ছে যদি TH1 = 254 ধরা হয়, তবে Baud হয় 14,400 এবং যদি TH1 = 255 ধরা হয়, তবে Baud হয় 28,800। এই সমস্যা সমাধানের জন্য অর্থাৎ 19,200 Baud এর জন্য TH1 = 253 এবং তার জন্য আবশ্যিক সেটিং নিম্নরূপ :

১। সিরিয়াল পোর্টকে মুড 1 বা 3 সাজানো।

২। টাইমার 1 কে টাইমার মুড 2 (8-বিট auto-reload) সেট করা।

৩। TH1 কে 253 তে সেট করা যাতে 19,200 Baud এ সঠিক ক্রিস্টাল ফ্রিকুয়েন্সি প্রতিফলিত হয়।

৪। PCON রেজিস্টারের PCON.7 (SMOD) কে সেট করতে হবে।

PCON রেজিস্টারের PCON.7 (SMOD) বিট সেট হলে Baud rate দ্বিগুণ হয়।

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | × | × | × | 0 | 0 | 0 | 0 | Value after reset |
| PCON | SMOD | | | | GF1 | GF0 | PD | IDL | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

## ১০.৭ SBUF রেজিস্টারের উদ্দেশ্য উল্লেখ

### (Mention the Purpose of SBUF Register)

Serial Data Buffer (SBUF) একটি 8 বিট বিশিষ্ট রেজিস্টার যার মেমোরি ঠিকানা 99h। এটি কমিউনিকেশনে সিরিয়াল ডাটা প্রেরণ ও গ্রহণ মুডে ব্যবহৃত হয়। 8 বিটের প্রেরিতব্য বাইট T × D লাইনের মাধ্যমে অবশ্যই SBUF এ স্থান করতে হয়। SBUF এ লিখনের সময় বাইটকে Start ও Stop এর সাথে ফ্রেমিং করতে হয় এবং T × D লাইনের মাধ্যমে সিরিয়ালি প্রেরণ করা হয়।

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | x | x | x | x | x | x | x | x | Value after reset |
| SBUF | | | | | | | | | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

SBUF 8051 এর R × D লাইনের মাধ্যমে গৃহীত ডাটা বাইট ধারণ করে। যখন ডাটা বিটগুলো R × D পিনের মাধ্যমে সিরিয়ালি গ্রহণ করা হয়, তখন 8051 এটিকে Deframe করে Start/Stop বিটকে দূর করে এবং গৃহীত ডাটার বাইটকে উদ্ধার করে। পরবর্তী ডাটা বাইটকে SBUF এ স্থাপন করা হয়। অর্থাৎ SBUF একটি আউটপুট পোর্ট যখন এতে ডাটা বাইট লিখতে (Written) হয় এবং একটি ইনপুট পোর্ট যখন এতে ডাটা বাইট পঠিত (Read) হয়। নিচে SBUF এর ডাটার লিখন/পঠন দেখানো হলো :

```
MOV SBUF, #' D'        ; Load SBUF = 44h, ASCII for 'D'
MOV SBUF, A            ; Copy Accumulator into SBUF
MOV A, SBUF            ; Copy SBUF into accumulator
```

## ১০.৮ 8051 এর সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণ ও গ্রহণের প্রোগ্রাম

### (Write Program to Transmit and Receive Data Through Serial Port)

**প্রোগ্রাম ১ :** সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের প্রোগ্রাম লেখ।

**সমাধান :**

সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #-6        ; 4800 baud rate
            MOV SCON, #50H      ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI,HERE         ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

**প্রোগ্রাম ২ :** সিরিয়াল পোর্টের মাধ্যমে ডাটা গ্রহণের প্রোগ্রাম লেখ।

**সমাধান :**

সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #-6        ; 4800 baud rate
            MOV SCON, #50H      ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
HERE :      JNB TI, HERE        ; wait for char to come in
            MOV A, SBUF         ; saving incoming byte in A
            MOV P1, A           ; send to port 1
            CLR TI              ; get ready to receive next byte
            SJMP HERE           ; keep getting data
```

**প্রোগ্রাম ৩ :** 4,800 baud এ "A" অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম লেখ।

**সমাধান :**

4,800 baud এ "A" অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #6         ; 4800 baud rate
            MOV SCON, #50H      ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI, HERE        ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

# অনুশীলনী-১০

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **SBUF রেজিস্টারের কাজ কী?** *[বাকাশিবো : '১৫]*

**উত্তর :** Serial Data Buffer (SBUF) একটি 8 বিটের রেজিস্টার, যার মেমোরি ঠিকানা 99h। এটি কমিউনিকেশনে সিরিয়াল ডাটা প্রেরণ ও গ্রহণ মুডে ব্যবহৃত হয়। 8 বিটের প্রেরিতব্য বাইট T × D লাইনের মাধ্যমে অবশ্যই SBUF এ স্থাপন করতে হয়। SBUF এ লিখনের সময় বাইটকে Start ও Stop এর সাথে ফ্রেমিং করতে হয় এবং T × D লাইনের মাধ্যমে সিরিয়ালি প্রেরণ করা হয়।

২। **Baud rate কীসের উপর নির্ভর করে?** *[বাকাশিবো : '১৪]*

**উত্তর :** PCON রেজিস্টারের PCON.7 (SMOD) বিটের উপর baud rate নির্ভর করে।

৩। **সিরিয়াল কমিউনিকেশন বলতে কী বুঝায়?**

**উত্তর :** সিরিয়াল কমিউনিকেশনে একটি একক তারের ভিতর দিয়ে একটি সময় অনুক্রমে (Timer sequence) অর্থাৎ একই সময়ে এক বিট আকারে ডাটা সম্প্রচারিত হয়। অর্থাৎ যে কমিউনিকেশনে সিগন্যাল প্রপাগেশন একটি একক পথ (single path) দিয়ে হয় এবং ডাটা বিটগুলো একটির পর একটি স্থানান্তরিত হয়, তাকে সিরিয়াল কমিউনিকেশন বলে।

৪। **প্যারালাল কমিউনিকেশন বলতে কী বুঝায়?**

**উত্তর :** যে কমিউনিকেশনে একাধিক তার (বাস) একে অপরের সাথে সমান্তরালে অবস্থান করে এবং সব তারে একই সাথে ডাটা স্থানান্তরিত হয়, অর্থাৎ একই সময়ে কোন বাইটের সব বিট স্থানান্তরিত হয়, তাকে প্যারালাল কমিউনিকেশন বলে।

৫। **RS-232 কী?**

**উত্তর :** এটি IBM কর্তৃক প্রবর্তিত কম্পিউটার ও অন্যান্য ডিভাইসের মধ্যে একটি বহুল ব্যবহৃত ইন্টারফেস।

৬। **RS-232 এ High ও Low ভোল্টেজের রেঞ্জ উল্লেখ কর।**

**উত্তর :** RS-232 এ High ও Low ভোল্টেজ নিচের voltage range দ্বারা প্রকাশ করা হয়।

| Bit | Voltage Range (Volts) | |
|---|---|---|
| 0 | + 3 | + 25 |
| 1 | – 25 | – 3 |

৭। **কম্পিউটার ও মাইক্রোকন্ট্রোলারের মাঝে সংযোগে কম্পিউটার সিরিয়াল পোর্টের ন্যূনতম কয়টি পিন দরকার ও কী কী?**

**উত্তর :** কম্পিউটার ও মাইক্রোকন্ট্রোলারের মাঝে সংযোগে কম্পিউটার সিরিয়াল পোর্টের ন্যূনতম তিনটি পিন দরকার, যথা : R × D (Receiver, pin 2), T × D (Transmitter, pin3) এবং Ground (pin5)।

৮। **লাইন ড্রাইভারের কাজ কী?**

**উত্তর :** লাইন ড্রাইভার মাইক্রোকন্ট্রোলার আউটপুট লেভেলকে RS-232 ভোল্টেজ লেভেলে রূপান্তর করে।

৯। **দুটি বহুল ব্যবহৃত লাইন ড্রাইভার বা ভোল্টেজ ড্রাইভারের নাম লিখ।**

**উত্তর :** ভোল্টেজ ড্রাইভার হিসেবে সাধারণত বহুল ব্যবহৃত হয় IC MAX 232, যা লাইন ড্রাইভার হিসেবে অধিক পরিচিত। তাছাড়া Maxim এর MAX233-ও লাইন ড্রাইভার হিসেবে অনেক ব্যবহার হয়।

১০। **দুটি লাইন ড্রাইভার আইসির নাম লিখ।**

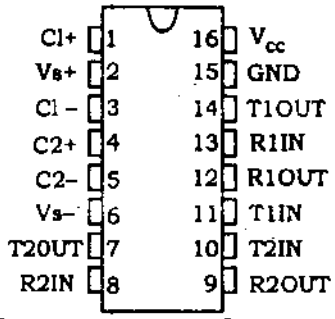
**উত্তর :** লাইন ড্রাইভার আইসি–

(ক) MAX232

(খ) MAX233.

**১১। MAX232 এর পিন ডায়াগ্রাম অংকন কর।**

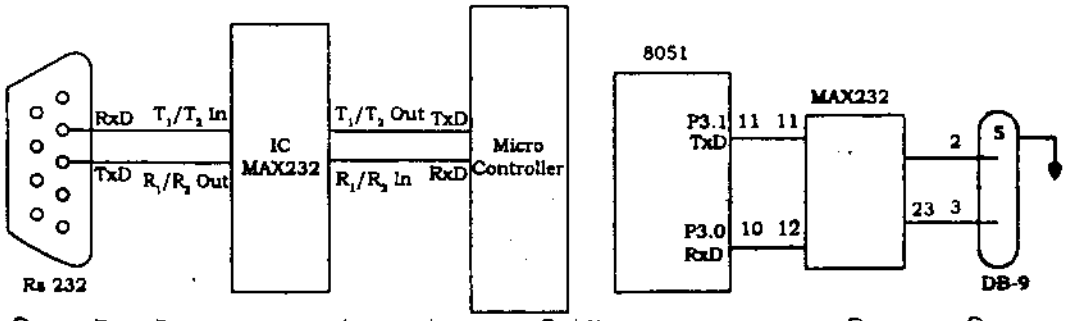
**উত্তর :** MAX232 এর পিন ডায়াগ্রাম নিচে অঙ্কন করা হলো :




**চিত্র :** MAX 232 এর পিন ডায়াগ্রাম

**১২। লাইন ড্রাইভার ব্যবহার করে PC এর সাথে মাইক্রোকন্ট্রোলারের সংযোগ চিত্র দেখাও।**

**উত্তর :** লাইন ড্রাইভার ব্যবহার করে PC এর সাথে মাইক্রোকন্ট্রোলারের সংযোগ চিত্র নিচে দেখানো হলো :




**চিত্র :** লাইন ড্রাইভারের সাহায্যে মাইক্রোকন্ট্রোলার ও কম্পিউটারের (T × D ও R × D পিনের) সংযুক্তি

**১৩। SCON (Serial control) রেজিস্টারের কাজ কী?**

**উত্তর :** SCON (Serial control) একটি 8 বিটের রেজিস্টার, যা সিরিয়াল কমিউনিকেশনে ডাটা ফ্রেমিং এর start bit, stop bit ও data bit এর প্রোগ্রামের জন্য প্রোগ্রাম করার জন্য ব্যবহৃত হয়। এটি একটি bit-addressable রেজিস্টার।

**১৪। SCON (Serial control) রেজিস্টারের সিরিয়াল মুড নির্ধারণী বিটগুলো কী কী?**

**উত্তর :** SCON (Serial control) রেজিস্টারের সিরিয়াল মুড নির্ধারণী বিট হলো SM0 ও SM1.

**১৫। SCON (Serial control) রেজিস্টারের flag bit দুটির নাম লিখ।**

**উত্তর :** SCON (Serial control) এর transmit interrupt flag (TI) এবং receive interrupt flag (RI).

**১৬। Band rate নির্ণয়ের সূত্রটি লিখ।**

**উত্তর :** $\text{Baud rate} = \frac{2SMOD}{33} \times \frac{\text{Crystal frequency}}{12 \times [256 - (TH1)]}$

**১৭। ডাটা কমিউনিকেশন বা আদান প্রদান কয়ভাবে ও কী কী ভাবে হয়?**

**উত্তর :** ডাটা কমিউনিকেশন বা আদান প্রদান দুইভাবে হয়। যথা–

(১) সিরিয়াল ডাটা।

(২) প্যারালাল ডাটা।

**১৮। কোন প্রকার কমিউনিকেশনের ডাটা সম্প্রচারের গতি বেশি?**

**উত্তর :** প্যারালাল কমিউনিকে কমিউনিকেশনের ডাটা সম্প্রচারের গতি বেশি।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **SCON রেজিস্টারের বিটগুলোর কাজ লিখ।**

**উত্তর :** সিরিয়াল কন্ট্রোলের (Serial control) সংক্ষিপ্ত রূপ হলো SCON এটি একটি 8 বিটের রেজিস্টার, যা সিরিয়াল কমিউনিকেশনে ডাটা ফ্রেমিং এর start bit, stop bit ও data bit এর প্রোগ্রামের জন্য ব্যবহৃত হয়। এটি একটি bit-addressable রেজিস্টার। নিচে এর বিটগুলো বর্ণনা করা হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Value after reset |
| SCON | SM0 | SM1 | SM2 | REN | TB8 | RB8 | T1 | R1 | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

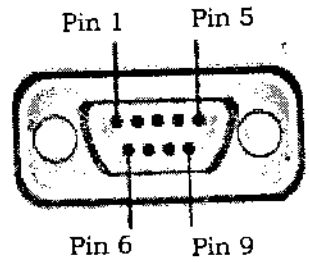
| বিটের নাম | বিটের ঠিকানা | বর্ণনা |
|---|---|---|
| SMO | SCON.7 | (Serial port mode specifier) এরা start ও stop বিট এবং প্রতি character এ বিট সংখ্যা নির্দিষ্টকরণের দ্বারা ডাটার ফ্রেম নির্ণয় করে। |
| SM1 | SCON.6 | এদের বিভিন্ন মিশ্রণ বিভিন্ন সিরিয়াল কমিউনিকেশন মুড নির্ধারণ করে। |
| SM2 | SCON.5 | এটি 8051 এর multiprocessing ক্ষমতা কার্যকর করে। মাইক্রোপ্রসেসর কমিউনিকেশনে ব্যবহৃত হয়। |
| REN | SCON.4 | (Receive Enable) এটি HIGH হলে 8051 R × D পিনের মাধ্যমে ডাটা গ্রহণ করে এবং LOW হলে রিসিভার অকার্যকর হয়। |
| TB8 | SCON.3 | (Transmitted Bit) এটি Mode 2 এবং 3 তে গৃহীত ডাটার 9th বিট, কম ব্যবহৃত হয়। |
| TI | SCON.1 | (Transmit Interrupt flag) 8051 যখন 8 বিট ক্যারেকটার সম্প্রচার শেষ করে, তখন TI flag কে সেট করে এটি নির্দেশ করে যে পরবর্তী বাইট সম্প্রচারের জন্য প্রস্তুত। Stop bit এর আগেও TI bit সেট হয়। |
| RI | SCON.0 | (Receive Interrupt flag) 8051 যখন R × D এর মাধ্যমে সিরিয়ালি ডাটা গ্রহণ করে, তখন এটি start ও stop থেকে মুক্ত হয় এবং বাইটকে SBUF রেজিস্টারে রাখে। এতে RI বিট সেট হয় এবং এটি নির্দেশ করে যে, একটি বাইট গৃহীত হয়েছে এবং হারানোর পূর্বে এটিকে সংগ্রহ করা উচিত। |

২। **DB-9 RS-232 এর পিন ডায়াগ্রাম অঙ্কন করে এর বিভিন্ন পিনগুলোর কাজ উল্লেখ কর।**

**উত্তর :** DB-9 RS-232 এর পিন ডায়াগ্রাম অংকন করে এর বিভিন্ন পিনগুলোর কাজ নিম্নরূপ :

| | |
|---|---|
| Pin 1 | DCD |
| Pin 2 | RXD |
| Pin 3 | TXD |
| Pin 4 | DTR |
| Pin 5 | GND |
| Pin 6 | DSR |
| Pin 7 | RTS |
| Pin 8 | CTS |
| Pin 9 | RI |

RS232 Pinout (9 Pin Male)




চিত্র : RS-232 এর পিন ডায়াগ্রাম

**পিনের বর্ণনা :**

| PIN | আদ্যাক্ষরা (Acronym) | পূর্ণ নাম | দিক | বর্ণনা |
|---|---|---|---|---|
| 1 | – DCD | Data carrier detect | ← | মডেম সংযুক্ত। |
| 2 | R × D | Receive data | ← | কম্পিউটার বা ডিভাইস ডাটা বাইট গ্রহণ করে। |
| 3 | T × D | Transmit Data | → | কম্পিউটার বা ডিভাইস থেকে ডাটা বাইট বের হয়। |
| 4 | DTR | Data terminal ready | → | এই ডিভাইস যোগাযোগের জন্য প্রস্তুত। |
| 5 | GND | Signal ground | | গ্রাউন্ড। |
| 6 | – DSR | Data set ready | ← | এই ডিভাইস যোগাযোগের জন্য প্রস্তুত। |
| 7 | – RTS | Request to send | → | RTS/CTS প্রবাহ নিয়ন্ত্রণ। |
| 8 | – CTS | Clear to send | ← | RTS/CTS প্রবাহ নিয়ন্ত্রণ। |
| 9 | RI | Ring indicator | ← | টেলিফোন লাইনে রিংগিং। |

৩। **সিরিয়াল মুড কয়টি Baud rate সহ উল্লেখ কর।**

**উত্তর :** নিচে সিরিয়াল মুড কয়টি Baud rate সহ উল্লেখ করা হলো :

| Mode | Description | Baud Rate |
|---|---|---|
| 0 | 8-bit Shift Register | Quartz frequency এর 1/12 |
| 1 | 8-bit UART | Timer 1 দ্বারা নির্ণীত হয় |
| 2 | 9-bit UART | Quartz frequency এর 1/32 বা 1/64 |
| 3 | 9-bit UART | Timer 1 দ্বারা নির্ণীত হয় |

৪। **সিরিয়াল পোর্টের Baud rate সেটিং বর্ণনা কর।**

**উত্তর :** প্রোগ্রামকে অবশ্যই সিরিয়াল পোর্টের Baud rate- সাজাতে হয় যদি সিরিয়াল পোর্ট মুড একবার সাজানো হয়ে যায়। এটি শুধুমাত্র সিরিয়াল পোর্ট মুড 1 ও 3 এর ক্ষেত্রে প্রযোজ্য। 8051 যখন মুড 0 ও 2 থাকে তখন অসিলেট ফ্রিকুয়েন্সির উপর ভিত্তি করে Baud rate নির্ণয় করা হয়। মুড 0 তে Baud rate সর্বদা অসিলেটর ফ্রিকুয়েন্সির 1/12 ভাগ অর্থাৎ 921,583 Baud এবং মুড 2 তে Baud rate নির্ভর করে PCON রেজিস্টারের PCON.7 (SMOD) এর মানের উপর। যদি SMOD = 0 হয়, তবে এটি অসিলেটর ফ্রিকুয়েন্সির 1/64 বা 172,797 Baud (অসিলেটর ফ্রিকুয়েন্সি : 11.059 মে. হার্টস) এবং যদি SMOD = 1 হয় তবে Baud rate হলো অসিলেটর ফ্রিকুয়েন্সির 1/32 ভাগ (12 ক্লক মুড বিবেচনায়)। মুড 1 ও 3 এ Baud rate নির্ধারিত হয় টাইমার 1 কত ঘন ঘন overflow হয় তার উপর। টাইমার 1 যত ঘন ঘন overflow হয় Baud rate তত বেশি হয়। টাইমার 1 কে একটি নির্দিষ্ট হারে (যা Baud rate নির্ণয় করে) overflow করার অনেক পদ্ধতির মধ্যে অতি সাধারণ পদ্ধতি হলো টাইমার 1 কে Timer mode 2 (auto reload mode) এ সাজানো এবং একটি ফ্রিকুয়েন্সিতে (যা একটি Baud rate উৎপন্ন করে) টাইমার 1 কে overflow করার জন্য একটি Reload value (TH1) সেট করা।

একটি নির্দিষ্ট Baud rate উৎপাদনের নিমিত্তে TH1 এর মান নির্ণয়ের জন্য নিচের সমীকরণ

$$\text{Baud rate} = \frac{2\text{SMOD}}{33} \times \frac{\text{Crystal frequency}}{12 \times [256 - (\text{TH1})]}$$

যদি PCON রেজিস্টারের বিট PCON.7 = 0 হয়, তবে উপরের সমীকরণ থেকে

TH1 = 256 – ((Crystal frequency/384)/Baud) ..... (i)

এবং যদি PCON.7 = 1 হয়, তবে Baud rate কার্যকরীভাবে দ্বিগুণ হয় এবং তজ্জন্য সমীকরণ :

TH1 = 256 – ((Crystal frequency/192)/Baud) ..... (ii)

উদাহরণস্বরূপ, যদি ক্রিস্টাল ফ্রিকুয়েন্সি 11.059 মে. হার্টস হয়, তবে সিরিয়াল পোর্টকে 9,200 baud এ ে করার জন্য সমীকরণ (i) অনুসারে–

TH1 = 256 – ((11059000/384)/19200) = 254.5

এখন দেখা যাচ্ছে যদি TH1 = 254 ধরা হয়, তবে Baud হয় 14,400 এবং যদি TH1 = 255 ধরা য তবে Baud হয় 28,800। এই সমস্যা সমাধানের জন্য অর্থাৎ 19,200 Baud এর জন্য TH1 = 253 এ তার জন্য আবশ্যিক সেটিং নিম্নরূপ :

১। সিরিয়াল পোর্টকে মুড 1 বা 3 সাজানো।

২। টাইমার 1 কে টাইমার মুড 2 (8-বিট auto-reload) সেট করা।

৩। TH1 কে 253 তে সেট করা যাতে 19,200 Baud এ সঠিক ক্রিস্টাল ফ্রিকুয়েন্সি প্রতিফলিত হয়।

৪। PCON রেজিস্টারের PCON.7 (SMOD) কে সেট করতে হবে।

PCON রেজিস্টারের PCON.7 (SMOD) বিট সেট হলে Baud rate দ্বিগুণ হয়।

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | × | × | × | 0 | 0 | 0 | 0 | Value after rese |
| PCON | SMOD | | | | GF1 | GF0 | PD | ID | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit | |

**৫। সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের 8051 প্রোগ্রাম লিখ।**

**উত্তর :** সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #-6        ; 4800 baud rate
            MOV SCON, #50H      ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI,HERE         ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

**৬। সিরিয়াল পোর্টের মাধ্যমে ডাটা গ্রহণের 8051 প্রোগ্রাম লিখ।**

**উত্তর :** সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #-6        ; 4800 baud rate
            MOV SCON,#50H       ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
HERE :      JNB TI, HERE        ; wait for char to come in
            MOV A, SBUF         ; saving incoming byte in A
            MOV P1, A           ; send to port 1
            CLR TI              ; get ready to receive next byte
            SJMP HERE           ; keep getting data
```

**৭। 4800 baud এ 'A' অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম লিখ।**

**উত্তর :** 4,800 baud এ "A" অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1,#-6         ; 4800 baud rate
            MOV SCON,#50H       ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI, HERE        ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

**৮। সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের প্রোগ্রাম লেখ।**

**উত্তর :** সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #-6        ; 4800 baud rate
            MOV SCON,#50H       ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI,HERE         ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

**৯। সিরিয়াল পোর্টের মাধ্যমে ডাটা গ্রহণের প্রোগ্রাম লেখ।**

**উত্তর :** সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণের একটি সরল প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1,#-6         ; 4800 baud rate
            MOV SCON,#50H       ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
HERE :      JNB TI,HERE         ; wait for char to come in
            MOV A, SBUF         ; saving incoming byte in A
            MOV P1, A           ; send to port 1
            CLR TI              ; get ready to receive next byte
            SJMP HERE           ; keep getting data
```

**১০। 4,800 baud এ "A" অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম লেখ।**

**উত্তর :** 4,800 baud এ "A" অক্ষরটি অবিরাম স্থানান্তর করার জন্য 8051 প্রোগ্রাম নিচে উল্লেখ করা হলো :

```
            MOV TMOD, #20H      ; timer 1, mode 2 (auto-reload)
            MOV TH1, #6         ; 4800 baud rate
            MOV SCON, #50H      ; 8-bit, 1 stop, REN enabled
            SETB TR1            ; start timer 1
AGAIN :     MOV SBUF, #" A"     ; letter "A" to transfer
HERE :      JNB TI, HERE        ; wait for the last bit
            CLR TI              ; clear TI for next char
            SJMP AGAIN          ; keep sending A
```

## রচনামূলক প্রশ্নোত্তর

[বাকাশিবো : '১

১। **সিরিয়াল ডাটা প্রেরণ ও গ্রহণের জন্য 8051 প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ১০.৮ নং দ্রষ্টব্য।

২। **সিরিয়াল কমিউনিকেশনের মুডগুলো বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১০.৫ নং দ্রষ্টব্য।

৩। **সিরিয়াল পোর্টের Baud rate সেটিং ও সজ্জার বর্ণনা দাও।**

**উত্তর :** অনুচ্ছেদ ১০.৬ নং দ্রষ্টব্য।

৪। **আরএস 232 ডিবি 9 পিন কানেক্টর আলোচনা কর।**

**উত্তর :** অনুচ্ছেদ ১০.২ নং দ্রষ্টব্য।

৫। **লাইন ড্রাইভারের পিন ও সংযোগ আলোচনা কর।**

**উত্তর :** অনুচ্ছেদ ১০.৩ নং দ্রষ্টব্য।

৬। **SCON রেজিস্টারের বিটগুলোর কাজ আলোচনা কর।**

**উত্তর :** অনুচ্ছেদ ১০.৪ নং দ্রষ্টব্য।

পলিটেকনিকের সকল বই ডাওনলোড করতে
ভিজিটঃ
www.BDeBooks.Com/polytechnic

# অধ্যায় ১১

# 8051 মাইক্রোকন্ট্রোলারের ইন্টারাপ্ট

# Under Stand the Interrupt of the 8051 Microcontroller

## ১১.০ ভূমিকা

### (Introduction)

ইন্টারাপ্ট হলো মাইক্রোপ্রসেসরের প্রতি অনুরোধ। কোন নির্দিষ্ট সার্ভিস রুটিন সম্পন্ন করার জন্য ইন্টারাপ্টের মাধ্যমে মাইক্রোপ্রসেসরকে অনুরোধ করা হয়। এই ধরনের সার্ভিস রুটিনকে ইন্টারাপ্ট সার্ভিস রুটিন বলা হয়। ইন্টারাপ্ট কার্যকরী হলে মাইক্রোপ্রসেসর চলমান কাজকে সাময়িকভাবে বন্ধ রেখে ইন্টারাপ্টের জন্য সার্ভিস রুটিন সম্পন্ন করে।

বিভিন্ন মাইক্রোপ্রসেসরে বিভিন্নভাবে বিভিন্ন ধরনের ইন্টারাপ্টের সুযোগ থাকে। সাধারণত মাইক্রোপ্রসেসরে ৩ ধরনের ইন্টারাপ্ট থাকে। যথা–

(ক) হার্ডওয়্যার ইন্টারাপ্ট (Hardware Interrupt)

(খ) সফটওয়্যার ইন্টারাপ্ট (Software Interrupt) এবং

(গ) ইন্টার্নাল ইন্টারাপ্ট (Internal Interrupt)

**(ক) হার্ডওয়্যার ইন্টারাপ্ট (Hardware Interrupt) :** মাইক্রোপ্রসেসরের পিনের মাধ্যমে বা সিগন্যালের মাধ্যমে যেসকল ইন্টারাপ্ট সংঘটিত হয়, তাদেরকে হার্ডওয়্যার ইন্টারাপ্ট বলা হয়। যেমন– NMI এবং INTR পিনের মাধ্যমে হার্ডওয়্যার ইন্টারাপ্ট সম্পাদিত হতে পারে।

**(খ) সফটওয়্যার ইন্টারাপ্ট (Software Interrupt) :** ইন্সট্রাকশনের মাধ্যমে যেসকল ইন্টারাপ্ট সংঘটিত হয়, তাদেরকে সফটওয়্যার ইন্টারাপ্ট বলা হয়। INT এবং INTO ইন্সট্রাকশনের মাধ্যমে সফটওয়্যার ইন্টারাপ্ট হতে পারে।

**(গ) ইন্টার্নাল ইন্টারাপ্ট (Internal Interrupt) :** মাইক্রোপ্রসেসরের আভ্যন্তরীন কোন অপারেশনের ফলে সৃষ্ট ইন্টারাপ্টকে ইন্টার্নাল ইন্টারাপ্ট বলা হয়।

## ১১.১ 8051 এর ইন্টারাপ্ট এর উৎসের তালিকা

### (List the Source of Interrupt of the 8051)

বর্তমান অপারেটিং প্রোগ্রামের বাইরের কোন ইভেন্ট দ্বারা একটি ইন্টারাপ্টের সূত্রপাত হয়। ইন্টারাপ্ট ইভেন্ট বর্তমান প্রোগ্রামে অসময়ে (asynchronously) সংঘটিত হয়, কারণ এটি কখন সংঘটিত হবে তা আগে জানার কোন উপায় নেই। ইন্টারাপ্ট হার্ডওয়্যার ও সফটওয়্যার উভয় ধরনের হতে পারে। একটি প্রোগ্রাম সাবরুটিন ডাকার মতো (Program subroutine call) ইন্টারাপ্ট মূল প্রোগ্রাম নির্বাহে একটি সাময়িক গতি পরিবর্তন করে। 8051 মাইক্রোকন্ট্রোলারে পাঁচটি ইন্টারাপ্ট উৎস আছে। যেহেতু প্রধান RESET কেও একটি ইন্টারাপ্ট উৎস হিসেবে বিবেচনা করা হয়, তাই 8051 এর জন্য ছয়টি ইন্টারাপ্ট উৎসের তালিকা নিচে দেওয়া হলো :

| No. | Interrupt | Flag |
|---|---|---|
| 1. | Timer/Counter 1 | TF1 |
| 2. | External interrupt | TF1 |
| 3. | Serial port | RI বা TI |
| 4. | Timer/Counter 0 | TF0 |
| 5. | System RESET | RST |
| 6. | External interrupt 0 | IE0 |

## ১১.২ ইন্টারাপ্ট সার্ভিস রুটিন

### (Define Interrupt Service Routine (ISR))

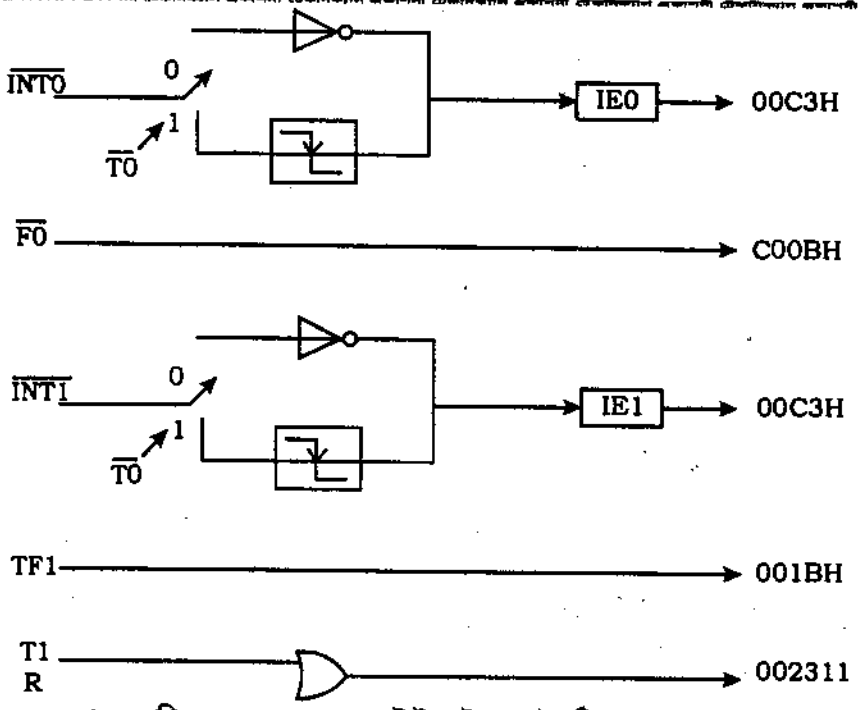
ইন্টারাপ্ট সার্ভিস রুটিন (Interrupt Service Routing) এর সংক্ষিপ্ত নাম হলো ISR। প্রতিটি ইন্টারাপ্টের জন্য একটি করে Interrupt Service Routine (ISR) থাকা আবশ্যক। প্রতিটি ইন্টারাপ্ট এর জন্য মেমোরিতে নির্দিষ্ট লোকেশন থাকে, যা ISR এর অ্যাড্রেস ধারণ করে একে ভেক্টর ঠিকানা বলে। ISR এর অ্যাড্রেসকে ধারণ করার জন্য মেমোরি লোকেশনের গ্রুপকে ইন্টারাপ্ট ভেক্টর টেবিল বলে। চলমান প্রোগ্রামে কোন ইন্টারাপ্ট সংঘটিত হলে যে ছোট প্রোগ্রাম প্রসেসর বা কন্ট্রোলারকে কী করতে হবে তা অবগত করে, তাকে Interrupt Service Routine (ISR) বলে। এটিকে Interrupt handler-ও বলে। ISR ইন্টারাপ্টের প্রত্যুত্তরে সম্পাদিত হয় এবং সাধারণত একটি ডিভাইসের ইনপুট/আউটপুট কাজ সমাধা করে। যখন কোন ইন্টারাপ্ট সংঘটিত হয়, তখন মূল প্রোগ্রাম সম্পাদন সাময়িক স্থগিত হয় এবং প্রোগ্রামকে ISR এ branching করে। ISR সম্পাদিত হয় এবং RETI নির্দেশনার মাধ্যমে সমাপ্ত হয়। ইন্টারাপ্ট সম্পাদন শেষে মূল প্রোগ্রাম (যেখান থেকে ইন্টারাপ্ট শুরু) পুনরায় চলতে থাকে। যখন কোন ইন্টারাপ্ট গৃহীত হয়, তখন PC (Program counter) তে যে মান জমা হয়, তাকে Interrupt vector বলে। এটি ইন্টারাপ্ট উৎসের জন্য ISR এর শুরুর ঠিকানা।

## ১১.৩ ইন্টারাপ্ট অগ্রগণ্যতা ও ভেক্টর অবস্থান উল্লেখ করণ

### (Mention the Interrupt Priority and Vector Locations)

একটি স্বতন্ত্র ইন্টারাপ্ট উৎসকে দুটি অগ্রগণ্যতা স্তরের (priority level) এর একটিতে ধার্য হতে পারে। প্রত্যেক ইন্টারাপ্ট উৎসের জন্য অগ্রগণ্যতা স্তর প্রোগ্রাম করার জন্য একটি বিশেষ রেজিস্টার, Interrupt priority (IP) ব্যবহৃত হয়। উচ্চ অগ্রগণ্যতার ইন্টারাপ্টের একটি ISR রুটিন বিঘ্নিত হতে পারে না। তবে একটি নিম্ন অগ্রগণ্যতার ইন্টারাপ্ট উচ্চ অগ্রগণ্যতার ইন্টারাপ্ট দ্বারা বিঘ্নিত হতে পারে। যদি একই সময়ে ভিন্ন অগ্রগণ্যতার দুটি ইন্টারাপ্ট আসে তবে প্রথমে উচ্চ অগ্রগণ্যতার ইন্টারাপ্টের একটি ISR রুটিন বিঘ্নিত হতে পারে না। তবে একটি নিম্ন অগ্রগণ্যতার ইন্টারাপ্ট উচ্চ অগ্রগণ্যতার ইন্টারাপ্ট দ্বারা বিঘ্নিত হতে পারে। যদি একই সময়ে ভিন্ন অগ্রগণ্যতার দুটি ইন্টারাপ্ট আসে তবে প্রথমে উচ্চ অগ্রগণ্যতার ইন্টারাপ্ট এবং পরে নিম্ন অগ্রগণ্যতার ইন্টারাপ্ট সম্পাদিত হয়। প্রকৃতপক্ষে অগ্রগণ্যতার পরিকল্পনা একটি অভ্যন্তরীণ polling sequence ছাড়া আর কিছু নয়, যাতে 8051 ঐ sequence এ ইন্টারাপ্টকে poll করে। যখন 8051 এ পাওয়ার প্রয়োগ করা হয়, তখন এর ইন্টারাপ্টসমূহের অগ্রগণ্যতা নিচের তালিকা অনুসারে হয় :

| ইন্টারাপ্ট উৎস (Interrupt source) | ফ্ল্যাগ (Flag) | অগ্রগণ্যতা স্তর (Priority level) | ভেক্টর ঠিকানা (Vector address) |
|---|---|---|---|
| External interrupt 0 | IE0 | Highest | 0003h |
| Timer/counter 0 | TF0 | ↓ | 000Bh |
| External interrupt 1 | IE1 | ↓ | 0013h |
| Timer/counter 1 | TF1 | ↓ | 001Bh |
| Serial port | RI or TI | Lowest | 0023h |




চিত্র-১১.১ : 8051 এর ইন্টারাপ্ট ও ভেক্টর ঠিকানা

## ১১.৪ ইন্টারাপ্ট এনাবল রেজিস্টার IE বর্ণনা

**(Describe Each Bit of the Interrupt Enable (IE) Register)**

8051 মাইক্রোকন্ট্রোলারের বিভিন্ন ইন্টারাপ্টের নিয়ন্ত্রণের জন্য IE register ব্যবহার করা হয়। এটি একটি 8 বিটের বিশেষ রেজিস্টার (SFR), যা অভ্যন্তরীণ মেমোরির A8h ঠিকানায় সংরক্ষিত। এটি একটি bit-addressable রেজিস্টার। নিচে এর বিটগুলো সম্পর্কে সংক্ষেপে বর্ণনা করা হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | X | 0 | 0 | 0 | 0 | 0 | 0 | Value after reset |
| IE | EA | | ET2 | ES | ET1 | EX1 | ET0 | EX0 | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

EA – Global interrupt enable/disable :

0- সমস্ত interrupt requests অকার্যকর করে।

1- সমস্ত interrupt requests কার্যকর করে।

ES– Enables or disables serial interrupt :

0- UART system কোন interrupt উৎপন্ন করতে পারে না।

1- UART system interrupt উৎপন্ন করতে পারে।

ET1– Bit enables or disables Timer 1 interrupt :

0- Timer 1 কোন interrupt উৎপন্ন করতে পারে না।

1- Timer interrupt উৎপন্ন করতে পারে।

EX1– Bit enables or disables external 1 interrupt :

0- Pin INT1 logic state এর পরিবর্তনে কোন interrupt উৎপন্ন করতে পারে না।

1- Pin INT 1 logic state এর পরিবর্তনে interrupt উৎপন্ন করতে পারে।

ET0– Bit enables or disables timer 0 interrupt :

0- Timer 0 কোন interrupt উৎপন্ন করতে পারে না।

1- Timer 0 interrupt উৎপন্ন করতে পারে।

EX0– Bit enables or disables external 0 interrupt :

0- Pin INT0 logic state এর পরিবর্তনে কোন interrupt উৎপন্ন করতে পারে না।

1- Pin INT0 logic state এর পরিবর্তনে interrupt উৎপন্ন করতে পারে।

## ১১.৫ ইন্টারাপ্ট কার্যকর ও অকার্যকর করার প্রক্রিয়া বর্ণনা

### (Describe the Procedure of Enabling and Disabling Interrupt)

প্রত্যেক ইন্টারাপ্ট উৎসকে IE (Interrupt enable) রেজিস্টারের মাধ্যমে স্বতন্ত্রভাবে কার্যকর বা অকার্যকর করা যায়। IE রেজিস্টারের EA বিট সেট বা clear করে সমস্ত ইন্টারাপ্ট উৎসকে সর্বজনীনভাবে কার্যকর বা অকার্যকর করা হয়। মাইক্রোকন্ট্রোলার রিসেটে সব ইন্টারাপ্ট অকার্যকর (masked) থাকে, অর্থাৎ এরা অকার্যকরী হলেও মাইক্রোকন্ট্রোলার ইন্টারাপ্ট সম্পাদন করবে না।

প্রোগ্রামে নির্দিষ্টভাবে উল্লেখ করে ইন্টারাপ্টকে কার্যকরী করা হয়। ইন্টারাপ্টকে কার্যকর (masking) বা অকার্যকর (unmasking) করার জন্য নিচের ধাপগুলো অনুসরণ করা হয় :

১। IE রেজিস্টারের D7 বিট (EA) কে অবশ্যই High তে সেট করতে হবে, যাতে রেজিস্টারের অন্যান্য ইন্টারাপ্টগুলো কার্যকর হয়।

২। EA এর মান নির্ধারণ।

যদি EA = 1 হয়, সব ইন্টারাপ্ট কার্যকর হয় এবং IE রেজিস্টারে সংশ্লিষ্ট বিট High থাকে।

যদি EA = 0 হয়, তবে সংশ্লিষ্ট বিট High থাকলেও কোন ইন্টারাপ্টই কার্যকর হবে না।

৩। এবার কোন ইন্টারাপ্টকে কার্যকর করার জন্য ঐ ইন্টারাপ্ট উৎসের Enable bit সেট করতে হয়।

উদাহরণস্বরূপ Timer 1 ইন্টারাপ্টকে কার্যকর করার জন্য কোড নিম্নরূপ :

```
SETB ET1                    ; Enable Timer 1 Interrupt
SETB EA                     ; Set Global enable bit
বা, MOV IE, # 10001000B     ; Set EA and ET1 or IE register.
```

## ১১.৬ একটি ইন্টারাপ্ট সম্পাদনের ধাপসমূহ উল্লেখ করণ

### (Mention the Steps in Executing an Interrupt) :

একটি ইন্টারাপ্টের সক্রিয়তার মাইক্রোকন্ট্রোলার নিচের ধাপগুলো অনুসরণ করে :

১। এটি তার চলমান নির্দেশনা সম্পাদনা শেষ করে।

২। এটি প্রোগ্রাম কাউন্টারে রক্ষিত পরবর্তী নির্দেশনার ঠিকানা stack এ জমা করে.

৩। এটি ইন্টারাপ্টের বর্তমান অবস্থা (status) অভ্যন্তরীণভাবে (stack এ নয়) জমা রাখে।

৪। এটি Interrupt vector location থেকে ISR এর ঠিকানা নেয় এবং তাতে জাম্প করে। মাইক্রোকন্ট্রোলার ISR সম্পাদন শুরু করে যতক্ষণ পর্যন্ত না সাবরুটিনের শেষ নির্দেশনা (RETI, Return from interrupt) আসে।

৫। RETI সম্পাদন শেষ হলে মাইক্রোকন্টোলার তার পূর্বের অবস্থানে (যেখান থেকে ইন্টারাপ্ট শুরু হয়) ফিরে আসে। এরপর প্রথমে এটি Stack থেকে প্রথম দুই বাইট প্রোগ্রাম কাউন্টারে POP করে পরবর্তী সম্পাদিত নির্দেশনার ঠিকানা নেয়। তারপর ঐ ঠিকানা থেকে সম্পাদন শুরু করে।

## ১১.৭ ইন্টারাপ্টের সময় রেজিস্টার সুরক্ষাকরণ বর্ণনা

### (Describe the Register Protection During Interrupt)

এটি Interrupt handler বা রুটিনে একটি অত্যন্ত গুরুত্বপূর্ণ নিয়ম হলো ইন্টারাপ্ট সূচিত হওয়ার সময় যে অবস্থায় ছিল ত্যাগ করার সময় প্রসেসরকে অবশ্যই একই অবস্থায় থাকতে হবে। অর্থাৎ যদি কোন ইন্টারাপ্ট রেজিস্টার (যথা, একুমুলেটর) ব্যবহার করে, তবে এটি অবশ্যই নিশ্চিত করতে হবে যে, রেজিস্টারের মান ইন্টারাপ্টের শুরুতে যা ছিল শেষেও তাই থাকে। ইন্টারাপ্ট রুটিন সম্পাদনকালে অনেক সময় (বিশেষ করে MOV নির্দেশনায় রেজিস্টার মান পরিবর্তন হয়ে যায়। সাধারণত ইন্টারাপ্টের সময় স্ট্যাক (stack) এ PUSH ও POP অনুক্রম দ্বারা রেজিস্টারের মানকে সুরক্ষা করা হয়।

ইন্টারাপ্ট রুটিনকে নিম্নবর্ণিত রেজিস্টারসমূহকে অবশ্যই সুরক্ষা করতে হবে–

1. PSW
2. PSW
3. B
4. DPTR (DPH/DPL)
5. ACC
6. Registers R0–R7

এখানে, PSW রেজিস্টারে অনেক স্বতন্ত্র বিট আছে, যাদেরকে 8051 এর বিভিন্ন নির্দেশনা দ্বারা সেট করা যায়। সেক্ষেত্রে ইন্টারাপ্টের শুরুতে এবং শেষে Stack এ Pushing ও Popping করে PSW রেজিস্টারকে সর্বদা সুরক্ষা করা একটি উত্তম পরিকল্পনা।

PUSH R0 কোডটিকে অধিকাংশ অ্যাসেম্বলার বিবেচনা করে না।

কারণ R0 কোন গ্রহণযোগ্য ঠিকানা নয়, যাতে PUSH ও POP নির্দেশনা ব্যবহার করা যায়। R0 অভ্যন্তরীণ মেমোরি ঠিকানা 00h, 08h, 10h বা 18 কে অর্পণ করতে পারে (refer)। তাই ইন্টারাপ্ট রুটিনে কোন "R" রেজিস্টার ব্যবহার হলে একে স্ট্যাকে পরম ঠিকানা (absolute address) তে PUSH করতে হয়। তাই PUSH R0 এর পরিবর্তে PUSH 00h সম্পাদন করা হয়। তবে এটি default রেজিস্টারের ক্ষেত্রে প্রযোজ্য। যদি বিকল্প রেজিস্টার নির্বাচন করা হয়, তবে ঐ রেজিস্টারের সঙ্গতিপূর্ণ ঠিকানায় PUSH করতে হবে।

## ১১.৮ বাহ্যিক হার্ডওয়্যার, টাইমার ও সিরিয়াল কমিউনিকেশন ইন্টারাপ্ট বর্ণনা

### (Describe External Hardware, Timer and Serial Communication Interrupt)

**বাহ্যিক হার্ডওয়্যার ইন্টারাপ্ট (External hardware interrupt) :**

8051 এর $\overline{INT0}$ (পিন 12) বা $\overline{INT1}$ (পিন 13) পিনের একটি Low level বা Negative edge এর ফলে বাহ্যিক ইন্টারাপ্ট ঘটে। এই ইন্টারাপ্টের ফলে উৎপন্ন flag bit গুলো হলো : যথাক্রমে IE0 ও IE1 (TCON রেজিস্টার)। এবং যদি ইন্টারাপ্ট Transition activated (Negative edge) হয়, এদেরকে হার্ডওয়্যার দ্বারা clear করা হয়। যদি ইন্টারাপ্ট Level-addtivated হয়, তবে মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ হার্ডওয়্যারের পরিবর্তে বাহ্যিক ইন্টারাপ্ট উৎস Request flag এর লেভেলকে নিয়ন্ত্রণ করে।

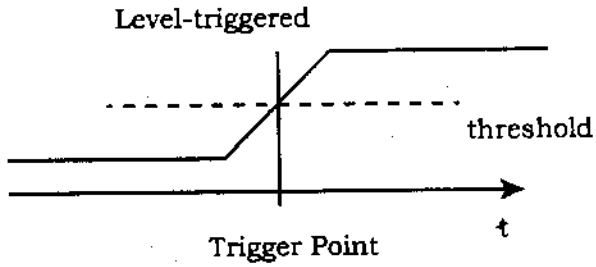
TCON রেজিস্টারের IT0 ও IT1 এর মাধ্যমে Low level activated ইন্টারাপ্ট বা negative-edge activated ইন্টারাপ্ট নির্বাচন প্রোগ্রাম করা হয়। যদি IT1 = 0 হয়, তবে $\overline{INT1}$ পিনে একটি Low level দ্বারা External interrupt 1 এ সূত্রপাত হয়। যদি IT1 = 1 হয়, তবে External interrupt 1 edge-triggered হয়। এই মুডে যদি $\overline{INT1}$ পিনে এক সাইকেলে High এবং পরবর্তী সাইকেলে Low দেখায় তবে TCON রেজিস্টারের IE1 flag বিটটি সেট হয় এবং এর Interrupt request করে। যেহেতু বাহ্যিক

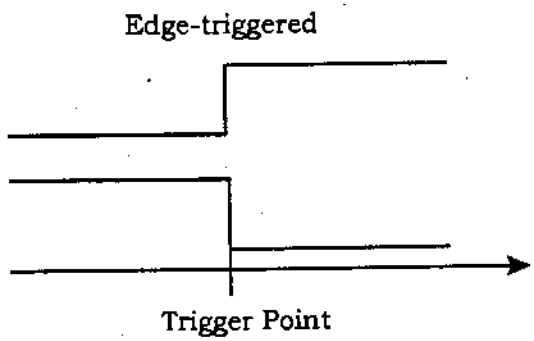
ইন্টারাপ্ট এক মেশিন সাইকেলে একবার স্যাম্পল হয়, তাই সঠিক স্যাম্পলিং নিশ্চিতকরণের জন্য ন্যূনতম 12 ক্লক পিরিয়ডে একবার ইনপুট হওয়া উচিত। যদি বাহ্যিক ইন্টারাপ্ট Transition-activated হয়, তবে বাহ্যিক উৎস request pin কে অন্তত এক সাইকেলের জন্য অবশ্যই high রাখবে এবং পরে Transition শনাক্তকরণ নিশ্চিতকরণের জন্য আরও এক সাইকেলের জন্য request পিনকে Low রাখবে। যখন CPU ইন্টারাপ্টকে ভেক্টর করে তখন IE0 ও IE1 স্বয়ংক্রিয়ভাবে Clear হয়।

যদি বাহ্যিক ইন্টারাপ্ট level-activated হয়, তবে বাহ্যিক উৎস অবশ্যই request কে কার্যকরী রাখবে যতক্ষণ পর্যন্ত না requested interrupt প্রকৃতভাবে উৎপন্ন হয়। পরে ISR শেষ হওয়ার পূর্বে একে অবশ্যই নিষ্ক্রিয় করতে হবে নতুবা অন্য ইন্টারাপ্ট উৎপন্ন হবে। সচরাচর ISR এর একটি ক্রিয়া ইন্টারাপ্ট উৎসকে ইন্টারাপ্টিং সিগন্যালের নিষ্ক্রিয় অবস্থায় নিয়ে যায়।

**টাইমার ইন্টারাপ্ট (Timer interrupt) :** 8051 মাইক্রোকন্ট্রোলারের T0 ও T1 নামের 16 বিটের দুটি টাইমার/কাউন্টার রয়েছে। এরা টাইম ডিলে উৎপন্ন করার জন্য বা মাইক্রোকন্ট্রোলারে বহিঃস্থ ঘটনা (event) এর কাউন্টার বা event counter হিসেবে ব্যবহার করা হয়। যেহেতু 8051 এর আর্কিটেকচার 8 বিটের, তাই 16 বিটের টাইমার access করার জন্য প্রত্যেকের জন্য দুটি করে 8 বিটের রেজিস্টার রয়েছে। এরা হলো TL0, TH0, TL1 ও TH1। প্রত্যয় 0 ও 1 যথাক্রমে Timer 0 ও Timer 1 প্রকাশ করে।

উভয় টাইমার বিভিন্ন টাইমার অপারেশন মুড সেটিং এর জন্য একই রেজিস্টার TMOD ব্যবহার করে। TMOD একটি 8 বিটের রেজিস্টার, যার নিচের 4 বিট Timer 0 এবং উপরের 4 বিট Timer1 এর জন্য বরাদ্দকৃত। টাইমার অপারেশনকে নিয়ন্ত্রণের জন্য অপর একটি বিশেষ রেজিস্টার TCON ব্যবহার করা হয়।

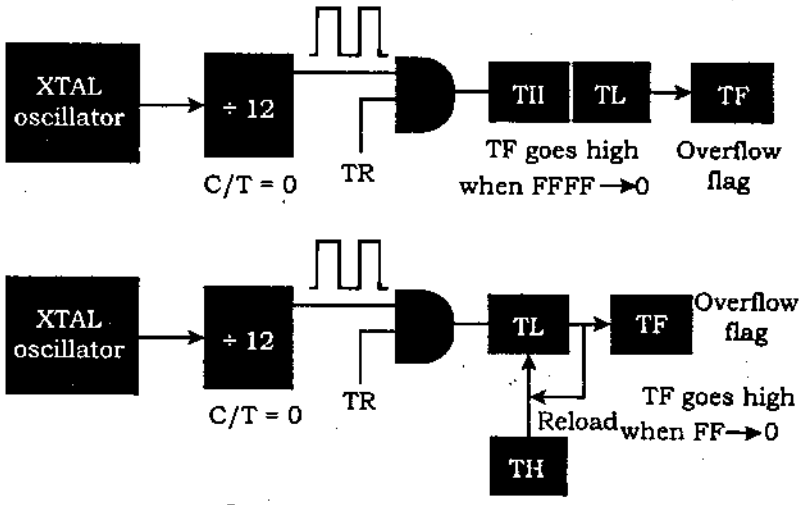



চিত্র-১১.২ : Level-trigger ও Edge-trigger

8051 টাইমার সর্বদা count up করে। যখন তাদের কাউন্ট সর্বোচ্চ কাউন্ট থেকে 0000 আবর্তন করে, তখন TCON রেজিস্টারে তাদের টাইমার flag (TF0 বা TF1) উৎপন্ন করে। ব্যবহারকারীর প্রোগ্রাম দ্বারা TCON

রেজিস্টারের টাইমার Run flag বিট TR0 বা TR1 সেট করে টাইমার অপারেশন শুরু করা হয়। TF0 বা TF1 সেট হলেই টাইমার ইন্টারাপ্ট ঘটে।




চিত্র-১১.৩ : টাইমার অপারেশন

**সিরিয়াল ইন্টারাপ্ট (Serial interrupt) :**

যখন সিরিয়াল পোর্ট কর্তৃক একটি বাইট গৃহীত হয়, তখন RI ইন্টারাপ্ট flag এবং পোর্ট থেকে এক বাইট প্রেরিত হলে তখন TI ইন্টারাপ্ট flag সেট হয়। RI ও TI এর যে কোনটি সেট হলে একটি সিরিয়াল ইন্টারাপ্ট Trigger হয়। Transmit interrupt (TI) হয় যখন SBUF রেজিস্টারে প্রেরিতব্য অক্ষরের লিখন শেষ হয় এবং Receive interrupt হয় যখন কোন অক্ষর সম্পূর্ণভাবে গৃহীত হয় এবং SBUF রেজিস্টারে পঠনের জন্য অপেক্ষা করে। সিরিয়াল পোর্ট ইন্টারাপ্ট, সিরিয়াল ইন্টারাপ্ট অন্যান্য ইন্টারাপ্টের চেয়ে সামান্য ভিন্ন, এর Flag হার্ডওয়্যার দ্বারা clear হয় না। কারণ একটি সিরিয়াল পোর্ট ইন্টারাপ্টের জন্য দুটি উৎস RI ও TI। ইন্টারাপ্টের উৎস অবশ্যই ISR এ নির্ধারণ করতে হয় এবং Interrupting flag সফটওয়্যার দ্বারা clear করা হয়।

## ১১.৯ ইন্টারাপ্টের সাধারণ সমস্যা বর্ণনা (State the Common Problem With Interrupt)

ইন্টারাপ্ট একটি শক্তিশালী হাতিয়ার হিসেবে 8051 প্রোগ্রাম উন্নয়ন করে কিন্তু এর অশুদ্ধ ব্যবহার প্রোগ্রামে ব্যাপক ত্রুটি আসে, যা নির্ণয় ও সংশোধন খুবই কষ্টসাধ্য। যদি প্রোগ্রামে ইন্টারাপ্ট থাকে এবং প্রোগ্রাম ক্রাশ করে বা আকাঙ্ক্ষিত কার্য সম্পাদন না করে, তবে নিম্নবর্ণিত ইন্টারাপ্ট-সম্পর্কিত বিষয়গুলো পর্যালোচনা করা হয় :

(১) **রেজিস্টার সুরক্ষাকরণ (Register protection)**

প্রোগ্রামে রেজিস্টার সুরক্ষা নিশ্চিত হতে হবে। যদি প্রোগ্রামে রেজিস্টার সুরক্ষা করা না হয়, তবে অনাকাঙ্ক্ষিত ফলাফল হতে পারে।

(২) **সুরক্ষিত মান পূর্বাবস্থায় ফিরিয়ে আনতে ভুলে যাওয়া (Forgetting to restore protected value) :**

অপর এক সাধারণ সমস্যা হলো রেজিস্টারসমূহকে Stack এ Push করা এবং ইন্টারাপ্ট থেকে বের হওয়ার পূর্বে তা Stack থেকে POP করতে ভুলে যাওয়া। ফলে একটি অতিরিক্ত মান Stack এ থেকে যায়। যখন RETI নির্দেশনা সম্পাদন হয়, তখন প্রত্যাবর্তন ঠিকানা সঠিক হলেও 8051 এই মানকে ব্যবহার করে। এতে প্রোগ্রাম প্রায়ই নিশ্চিত ক্রাশ করে।

(৩) **RETI এর পরিবর্তে RET এর ব্যবহার :**

ইন্টারাপ্ট সবসময় RETI নির্দেশনা দ্বারা শেষ হয়। RET নির্দেশনা ইন্টারাপ্টের শেষ করে না। RETI এর পরিবর্তে RET ব্যবহার করলে মূল প্রোগ্রাম স্বাভাবিকভাবে চলছে মনে হয়, কিন্তু ইন্টারাপ্ট শুধুমাত্র একবার সম্পাদিত হতে পারে এবং ইন্টারাপ্ট সম্পাদন হঠাৎ করে থেমে যেতে পারে।

# অনুশীলনী-১১

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **8051 মাইক্রোকন্ট্রোলারের বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের কাজ কী?** *[বাকাশিবো : '১৫]*

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের কাজ হলো ইন্টারাপ্টের ফলে উৎপন্ন flag bit হতে সমস্যা গুলো Clear করা।

২। **TI ও RI কখন সেট হয়?** *[বাকাশিবো : '১৪]*

**উত্তর :** যখন সিরিয়াল পোর্ট কর্তৃক একটি বাইট গৃহীত হয়, তখন RI ইন্টারাপ্ট flag এবং পোর্ট থেকে এক বাইট প্রেরিত হলে তখন TI ইন্টারাপ্ট flag সেট হয়।

৩। **Interrupt service routine (ISR) বা Interrupt handler বলতে কী বুঝায়?**

**উত্তর :** চলমান প্রোগ্রামে কোন ইন্টারাপ্ট সংঘটিত হলে যে ছোট প্রোগ্রাম প্রসেসর বা কন্ট্রোলারকে কী করতে হবে তা অবগত করে তাকে Interrupt service routine (ISR) বলে। একে Interrupt handler-ও বলে। ISR ইন্টারাপ্টের প্রত্যুত্তরে সম্পাদিত হয় এবং সাধারণত একটি ডিভাইসের ইনপুট/আউটপুট কাজ সমাধা করে।

৪। **ইন্টারাপ্ট ভেক্টর ও ইন্টারাপ্ট ভেক্টর টেবিল কাকে বলে?**

**উত্তর :** প্রতিটি ইন্টারাপ্ট-এর জন্য মেমোরিতে নির্দিষ্ট লোকেশন থাকে, যা ISR এর অ্যাড্রেস ধারণ করে একে ভেক্টর ঠিকানা বলে। ISR এর অ্যাড্রেসকে ধারণ করার জন্য মেমোরি লোকেশনের গ্রুপকে ইন্টারাপ্ট ভেক্টর টেবিল বলে।

৫। **IE রেজিস্টারের কাজ কী?**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের বিভিন্ন ইন্টারাপ্টের নিয়ন্ত্রণের জন্য IE register ব্যবহৃত হয়। এটি 8 বিটের একটি বিশেষ রেজিস্টার (SFR), যা অভ্যন্তরীণ মেমোরির A8h ঠিকানায় সংরক্ষিত।

৬। **Timer 1 ইন্টারাপ্টকে কার্যকর করার জন্য কোড লিখ।**

**উত্তর :** Timer 1 ইন্টারাপ্টকে কার্যকর করার জন্য কোড নিম্নরূপ :

```
SETB ET1                ; Enable Timer 1 interrupt
SETB EA                 ; Set Global enable bit
```

বা

```
MOV IE, # 10001000B     ; Set EA And ET1 of IE register.
```

৭। **8051 বাহ্যিক ইন্টারাপ্ট কখন ঘটে?**

**উত্তর :** 8051 এর $\overline{INT0}$ (পিন 12) বা $\overline{INT1}$ (পিন 13) পিনের একটি Low level বা Negative edge এর ফলে বাহ্যিক ইন্টারাপ্ট ঘটে।

৮। **বাহ্যিক ইন্টারাপ্টের flag bit গুলোর নাম লিখ।**

**উত্তর :** বাহ্যিক ইন্টারাপ্টের ফলে উৎপন্ন flag bit গুলো হলো যথাক্রমে IE0 ও IE1.

৯। **টাইমার ইন্টারাপ্টের flag bit গুলোর নাম লিখ।**

**উত্তর :** টাইমার ইন্টারাপ্টের Overflow bit গুলোর নাম TF0 ও TF1.

১০। **সিরিয়াল ইন্টারাপ্টের flag bit গুলোর নাম লিখ।**

**উত্তর :** সিরিয়াল ইন্টারাপ্টের flag বিটগুলোর নাম TI ও RI.

**১১। ইন্টারাপ্ট রুটিনে রেজিস্টারের মান সুরক্ষার জন্য কী নিয়ম অনুসরণ করা হয়?**

**উত্তর :** সমস্ত Interrupt handler বা রুটিন এ একটি অত্যন্ত গুরুত্বপূর্ণ নিয়ম প্রয়োগ করা হয়, তা হলো : ইন্টারাপ্ট সূচিত হওয়ার সময় যে অবস্থায় ছিল ত্যাগ করার সময় প্রসেসরকে অবশ্যই একই অবস্থায় থাকতে হবে। অর্থাৎ যদি কোন ইন্টারাপ্ট রেজিস্টার (যথা : একুমুলেটর) ব্যবহার করে, তবে এটি অবশ্যই নিশ্চিত করতে হবে যে, রেজিস্টারের মান ইন্টারাপ্টের শুরুতে যা ছিল শেষেও তাই থাকে।

**১২। ইন্টারাপ্ট রুটিনে PSW রেজিস্টারকে সুরক্ষার প্রয়োজনীয়তা এবং উপায় কী?**

**উত্তর :** PSW রেজিস্টারের অনেক স্বতন্ত্র বিট আছে, যাদেরকে 8051 এর বিভিন্ন নির্দেশনা দ্বারা সেট করা যায়। সেক্ষেত্রে ইন্টারাপ্টের শুরুতে এবং শেষে Stack এ Pushing ও Popping করে PSW রেজিস্টারকে সর্বদা সুরক্ষা করা একটি ভাল পরিকল্পনা।

**১৩। ইন্টারাপ্ট রুটিন চলাকালীন রেজিস্টার সুরক্ষার পদ্ধতি বর্ণনা কর।**

**উত্তর :** ইন্টারাপ্ট রুটিনকে নিম্নবর্ণিত রেজিস্টারসমূহকে অবশ্যই সুরক্ষা করতে হবে–

1. PSW
2. PSW
3. B
4. DPTR (DPH/DPL)
5. ACC
6. Registers R0-R7

এখানে, PSW রেজিস্টারে অনেক স্বতন্ত্র বিট আছে, যাদেরকে 8051 এর বিভিন্ন নির্দেশনা দ্বারা সেট করা যায়। সেক্ষেত্রে ইন্টারাপ্টের শুরুতে এবং শেষে Stack এ Pushing ও Popping করে PSW রেজিস্টারকে সর্বদা সুরক্ষা করা একটি উত্তম পরিকল্পনা।

**১৪। বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের Flag বিটগুলোর নাম কী?**

**উত্তর :** বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের Flag বিটগুলোর নাম হলো : IE0 ও IE1

**১৫। বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টকে কয়ভাবে সচল করা যায় ও কী কী?**

**উত্তর :** বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টকে দুইভাবে trigger করা যায়, যথা– ১. Low Level Trigger ২. Negative Edge Trigger.

**১৬। টাইমার ইন্টারাপ্টের flag বিটগুলোর নাম লিখ।**

**উত্তর :** টাইমার ইন্টারাপ্টের দুই টাইমারের জন্য দুটি overflow flag TF0 ও TF1 আছে।

**১৭। ISR এর সংক্ষিপ্ত নাম কী?**

**উত্তর :** ইন্টারাপ্ট সার্ভিস রুটিন (Interrupt Service Routing) এর সংক্ষিপ্ত নাম হলো ISR।

**১৮। ISR এর ভেক্টর ঠিকানা বলতে কী বোঝায়?**

**উত্তর :** প্রতিটি ইন্টারাপ্ট এর জন্য মেমোরিতে নির্দিষ্ট লোকেশন থাকে, যা ISR এর অ্যাড্রেস ধারণ করে একে ভেক্টর ঠিকানা বলে।

## সংক্ষিপ্ত প্রশ্নোত্তর

**১। একটি ইন্টারাপ্ট সম্পাদনের ধাপগুলো লিখ।** *[বাকাশিবো : '১৪, '১৪R]*

**উত্তর :** একটি ইন্টারাপ্টের সক্রিয়তার মাইক্রোকন্ট্রোলার নিচের ধাপগুলো অনুসরণ করে :

১। এটি তার চলমান নির্দেশনা সম্পাদনা শেষ করে।

২। এটি প্রোগ্রাম কাউন্টারে রক্ষিত পরবর্তী নির্দেশনার ঠিকানা stack এ জমা করে।

৩। এটি ইন্টারাপ্টের বর্তমান অবস্থা (status) অভ্যন্তরীণভাবে (stack এ নয়) জমা রাখে।

৪। এটি Interrupt vector location থেকে ISR এর ঠিকানা নেয় এবং তাতে জাম্প করে। মাইক্রোকন্ট্রোলার ISR সম্পাদন শুরু করে যতক্ষণ পর্যন্ত না সাবরুটিনের শেষ নির্দেশনা (RETI, Return from interrupt) আসে।

৫। RETI সম্পাদন শেষ হলে মাইক্রোকন্টোলার তার পূর্বের অবস্থানে (যেখান থেকে ইন্টারাপ্ট শুরু হয়) ফিরে আসে। এরপর প্রথমে এটি Stack থেকে প্রথম দুই বাইট প্রোগ্রাম কাউন্টারে POP করে পরবর্তী সম্পাদিত নির্দেশনার ঠিকানা নেয়। তারপর ঐ ঠিকানা থেকে সম্পাদন শুরু করে।

২। **8051 মাইক্রোকন্ট্রোলারের ইন্টারাপ্ট উৎস কয়টি ও কী কী? এদের flag বিটগুলোর নাম লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারে পাঁচটি ইন্টারাপ্ট উৎস আছে। যেহেতু প্রধান RESET কেও একটি ইন্টারাপ্ট উৎস হিসেবে বিবেচনা করা হয়, তাই 8051 এর জন্য ছয়টি ইন্টারাপ্ট উৎসের তালিকা উল্লেখ করা হলো :

| No | Interrupt | Flag |
|---|---|---|
| 1. | System RESET | RST |
| 2. | External interrupt 0 | IE0 |
| 3. | Timer/Counter 0 | TF0 |
| 4. | External interrupt | TF1 |
| 5. | Timer/Counter 1 | TE1 |
| 6. | Serial port | RI বা TI |

৩। **Interrupt service routine (ISR) বা Interrupt handler এর কাজ কী উল্লেখ কর।**

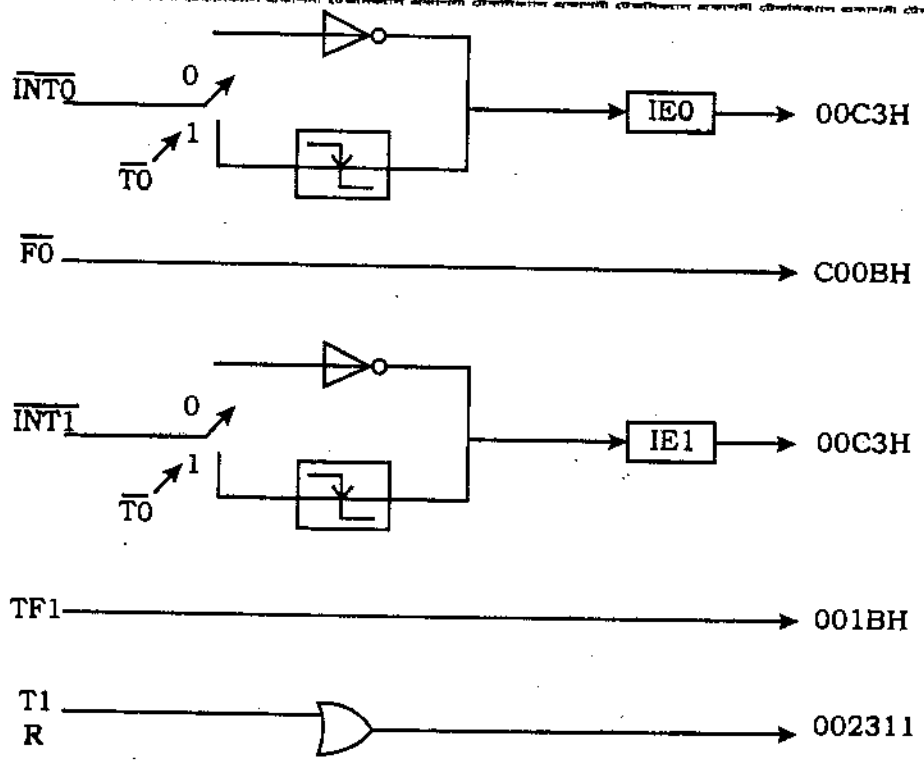
**অথবা, 8051 এর ISR ব্যাখ্যা কর।**

**উত্তর :** ইন্টারাপ্ট সার্ভিস রুটিন (Interrupt Service Routing) এর সংক্ষিপ্ত নাম হলো ISR। প্রতিটি ইন্টারাপ্ট এর জন্য একটি করে Interrupt Service Routine (ISR) থাকা আবশ্যক। প্রতিটি ইন্টারাপ্টের জন্য মেমোরিতে নির্দিষ্ট লোকেশন থাকে, যা ISR এর অ্যাড্রেস ধারণ করে একে ভেক্টর ঠিকানা বলে। ISR এর অ্যাড্রেসকে ধারণ করার জন্য মেমোরি লোকেশনের গ্রুপকে ইন্টারাপ্ট ভেক্টর টেবিল বলে। চলমান প্রোগ্রামে কোন ইন্টারাপ্ট সংঘটিত হলে যে ছোট প্রোগ্রাম প্রসেসর বা কন্ট্রোলারকে কী করতে হবে তা অবগত করে, তাকে Interrupt Service Routine (ISR) বলে। এটিকে Interrupt handler-ও বলে। ISR ইন্টারাপ্টের প্রত্যুত্তরে সম্পাদিত হয় এবং সাধারণত একটি ডিভাইসের ইনপুট/আউটপুট কাজ সমাধা করে। যখন কোন ইন্টারাপ্ট সংঘটিত হয়, তখন মূল প্রোগ্রাম সম্পাদন সাময়িক স্থগিত হয় এবং প্রোগ্রামকে ISR এ branching করে। ISR সম্পাদিত হয় এবং RETI নির্দেশনার মাধ্যমে সমাপ্ত হয়। ইন্টারাপ্ট সম্পাদন শেষে মূল প্রোগ্রাম (যেখান থেকে ইন্টারাপ্ট শুরু) পুনরায় চলতে থাকে। যখন কোন ইন্টারাপ্ট গৃহীত হয়, তখন PC (Program counter) তে যে মান জমা হয়, তাকে Interrupt vector বলে। এটি ইন্টারাপ্ট উৎসের জন্য ISR এর শুরুর ঠিকানা।

৪। **8051 এর ইন্টারাপ্টের অগ্রগণ্যতা বা প্রাইওরিটি ও ভেক্টর ঠিকানা উল্লেখ কর।**

**উত্তর :** যখন 8051 এ পাওয়ার প্রয়োগ করা হয়, তখন এর ইন্টারাপ্টসমূহের অগ্রগণ্যতা নিচের তালিকা অনুসারে হয় :

| ইন্টারাপ্ট উৎস (Interrupt source) | ফ্ল্যাগ (Flag) | অগ্রগণ্যতা স্তর (Priority level) | ভেক্টর ঠিকানা (Vector address) |
|---|---|---|---|
| External interrupt 0 | IEO | Highest | 0003h |
| Timer/counter 0 | TF0 | ↓ | 000Bh |
| External interrupt 1 | IE1 | ↓ | 0013h |
| Timer/counter 1 | TF1 | ↓ | 001Bh |
| Serial port | RI or TI | Lowest | 0023h |




চিত্র : 8051 এর ইন্টারাপ্ট ও ভেক্টর ঠিকানা

৫। **ইন্টারাপ্টের সাধারণ সমস্যাগুলো বর্ণনা কর।**

**উত্তর :** ইন্টারাপ্ট একটি শক্তিশালী হাতিয়ার হিসেবে 8051 প্রোগ্রাম উন্নয়ন করে কিন্তু এর অশুদ্ধ ব্যবহার প্রোগ্রামে ব্যাপক ত্রুটি আসে, যা নির্ণয় ও সংশোধন খুবই কষ্টসাধ্য। যদি প্রোগ্রামে ইন্টারাপ্ট থাকে এবং প্রোগ্রাম ক্রাশ করে বা আকাঙ্ক্ষিত কার্য সম্পাদন না করে, তবে নিম্নবর্ণিত ইন্টারাপ্ট-সম্পর্কিত বিষয়গুলো পর্যালোচনা করা হয় :

**(১) রেজিস্টার সুরক্ষাকরণ (Register protection)**

প্রোগ্রামে রেজিস্টার সুরক্ষা নিশ্চিত হতে হবে। যদি প্রোগ্রামে রেজিস্টার সুরক্ষা করা না হয়, তবে অনাকাঙ্ক্ষিত ফলাফল হতে পারে।

**(২) সুরক্ষিত মান পূর্বাবস্থায় ফিরিয়ে আনতে ভুলে যাওয়া (Forgetting to restore protected value) :**

অপর এক সাধারণ সমস্যা হলো রেজিস্টারসমূহকে Stack এ Push করা এবং ইন্টারাপ্ট থেকে বের হওয়ার পূর্বে তা Stack থেকে POP করতে ভুলে যাওয়া। ফলে একটি অতিরিক্ত মান Stack এ থেকে যায়। যখন RETI নির্দেশনা সম্পাদন হয়, তখন প্রত্যাবর্তন ঠিকানা সঠিক হলেও 8051 এই মানকে ব্যবহার করে। এতে প্রোগ্রাম প্রায়ই নিশ্চিত ক্রাশ করে।

**(৩) RETI এর পরিবর্তে RET এর ব্যবহার :**

ইন্টারাপ্ট সবসময় RETI নির্দেশনা দ্বারা শেষ হয়। RET নির্দেশনা ইন্টারাপ্টের শেষ করে না। RETI এর পরিবর্তে RET ব্যবহার করলে মূল প্রোগ্রাম স্বাভাবিকভাবে চলছে মনে হয়, কিন্তু ইন্টারাপ্ট শুধুমাত্র একবার সম্পাদিত হতে পারে এবং ইন্টারাপ্ট সম্পাদন হঠাৎ করে থেমে যেতে পারে।

৬। **ইন্টারাপ্টকে কার্যকর বা অকার্যকর করার প্রক্রিয়া বর্ণনা কর।**

**উত্তর :** ইন্টারাপ্টকে কার্যকর বা অকার্যকর করার প্রক্রিয়া বর্ণনা করা হলো :

কার্যকর (masking) বা অকার্যকর (limasking) করার জন্য নিচের ধাপগুলো অনুসরণ করা হয় :

১। IE রেজিস্টারের D7 বিট (EA) কে অবশ্যই High তে সেট করতে হবে, যাতে রেজিস্টারের অন্যান্য ইন্টারাপ্টগুলো কার্যকর হয়।

২। EA এর মান নির্ধারণ।

যদি EA = 1 হয়, সব ইন্টারাপ্ট কার্যকর হয় এবং IE রেজিস্টারে সংশ্লিষ্ট বিট High থাকে।

যদি EA = 0 হয়, তবে সংশ্লিষ্ট বিট High থাকলেও কোন ইন্টারাপ্টই কার্যকর হবে না।

৩। এবার কোন ইন্টারাপ্টকে কার্যকর করার জন্য ঐ ইন্টারাপ্ট উৎসের Enable bit সেট করতে হয়।

উদাহরণস্বরূপ Timer 1 ইন্টারাপ্টকে কার্যকর করার জন্য কোড নিম্নরূপ :

```
SETB ET1                    ; Enable Timer 1 Interrupt
SETB EA                     ; Set Global enable bit
বা, MOV IE, # 10001000B     ; Set EA and ET1 or IE register.
```

৭। **বাহ্যিক হার্ডওয়্যার ইন্টারাপ্ট বর্ণনা কর।**

**উত্তর :** 8051 এর $\overline{INT0}$ (পিন 12) বা $\overline{INT1}$ (পিন 13) পিনের একটি Low level বা Negative edge এর ফলে বাহ্যিক ইন্টারাপ্ট ঘটে। এই ইন্টারাপ্টের ফলে উৎপন্ন flag bit গুলো হলো : যথাক্রমে IE0 ও IE1 (TCON) রেজিস্টার) এবং যদি ইন্টারাপ্ট Transition activated (Negative edge) হয়, এদেরকে হার্ডওয়্যার দ্বারা clear করা হয়। যদি ইন্টারাপ্ট Level-addtivated হয়, তবে মাইক্রোকন্ট্রোলারের অভ্যন্তরীণ হার্ডওয়্যারের পরিবর্তে বাহ্যিক ইন্টারাপ্ট উৎস Request flag এর লেভেলকে নিয়ন্ত্রণ করে।

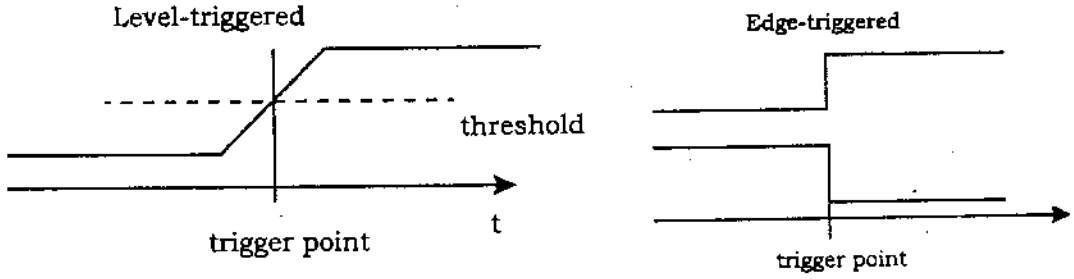
TCON রেজিস্টারের IT0 ও IT1 এর মাধ্যমে Low level activated ইন্টারাপ্ট বা negative-edge activated ইন্টারাপ্ট নির্বাচন প্রোগ্রাম করা হয়। যদি IT1 = 0 হয়, তবে $\overline{INT1}$ পিনে একটি Low level দ্বারা External interrupt 1 এ সূত্রপাত হয়। যদি IT1 = 1 হয়, তবে External interrupt 1 edge-triggered হয়। এই মুডে যদি $\overline{INT1}$ পিনে এক সাইকেলে High এবং পরবর্তী সাইকেলে Low দেখায় তবে TCON রেজিস্টারের IE1 flag বিটটি সেট হয় এবং এর Interrupt request করে। যেহেতু বাহ্যিক ইন্টারাপ্ট এক মেশিন সাইকেলে একবার স্যাম্পল হয়, তাই সঠিক স্যাম্পলিং নিশ্চিতকরণের জন্য ন্যূনতম 12 ক্লক পিরিয়ডে একবার ইনপুট হওয়া উচিত। যদি বাহ্যিক ইন্টারাপ্ট Transition-activated হয়, তবে বাহ্যিক উৎস request pin কে অন্তত এক সাইকেলের জন্য অবশ্যই high রাখবে এবং পরে Transition শনাক্তকরণ নিশ্চিতকরণের জন্য আরও এক সাইকেলের জন্য request পিনকে Low রাখবে। যখন CPU ইন্টারাপ্টকে ভেক্টর করে তখন IE0 ও IE1 স্বয়ংক্রিয়ভাবে Clear হয়।

যদি বাহ্যিক ইন্টারাপ্ট level-activated হয়, তবে বাহ্যিক উৎস অবশ্যই request কে কার্যকরী রাখবে যতক্ষণ পর্যন্ত না requested interrupt প্রকৃতভাবে উৎপন্ন হয়। পরে ISR শেষ হওয়ার পূর্বে একে অবশ্যই নিষ্ক্রিয় করতে হবে নতুবা অন্য ইন্টারাপ্ট উৎপন্ন হবে। সচরাচর ISR এর একটি ক্রিয়া ইন্টারাপ্ট উৎসকে ইন্টারাপ্টিং সিগন্যালের নিষ্ক্রিয় অবস্থায় নিয়ে যায়।

৮। **টাইমার ইন্টারাপ্ট বর্ণনা কর।**

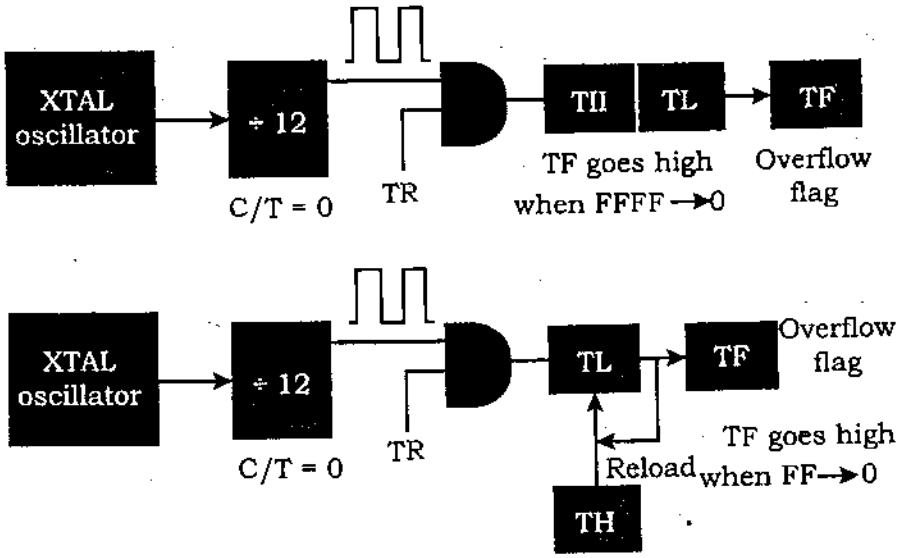
**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের T0 ও T1 নামের 16 বিটের দুটি টাইমার/কাউন্টার রয়েছে। এরা টাইম ডিলে উৎপন্ন করার জন্য বা মাইক্রোকন্ট্রোলারে বহিঃস্থ ঘটনা (event) এর কাউন্টার বা event counter হিসেবে ব্যবহার করা হয়। যেহেতু 8051 এর আর্কিটেকচার 8 বিটের, তাই 16 বিটের টাইমার access করার জন্য প্রত্যেকের জন্য দুটি করে 8 বিটের রেজিস্টার রয়েছে। এরা হলো TL0, TH0, TL1 ও TH1। প্রত্যয় 0 ও 1 যথাক্রমে Timer 0 ও Timer 1 প্রকাশ করে।

উভয় টাইমার বিভিন্ন টাইমার অপারেশন মুড সেটিং এর জন্য একই রেজিস্টার TMOD ব্যবহার করে। TMOD একটি 8 বিটের রেজিস্টার, যার নিচের 4 বিট Timer 0 এবং উপরের 4 বিট Timer1 এর জন্য বরাদ্দকৃত। টাইমার অপারেশনকে নিয়ন্ত্রণের জন্য অপর একটি বিশেষ রেজিস্টার TCON ব্যবহার করা হয়।




**চিত্র :** Level-trigger ও Edge-trigger

8051 টাইমার সর্বদা count up করে। যখন তাদের কাউন্ট সর্বোচ্চ কাউন্ট থেকে 0000 আবর্তন করে, তখন TCON রেজিস্টারে তাদের টাইমার flag (TF0 বা TF1) উৎপন্ন করে। ব্যবহারকারীর প্রোগ্রাম দ্বারা TCON রেজিস্টারের টাইমার Run flag বিট TR0 বা TR1 সেট করে টাইমার অপারেশন শুরু করা হয়। TF0 বা TF1 সেট হলেই টাইমার ইন্টারাপ্ট ঘটে।




**চিত্র :** টাইমার অপারেশন

**৯। ইন্টারাপ্ট চলাকালীন কোন কোন রেজিস্টারকে সুরক্ষা করতে হয়?**

**উত্তর :** ইন্টারাপ্ট রুটিনকে নিম্নবর্ণিত রেজিস্টারসমূহকে অবশ্যই সুরক্ষা করতে হবে :

1. PSW
2. DPTR (DPH/DPL)
3. PSW
4. ACC
5. B
6. Registers R0-R7.

**১০। IE রেজিস্টারের বিটগুলোর কাজ লিখ।**

**উত্তর :** 8051 মাইক্রোকন্ট্রোলারের বিভিন্ন ইন্টারাপ্টের নিয়ন্ত্রণের জন্য IE register ব্যবহার করা হয়। এটি একটি 8 বিটের বিশেষ রেজিস্টার (SFR), যা অভ্যন্তরীণ মেমোরির A8h ঠিকানায় সংরক্ষিত। এটি একটি bit-addressable রেজিস্টার। নিচে এর বিটগুলো সম্পর্কে সংক্ষেপে বর্ণনা করা হলো :

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | X | 0 | 0 | 0 | 0 | 0 | 0 | Value after reset |
| IE | EA | | ET2 | ES | ET1 | EX1 | ET0 | EX0 | Bit name |
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | |

EA – Global interrupt enable/disable :

0- সমস্ত interrupt requests অকার্যকর করে।

1- সমস্ত interrupt requests কার্যকর করে।

ES– Enables or disables serial interrupt :

0- UART system কোন interrupt উৎপন্ন করতে পারে না।

1- UART system interrupt উৎপন্ন করতে পারে।

ET1– Bit enables or disables Timer 1 interrupt :

0- Timer 1 কোন interrupt উৎপন্ন করতে পারে না।

1- Timer interrupt উৎপন্ন করতে পারে।

EX1– Bit enables or disables external 1 interrupt :

0- Pin INT1 logic state এর পরিবর্তনে কোন interrupt উৎপন্ন করতে পারে না।

1- Pin INT 1 logic state এর পরিবর্তনে interrupt উৎপন্ন করতে পারে।

ET0– Bit enables or disables timer 0 interrupt :

0- Timer 0 কোন interrupt উৎপন্ন করতে পারে না।

1- Timer 0 interrupt উৎপন্ন করতে পারে।

EX0– Bit enables or disables external 0 interrupt :

0- Pin INT0 logic state এর পরিবর্তনে কোন interrupt উৎপন্ন করতে পারে না।

1- Pin INT0 logic state এর পরিবর্তনে interrupt উৎপন্ন করতে পারে।

**১১। 8051 এর সিরিয়াল ইন্টারাপ্ট বর্ণনা কর।**

**উত্তর :** যখন সিরিয়াল পোর্ট কর্তৃক একটি বাইট গৃহীত হয়, তখন RI ইন্টারাপ্ট flag এবং পোর্ট থেকে এক বাইট প্রেরিত হলে তখন TI ইন্টারাপ্ট flag সেট হয়। RI ও TI এর যে কোনটি সেট হলে একটি সিরিয়াল ইন্টারাপ্ট Trigger হয়। Transmit interrupt (TI) হয় যখন SBUF রেজিস্টারে প্রেরিতব্য অক্ষরের লিখন শেষ হয় এবং Receive interrupt হয় যখন কোন অক্ষর সম্পূর্ণভাবে গৃহীত হয় এবং SBUF রেজিস্টারে পঠনের জন্য অপেক্ষা করে। সিরিয়াল পোর্ট ইন্টারাপ্ট, সিরিয়াল ইন্টারাপ্ট অন্যান্য ইন্টারাপ্টের চেয়ে সামান্য ভিন্ন, এর Flag হার্ডওয়্যার দ্বারা clear হয় না। কারণ একটি সিরিয়াল পোর্ট ইন্টারাপ্টের জন্য দুটি উৎস RI ও TI। ইন্টারাপ্টের উৎস অবশ্যই ISR এ নির্ধারণ করতে হয় এবং Interrupting flag সফটওয়্যার দ্বারা clear করা হয়।

## রচনামূলক প্রশ্নোত্তর

**১। 8051 এর বাহ্যিক হার্ডওয়্যার টাইমার ও সিরিয়াল কমিউনিকেশন ইন্টারাপ্ট বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১১.৮ নং দ্রষ্টব্য।

**২। ইন্টারাপ্ট সার্ভিস রুটিন বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১১.২ নং দ্রষ্টব্য।

**৩। ইন্টারাপ্ট অগ্রগণ্যতা ও ভেক্টর অবস্থান উল্লেখ কর।**

**উত্তর :** অনুচ্ছেদ ১১.৩ নং দ্রষ্টব্য।

**৪। ইন্টারাপ্ট এনাবল রেজিস্টার IE বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১১.৪ নং দ্রষ্টব্য।

**৫। ইন্টারাপ্ট কার্যকর ও অকার্যকর করার প্রক্রিয়া বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১১.৫ নং দ্রষ্টব্য।

**৬। ইন্টারাপ্টের সময় রেজিস্টার সুরক্ষাকরণ বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১১.৭ নং দ্রষ্টব্য।

**৭। ইন্টারাপ্টের সাধারণ সমস্যা বর্ণনা কর।**

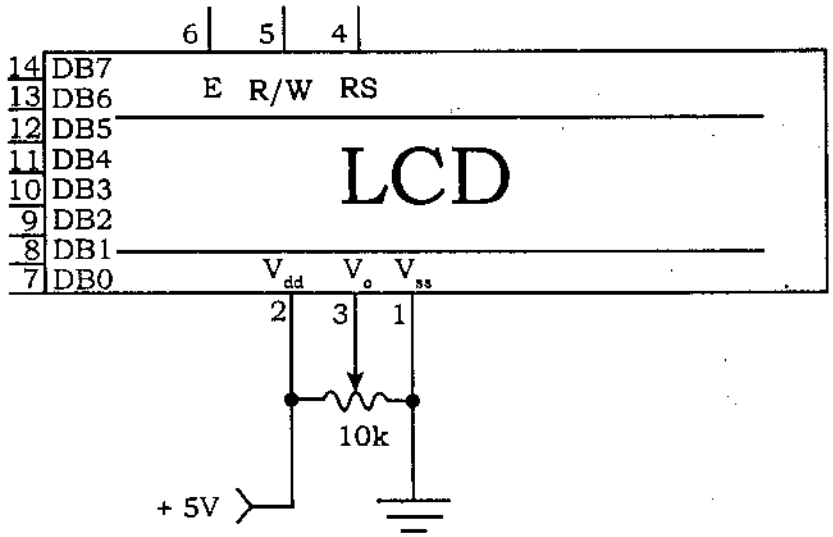
**উত্তর :** অনুচ্ছেদ ১১.৯ নং দ্রষ্টব্য।

# অধ্যায় ১২ এলসিডি ও কী-বোর্ড ইন্টারফেসিং
# Under Stand LCD and Keyboard Interfacing

## ১২.১ এলসিডি এর পিন ডায়াগ্রাম বর্ণনা

**(Describe the Pin Diagram of LCD)**

আধুনিক যুগে LCD মনিটর, টেলিভিশন ব্যাপক সমাদৃত। LCD (Liquid Crystal Display) পর্দা একটি বহুল ব্যবহৃত ইলেকট্রনিক ডিসপ্লে মডিউল। 16 × 2 LCD ডিসপ্লে একটি অত্যন্ত মৌলিক মডিউল, যা বিভিন্ন ডিভাইস ও সার্কিটে সাধারণত অধিক ব্যবহার করা হয়। 16 × 2 LCD বলতে এটি প্রতি লাইনে 16টি করে অক্ষর নিয়ে 2 লাইন প্রদর্শন করতে পারে। এর প্রতিটি অক্ষর 5 × 6 পিক্সেল মেট্রিক্সে প্রদর্শন করে। নিচে একটি 16 × 2 LCD, HD44780-এর পিন ডায়াগ্রাম দেখানো হলো :




চিত্র-১২.১ : HD44780 (16 × 2) LCD-এর পিন ডায়াগ্রাম

**একটি HD44780 (16 × 2) LCD-এর পিনের বর্ণনা :**

i. DB0 – DB7 : 4 বিট ডাটা বাস, ডাটা লিখন/পঠন করে।

ii. Vss, Vdd : ভোল্টেজ সাপ্লাই পিন।

iii. R/W : LCD তে লিখন বা LCD হতে পঠন সিগন্যাল পিন।

iv. RS : রেজিস্টার নির্বাচন পিন। এর সাহায্যে ইনস্ট্রাকশন রেজিস্টার ও ডাটা রেজিস্টার নির্বাচন করা হয়।

v. E : এনাবল পিন। এটি LCD তে লিখন/পঠন (R/W) অপারেশন কার্যকর বা অকার্যকর করে। এটি High হলে কার্যকর হয় এবং Low হলে অকার্যকর হয়।

vi. Vo : এটি LCD contrast পিন।

## ১২.২ ইনস্ট্রাকশন রেজিস্টার, ডাটা রেজিস্টার ও বিজি ফ্ল্যাগ বর্ণনা

**(Describe the Instruction Register, Data Register and Busy Flag)**

HD44780 (16 × 2) LCD-এর দুটি রেজিস্টার আছে :

(ক) ডাটা রেজিস্টার (Data Register)

(খ) কমান্ড রেজিস্টার বা ইনস্ট্রাকশন রেজিস্টার (Command Register or Instruction Register)

**(ক) ডাটা রেজিস্টার (Data Register) :** ডাটা রেজিস্টার একটি 8 বিটের রেজিস্টার। এটি LCD-তে প্রদর্শিত তথ্য ধারণ করে। LCD তে প্রদর্শিত তথ্য ASCII মানের হয়। এটি Display data ram (DDRAM) এবং character data RAM (CGRAM)-এ যে তথ্য লিখিত (write) হবে তা সাময়িকভাবে ধারণ করে। DDRAM বা CGRAM হতে যে তথ্য পঠিত (Read) হবে তাও সাময়িকভাবে ধারণ করার জন্য ব্যবহার করা হয়।

Register Select (RS) সিগন্যালের মাধ্যমে উপরের দুই রেজিস্টার নির্বাচন করা হয়। নিচে ছক আকারে দেখানো হলো–

| Register Selector | | |
|---|---|---|
| **RS** | **R/W** | |
| 0 | 0 | LCD তে একটি কমান্ড পাঠানো হয়। |
| 0 | 1 | Read *busy flag (DB7) এবং address counter (DB0 to DB6) |
| 1 | 0 | LCD তে তথ্য পাঠায়। |
| 1 | 1 | LCD থেকে তথ্য পড়ে। |

**(খ) কমান্ড রেজিস্টার বা ইনস্ট্রাকশন রেজিস্টার (Command Register or Instruction Register) :** Command register বা Instruction Register একটি 8 বিটের রেজিস্টার। এটি LCD তে প্রদেয় কমান্ড ইনস্ট্রাকশনসমূহ ধারণ করে। LCD তে প্রদেয় একটি নির্দেশনা হলো কমান্ড, যা পূর্বনির্ধারিত কাজ যেমন LCD সূচনা করা, পর্দা পরিষ্কার করা, কার্সরের অবস্থান সেটিং করা, ডিসপ্লে নিয়ন্ত্রণ ইত্যাদি সম্পাদন করে। এটি Display data ram (DDRAM) এবং Character data RAM (CGRAM)-এর তথ্যও ধারণ করে।

**বিজি ফ্ল্যাগ (BF-Busy Flag) :** পূর্বের নির্দেশনা সমাপ্তি বা পরবর্তী নির্দেশনার জন্য প্রস্তুত কি না তার জন্য BF ফ্ল্যাগ LCD তে একটি ইঙ্গিত দেয়।

## ১২.৩ এলসিডি কমান্ড তালিকা

**(List the LCD Command Codes)**

LCD-এর কমান্ড (Hex code)-এর তালিকা নিচে উল্লেখ করা হলো :

| Code (Hex) | LCD Instruction Register Commands |
|---|---|
| 1 | LCD পর্দা পরিষ্কার করে (Clear display screen)। |
| 2 | Return home |
| 4 | কার্সরের অবস্থান বামদিকে একঘর স্থানান্তরিত করে (Shift cursor to left)। |
| 5 | ডিসপ্লেকে ডানদিকে স্থানারিত করে (Shift display right)। |

| | |
|---|---|
| 6 | কার্যরের অবস্থান ডানদিকে একঘর স্থানান্তরিত করে (Shift cursor to right)। |
| 7 | ডিসপ্লেকে বামদিকে স্থানান্তরিত করে (Shift display left)। |
| 8 | ডিসপ্লে ও কার্সর অফ (Display off, cursor off)। |
| A | ডিসপ্লে অফ ও কার্সর অন (Display off, cursor on)। |
| C | ডিসপ্লে অন ও কার্সর অফ (Display on, cursor off), |
| E | ডিসপ্লে অন ও কার্সর মিটমিট করে জ্বলে (Display on, cursor blinking)। |
| F | ডিসপ্লে অন ও কার্সর মিটমিট করে জ্বলে (Display on, cursor blinking)। |
| 10 | কার্সরের অবস্থান বামদিকে স্থানান্তরিত করে (Shift cursor position to left)। |
| 14 | কার্সরের অবস্থান ডানদিকে একঘর স্থানান্তরিত করে (Shift cursor position to right)। |
| 18 | সম্পূর্ণ ডিসপ্লেকে বামদিকে স্থানান্তরিত করে (Shift the entire display to the left)। |
| 1C | সম্পূর্ণ ডিসপ্লেকে ডানদিকে স্থানান্তরিত করে (Shift the entire display to the right)। |
| 80 | কার্সরকে ১ম লাইনের শুরুতে আনার জন্য জোর দেয় (Force cursor to beginning to 1st line)। |
| C0 | কার্সরকে ২য় লাইনের শুরুতে আনার জন্য জোর দেয় (Force cursor to beginning to 2nd line)। |
| 38 | 2 লাইন ও 5 × 7 মেট্রিক্স (2 lines and 5 × 7 matrix)। |

## ১২.৪ LCD- তে ডাটা প্রদর্শনের প্রোগ্রাম

### (Write Program for Displaying Data to LCD)

**'NO' শব্দটি LCD- তে ডিসপ্লে করার জন্য প্রোগ্রাম :**

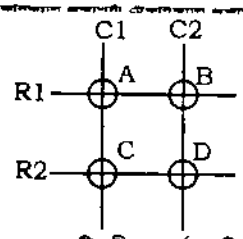
```
; calls a time delay before sending next data/command
; P1.0 – P1.7 are connected to LCD data pins D0 – D7
; P2.0 is connected to RS pin of LCD
; P2.1 is connected to R/W pin of LCD
; P2.2 is connected to E pin of LCD
        ORG 0H
        MOV A, # 38H        ; INIT.LCD 2 LINES, 5×7 MATRIX
        ACALL COMNWRT       ; call command subroutine
        ACALL DELAY         ; give LCD some time
        MOV A, #0EH         ; display on cursor on
        ACALL COMNWRT       ; call command suroutine
        ACALL DELAY         ; give LCD some time
```

```
            MOV A, #01          ; clear LCD
            ACALL COMNWRT       ; call command subroutine
            ACALL DELAY         ; give LCD some time
            MOV A, #'N'         ; display letter N
            ACALL DATA WRT      ; call display subroutine
            ACALL DELAY         ; give LCD some time
            MOV A, # 'O'        ; display letter O
            ACALL DATAWRT       ; call display subroutine
AGAIN :     SJMP AGAIN          ; stay here
COMNWRT                         ; send command to LCD
            MOV P1, A           ; copy reg A to port 1
            CLR P2.0            ; RS = 0 for command
            CLR P2.1            ; R/W = 0 for write
            SETB P2.2           ; E = 1 for high pulse
            ACALL DELAY         ; give LCD some time
            CLR P2.2            ; E = 0 for H-to-L pulse
            RET
DATAWRT :                       ; write date to LCD
            MOV P1, A           ; copy reg A to port 1
            SETB P2.0           ; RS = 1 for data
            CLR P2.1            ; R/W = 0 for data
            SETB P2.2           ; E = 1 for high pulse
            ACALL DELAY         ; give LCD some time
            CLR P2.2            ; E = 0 for H-to-L pulse
            RET
DELAY :     MOV R3, #50         ; 50 or higher for fast CPUs
HERE2 :     MOV R4, #255        ; R4 = 255
HERE :      DJNZ R4, HERE       ; stay until R4 becomes 0
            DJNZ R3, HERE 2
            RET
            END
```

## ১২.৫ একটি মেট্রিক্স কী-বোর্ডের গঠন বর্ণনা

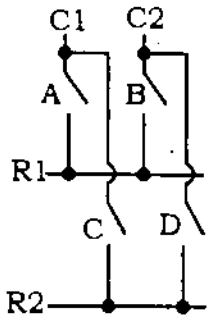
### (Describe the Organization of a Matrix Keyboard)

মেট্রিক্স কী-বোর্ড খুবই জনপ্রিয়। এর সরল গঠন ও সহজ ইন্টারফেসিং জন্য এটি মাইক্রকন্ট্রোলারে ব্যাপক ভাবে ব্যবহার করা হয়। কী-বোর্ড একটি HMI (Human Machine Interface)-এর অংশ এবং একটি ছোট embedded system-এ এর ভূমিকা অনেক। মেট্রিক্স কী-বোর্ড তারের তৈরি সারি ও কলামের একটি মেট্রিক্স নিয়ে গঠিত। প্রত্যেক কী একটি সুইচের মতো আচরণ করে। যখন একটি কীতে চাপ প্রয়োগ করা হয় তখন একটি কলামের তার একটি সারির তারের সাথে সংযোগ হয়ে একটি সার্কিট পূর্ণ করে। কী-বোর্ড কন্ট্রোলার এই ক্লোজ সার্কিটকে শনাক্ত করে এবং একটি "কী চাপ" রেজিস্টার করে। পাশে একটি ধারণাসংক্রান্ত সরল কী-বোর্ড মেট্রিক্স সার্কিট দেখানো হলো।



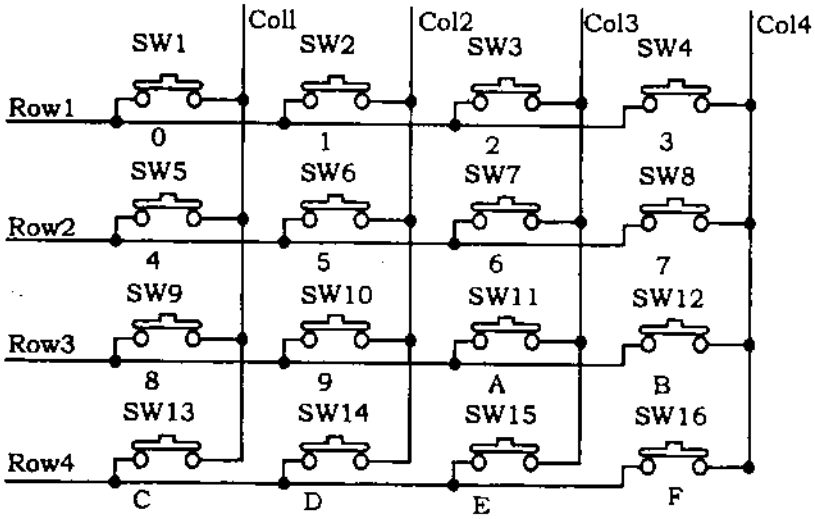

চিত্র-১২.২ : একটি কী-বোর্ড মেট্রিক্স সার্কিট

এই কী-বোর্ডের 4টি কী- A, B, C ও D। গ্রাফের বিন্দুর মতো প্রত্যেক কী এর একটি অদ্বিতীয় গ্রিড লোকেশন আছে। A কী এর নুড C1R1, B কী এর নুড C2R1, C কী এর নুড C1R2 এবং D কী এর নুড C2R2। এর ইলেকট্রনিক সার্কিট দেখতে অনেকটা নিচের চিত্রের মতো।




চিত্র-১২.৩ : সুইচ খোলা

প্রকৃত কী-বোর্ড অনেকগুলো কলাম ও সারি নিয়ে গঠিত। মাইক্রোকন্ট্রোলার পোর্টের মাধ্যমে উভয় সারি ও কলামকে access করে। 8 বিটের পোর্টের সাহায্যে একটি 8 × 8 কী এর মেট্রিক্সকে একটি মাইক্রোকন্ট্রোলারের সাথে সংযোগ করা যায়। যখন একটি কীতে চাপ প্রয়োগ করা হয় তখন একটি সারি ও একটি কলাম একত্রে একটি সংযোগ তৈরি করে, অন্যথায় সারি ও কলামের মাঝখানে কোন সংযোগ হয় না। নিচে 4 × 4 মেট্রিক্সের একটি কী-বোর্ড নিচে দেখানো হলো :

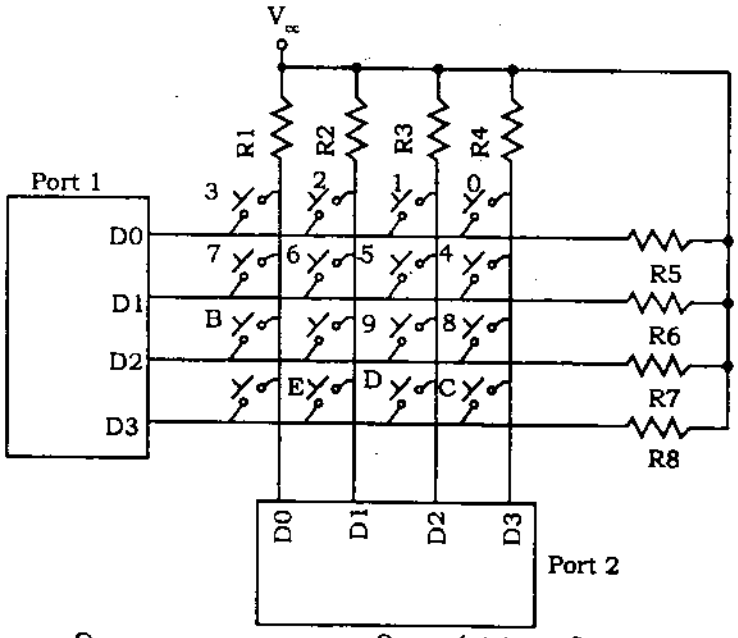



চিত্র-১২.৪ : 8 × 8 মেট্রিক্স কী-বোর্ড

## ১২.৬ একটি কী চাপ শনাক্তকরণ ও চিহ্নিতকরণের ধাপ ব্যাখ্যা

**(Explain the Steps to Detect and Identify the Key Pressed)**

নিচের চিত্রে মাইক্রোকন্ট্রোলারের সাথে কী-বোর্ডের সংযোগ দেখানো হলো। এর কলামকে ইনপুট এবং সারিকে আউটপুট করা হয়েছে। কী-বোর্ড পঠনের সম্পূর্ণ প্রক্রিয়াকে scanning বলে।




চিত্র-১২.৫ : 8051 এর সাথে কী-বোর্ডের ইন্টারফেসিং

মাইক্রোকন্ট্রোলারের কাজ হলো কী-চাপ শনাক্ত ও চিহ্নিত করার জন্য নিচের ধাপগুলো অনুসরণ করা :

১। মাইক্রোকন্ট্রোলার কী-বোর্ডকে অনবরত স্ক্যান করে।

২। মাইক্রোকন্ট্রোলার সব সারিকে গ্রাউন্ড করে আউটপুট latch কে 0 করে।

(ক) যদি কলাম থেকে ডাটার মান, D3 – D0 = 11 – 11 হয় তবে কোন কী-চাপ হয়নি এবং কী-চাপ না হওয়া পর্যন্ত এ প্রক্রিয়া চলতে থাকবে।

(খ) যদি কোন কলাম বিটের মান 0 হয় তাতে কোন কী-চাপ হয়েছে বুঝায়। ধরি D3 – D0 = 1101, তবে D1 কলামের কোন কী-চাপ হয়েছে।

(গ) একটি কী-চাপ শনাক্তকরণের পর মাইক্রোকন্ট্রোলার কী চিহ্নিতকরণের জন্য প্রক্রিয়া করে।

৩। মাইক্রোকন্ট্রোলার শুধুমাত্র D0 সারিতে 0 (Low) সরবরাহ দ্বারা সবার উপরের সারি থেকে গ্রাউন্ড করা শুরু করে। এবার মাইক্রোকন্ট্রোলার কলামসমূহকে পড়ে এবং যদি পঠিত ডাটার মান সব 1 হয় তবে এই সারিতে কোন কী-চাপ হয়নি। অতঃপর প্রক্রিয়া পরবর্তী সারিতে অগ্রসর হয়।

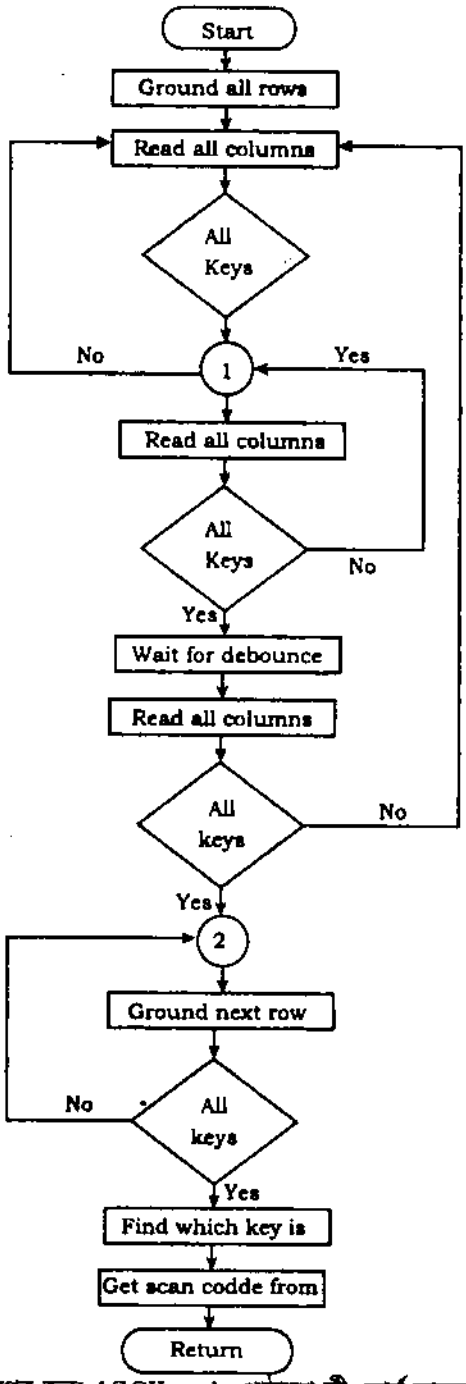
৪। মাইক্রোকন্ট্রোলার পরবর্তী সারিকে গ্রাউন্ড করে এবং কলামকে পড়ে কোন জিরো থাকলে যাচাই করে। এ প্রক্রিয়া কোন সারি শনাক্ত না হওয়ার পর্যন্ত চলতে থাকে।

৫। এভাবে এক পর্যায়ে যে সারিতে কী-চাপ তা চিহ্নিত হয় এবং এই সারিতে যে কলামের বিট জিরো হয় তা বের করে সংশ্লিষ্ট কী শনাক্ত করা হয়।

উদাহরণস্বরূপ : সারির জন্য D3 – D0 = 1110 এবং কলামের জন্য D3 – D0 = 1011 হলে যে কী চাপ দেয়া হয়েছে তার সংশ্লিষ্ট সারি হলো D0 এবং কলাম D2। সুতরাং যে কী চাপ দেয়া হয়েছে তার নং 2।

## ১২.৭ কোন পোর্টে কী চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম (Write a Keyboard Subroutine to Send the ASCII Code for Pressed Key to a Port)

কোন পোর্টে কী চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম প্রবাহ চিত্র (Flow Chart) নিচে উল্লেখ করা হলো :




চিত্র-১২.৬ : পোর্টে কী-চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রামের প্রবাহ চিত্র

```
; keyboard subroutine. This program sends the ASCII code for pressed key to P0.1
; P1.0 – P1.3 connected to rows and
; P2.0 – P2.3 connected to coloumn
        MOV P2, #0FFH                   ; make P2 an input port
K1 :    MOV P1, #0                      ; ground all rows at one
        MOV A, P2                       ; read all column (ensure keys open)
        ANL A, #00001111B               ; mased unused bits
        CNINE A, #00001111B, K1         ; till all keys release
K2 :    ACALL DELAY                     ; call 20 ms delay
        MOV A, P2                       ; see any key is pressed
        ANL A, #00001111b               ; masked unused bits
        CJNE A, #00001111B, OVER        ; key pressed, find any row
        SJMP K2                         ; check till key pressed
OVER :  ACALL DELAY                     ; wait 20ms debounce time
        MOV A, P2                       ; check key closure
        ANL A, 00001111B                ; masked unused bits
        CJNE A, #00001111B, OVER1       ; key pressed, find row
        SJMP K2                         ; if none, keep poling
OVER1:  MOV P1, #11111110B              ; ground row 0
        MOV A, P2                       ; read all columns
        ANL A, #00001111B               ; mask unused bits
        CJNE A, #00001111B, ROW_0       ; key row 0, find col.
        MOV P1, #11111101B              ; ground row 1
        MOV A, P2                       ; read all columns
        ANL A, #00001111B               ; mask unused bits
        CNJE A, #00001111B,ROW_1        ; key row 1, find col.
        MOV P1, #11111011B              ; ground row 2
        MOV A, P2                       ; read all columns
        ANL A, #00001111B               ; mask unused bits
        CJNE A, #00001111B,ROW_2        ; key row 2, find col.
        MOV P1, #11110111B              ; ground row 3
        MOV A, P2                       ; read all columns
        ANL A, #00001111B               ; mask unused bits
```

```
        CJNE A, #00001111B, ROW_3     ; key row 3, find col.
        LJMP K2                       ; if none, false input,
                                      ; repeat
        ROW_0: MOV DPTR, #KCODE0      ; set DPTR = start of row 0
        SJMP FIND                     ; find col. Key belongs to
        ROW_1:MOV DPTR, #KCODE1       ; set DPTR = start of row
        SJMP FIND                     ; find col. Key belongs to
ROW_2:  MOV DPTR, #KCODE2             ; set DPTR = start of row2
        SJMP FIND                     ; find col. Key belongs to
ROW_3:  MOV DPTR, #KCODE3             ; set DPTR = start of row3
FIND :  RRC A                         ; see if any CY bit low
        JNC MATCH                     ; if zero, get ASCII code
        INC DPTR                      ; point to next col. addr
        SJMP FIND                     ; keep searching
MATCH:  CLR A                         ; set A = 0 (match is found)
        MOVC A, @a+DPTR               ; get ASCII from table
        MOV P0, A                     ; display pressed key
        LJMP K1
; ASCII look-up table for each row
        ORG 300H                      ; ROW 0
        KCODE0 : DB '0', '1','2','3'  ; ROW 1
        KCODE1 : DB '4', '5', '6', '7' ; ROW 2
        KCODE2 : DB '8', '9', '9', 'A', 'B' ; ROW 3
        KCODE3 : DB 'C', 'D', 'E', 'F'
        END
```

# অনুশীলনী-১২

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

১। **Busy Flag কী?** *[বাকাশিবো : '১৪]*

**উত্তর :** LCD-এর Register Select (RS) সিগন্যালের মাধ্যমে উপরের কমান্ড রেজিস্টার ও ডাটা রেজিস্টার নির্বাচন করা হয়।

২। **16 × 2 LCD দ্বারা কী বুঝায়?**

**উত্তর :** 16 × 2 LCD বলতে এটি প্রতি লাইনে 16টি করে অক্ষর নিয়ে 2 লাইন প্রদর্শন করতে পারে।

৩। **একটি বাণিজ্যিক LCD-এর নাম লিখ।**

**উত্তর :** একটি বাণিজ্যিক LCD-এর নাম হলো : HD44780। এটি একটি 16 × 2 LCD।

৪। **LCD-এর কমান্ড রেজিস্টারের কাজ কী?**

**উত্তর :** কমান্ড রেজিস্টার LCD-তে প্রদেয় কমান্ড ইনস্ট্রাকশনসমূহ ধারণ করে। কমান্ড হলো LCD তে প্রদেয় একটি নির্দেশনা, যা পূর্বনির্ধারিত কাজ যেমন- LCD সূচনা করা, পর্দা পরিষ্কার করা, কার্সরের অবস্থান সেটিং করা, ডিসপ্লে নিয়ন্ত্রণ ইত্যাদি সম্পাদন করে। এটি Display data ram (DDRAM) এবং character data RAM (CGRAM)-এর তথ্য ধারণ করে।

৫। **LCD-এর ডাটা রেজিস্টারের কাজ কী?**

**উত্তর :** LCD-এর ডাটা রেজিস্টার LCD তে প্রদর্শিত তথ্য ধারণ করে। LCD তে প্রদর্শিত তথ্য ASCII মানে হয়। এটি Display data ram (DDRAM) এবং character data RAM (CGRAM)-এ যে তথ্য লিখিত (write) হবে তা সাময়িকভাবে ধারণ করে। DDRAM বা CGRAM হতে যে তথ্য পঠিত (Read) হবে তাও সাময়িকভাবে ধারণ করার জন্য ব্যবহৃত হয়।

৬। **LCD-এর Register select সিগন্যালের কাজ কী?**

**উত্তর :** LCD-এর Register Select (RS) সিগন্যালের মাধ্যমে উপরের কমান্ড রেজিস্টার ও ডাটা রেজিস্টার নির্বাচন করা হয়।

৭। **LCD-এর Busy Flag (F)-এর কাজ কী?**

**উত্তর :** LCD-এর Busy Flag (BF) LCD তে পূর্বের নির্দেশনা সমাপ্তি বা পরবর্তী নির্দেশনার জন্য প্রস্তুত কি না তার একটি ইঙ্গিত দেয়।

৮। **Hex code-এ যে কোন 4 LCD command-এর নাম লিখ।**

**উত্তর :** LCD-এর কমান্ড (Hex code)-এর তালিকা নিচে উল্লেখ করা হলো :

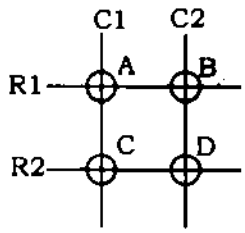
| Code (Hex) | LCD Instruction Register Commands |
|---|---|
| 1 | LCD পর্দা পরিষ্কার করে (Clear display screen)। |
| 2 | Return home |
| 4 | কার্সরের অবস্থান বামদিকে একঘর স্থানান্তরিত করে (Shift cursor to left)। |
| 5 | ডিসপ্লেকে ডানদিকে স্থানারিত করে (Shift display right)। |

**৯। মেট্রিক্স কী-বোর্ডের সরল গঠন কী?**

**উত্তর :** মেট্রিক্স কী-বোর্ডে তারের তৈরি সারি ও কলামের একটি মেট্রিক্স নিয়ে গঠিত। প্রত্যেক কী একটি সুইচের মতো আচরণ করে। যখন একটি কীতে চাপ প্রয়োগ করা হয় একটি কলামের তার একটি সারির তারের সাথে সংযোগ হয়ে একটি সার্কিট পূর্ণ করে। কী-বোর্ড কন্ট্রোলার এই ক্লোজ সার্কিটকে শনাক্ত করে এবং একটি "কী-চাপ" হিসাবে রেজিস্টার করে।

**১০। বিভিন্ন নুড চিহ্নিত করে একটি সরল মেট্রিক্স কী-বোর্ড অঙ্কন কর।**

**উত্তর :** নিচে একটি সরল মেট্রিক্স কী-বোর্ড দেখানো হলো।




**চিত্র : একটি কী-বোর্ড মেট্রিক্স সার্কিট**

এর নুডগুলো A(R1C1), B(R1C2), C(R2C1) ও D(R2C2)

**১১। একটি 4 × 4 মেট্রিক্স কী-বোর্ডের নিচের আউটপুটের জন্য pressed key-এর কলাম ও সারি শনাক্ত কর এবং pressed key নির্ণয় কর।**

**সারির জন্য :** D3 – D0 = 1101 এবং কলামের জন্য D3 – D0 = 0111

**উত্তর :** D3 – D0 = 1101-এর জন্য নির্বাচিত সারি D0 এবং D3 – D0 = 0111-এর জন্য নির্বাচিত কলাম D2। সুতরাং এই কলাম ও সারির জন্য pressed key হলো 2(Key 2)।

**১২। LCD এর পূর্ণরূপ লেখ :**

**উত্তর :** LCD এর পূর্ণরূপ = Liquid Crystal Display.

**১৩। LCD-এর তৈরি দুটি ডিভাইসের নাম লেখ।**

**উত্তর :** LCD-এর তৈরি দুটি ডিভাইসের নাম হলো :

(ক) মনিটর ও

(খ) টেলিভিশন।

**১৪। পূর্ণরূপ লেখ : (i) DDRAM (ii) CGRAM**

**উত্তর :** (i) DDRAM (ii) CGRAM এর পূর্ণরূপ নিচে দেওয়া হলো :

(i) DDRAM = Display Data Read Only Memory

(ii) CGRAM = Character Data Read Only Memory

## সংক্ষিপ্ত প্রশ্নোত্তর

**১। একটি 16 × 2 LCD (HD44780)-এর পিনগুলোর কাজ লিখ।**

**অথবা, একটি 14 পিন LCD এর পিনগুলোর কাজ লেখ।** *[বাকাশিবো : '১৫]*

**উত্তর :** একটি HD44780 (16 × 2) LCD-এর পিনের বর্ণনা :

i. DB0 – DB7 : 4 বিট ডাটা বাস, ডাটা লিখন/পঠন করে।

ii. Vss, Vdd : ভোল্টেজ সাপ্লাই পিন।

iii. R/W : LCD তে লিখন বা LCD হতে পঠন সিগন্যাল পিন।

iv. RS : রেজিস্টার নির্বাচন পিন। এর সাহায্যে ইনস্ট্রাকশন রেজিস্টার ও ডাটা রেজিস্টার নির্বাচন

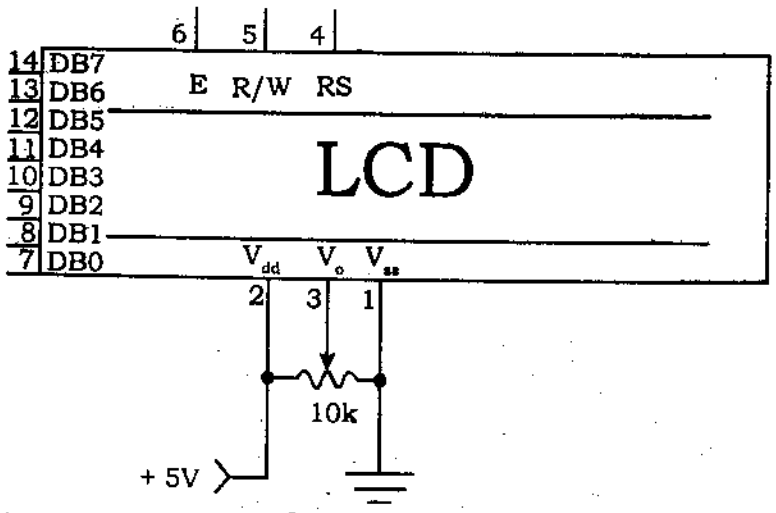
v. E : এনাবল পিন। এটি LCD তে লিখন/পঠন (R/W) অপারেশন কার্যকর বা অকার্যকর করে। এটি High হলে কার্যকর হয় এবং Low হলে অকার্যকর হয়।

vi. Vo : এটি LCD contrast পিন।

**২। LCD, HD44780-এর পিন ডায়াগ্রাম অঙ্কন কর।**

**অথবা, LCD-এর পিন ডায়াগ্রাম অঙ্কন কর।** *[বাকাশিবো : '১৪, '১৫]*

**উত্তর :** নিচে LCD, HD44780-এর পিন ডায়াগ্রাম দেখানো হলো :




**চিত্র :** HD44780 (16 × 2) LCD-এর পিন ডায়াগ্রাম

**৩। HD44780 LCD-এর ইনস্ট্রাকশন রেজিস্টার ও ডাটা রেজিস্টার বর্ণনা কর।**

**উত্তর :** HD44780 (16 × 2) LCD-এর দুটি রেজিস্টার আছে :

(ক) ডাটা রেজিস্টার (Data Register)

(খ) কমান্ড রেজিস্টার বা ইনস্ট্রাকশন রেজিস্টার (Command Register or Instruction Register)

**(ক) ডাটা রেজিস্টার (Data Register) :** ডাটা রেজিস্টার একটি 8 বিটের রেজিস্টার। এটি LCD- তে প্রদর্শিত তথ্য ধারণ করে। LCD তে প্রদর্শিত তথ্য ASCII মানে হয়। এটি Display data ram (DDRAM) এবং character data RAM(CGRAM)-এ যে তথ্য লিখিত (write) হবে তা

সাময়িকভাবে ধারণ করে। DDRAM বা CGRAM হতে যে তথ্য পঠিত (Read) হবে তাও সাময়িকভাবে ধারণ করার জন্য ব্যবহার করা হয়।

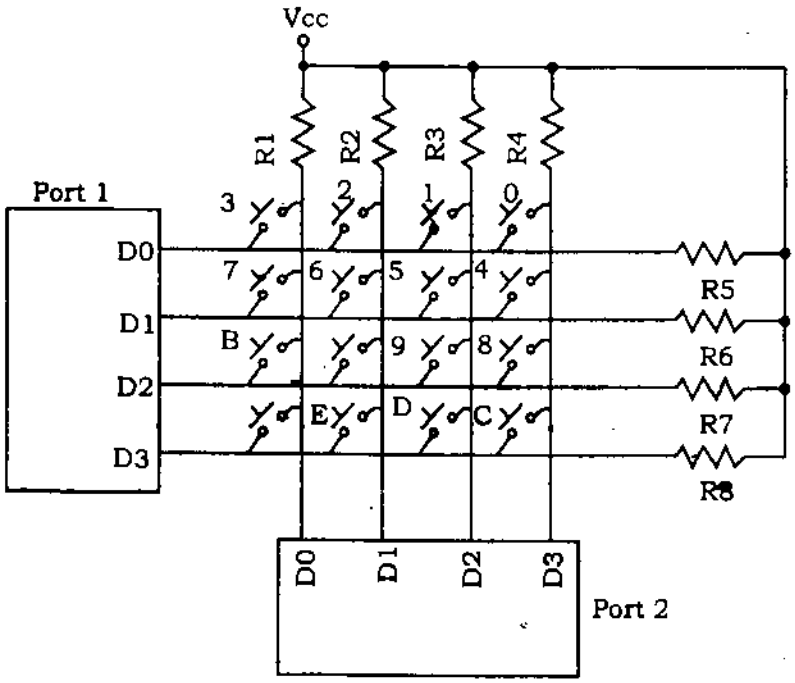
Register Select (RS) সিগন্যালের মাধ্যমে উপরের দুই রেজিস্টার নির্বাচন করা হয়। নিচে ছক আকারে দেখানো হলো–

| Register Selector | | |
|---|---|---|
| **RS** | **R/W** | |
| 0 | 0 | LCD তে একটি কমান্ড পাঠানো হয়। |
| 0 | 1 | Read *busy flag (DB7) এবং address counter (DB0 to DB6) |
| 1 | 0 | LCD তে তথ্য পাঠায়। |
| 1 | 1 | LCD থেকে তথ্য পড়ে। |

**(খ) কমান্ড রেজিস্টার বা ইনস্ট্রাকশন রেজিস্টার (Command Register or Instruction Register) :** Command register বা Instruction Register একটি 8 বিটের রেজিস্টার। এটি LCD তে প্রদেয় কমান্ড ইনস্ট্রাকশনসমূহ ধারণ করে। LCD তে প্রদেয় একটি নির্দেশনা হলো কমান্ড, যা পূর্বনির্ধারিত কাজ যেমন LCD সূচনা করা, পর্দা পরিষ্কার করা, কার্সরের অবস্থান সেটিং করা, ডিসপ্লে নিয়ন্ত্রণ ইত্যাদি সম্পাদন করে। এটি Display data ram (DDRAM) এবং character data RAM (CGRAM)-এর তথ্যও ধারণ করে।

**৪। একটি 4 × 4 মেট্রিক্স কী-বোর্ডের সাথে 8051 মাইক্রোকন্ট্রোলারের ইন্টারফেসিং সার্কিট অঙ্কন কর।**

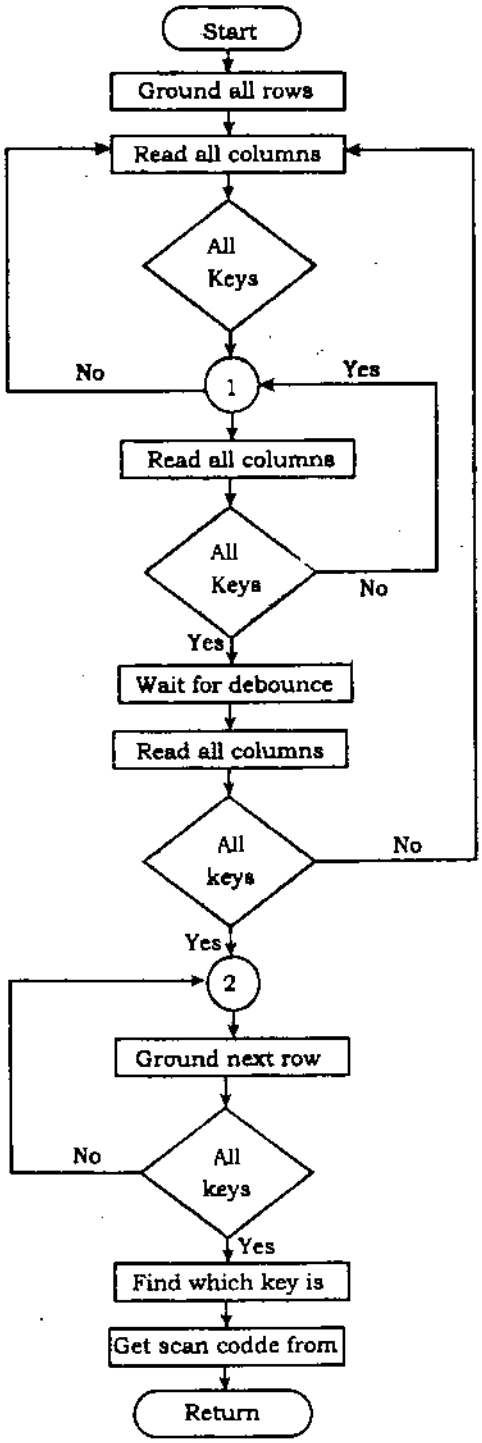
**উত্তর :** একটি 4 × 4 মেট্রিক্স কী-বোর্ডের সাথে 8051 মাইক্রোকন্ট্রোলারের ইন্টারফেসিং সার্কিট অঙ্কন করা হলো :




চিত্র : 8051 এর সাথে কী-বোর্ডের ইন্টারফেসিং

**৫। কোন পোর্টে কী চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম প্রবাহ চিত্র অঙ্কন কর।**

**উত্তর :** কোন পোর্টে কী চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম প্রবাহ চিত্র নিচে উল্লেখ করা হলো :




**চিত্র :** পোর্টে কী-চাপের জন্য ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রামের প্রবাহ চিত্র

## রচনামূলক প্রশ্নোত্তর

১। **একটি মেট্রিক্স কী-বোর্ডের গঠনপ্রণালী বর্ণনা কর।** *[বাকাশিবো : '১৪]*

**উত্তর :** অনুচ্ছেদ ১২.৫ নং দ্রষ্টব্য।

২। **LCD তে ELECTRONIC শব্দটি ডিসপ্লে করার জন্য একটি 8021 অ্যাসেম্বলি প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ১২.৪ নং দ্রষ্টব্য।

৩। **একটি কী-চাপ শনাক্তকরণ ও চিহ্নিতকরণের ধাপগুলো বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১২.৬ নং দ্রষ্টব্য।

৪। **কোন পোর্টে কী চাপের ASCII code পাঠানো কী-বোর্ড সাবরুটিন প্রোগ্রাম লিখ।**

**উত্তর :** অনুচ্ছেদ ১২.৭ নং দ্রষ্টব্য।

৫। **ইনস্ট্রাকশন রেজিস্টার, ডাটা রেজিস্টার ও বিজি ফ্ল্যাগ বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১২.২ নং দ্রষ্টব্য।

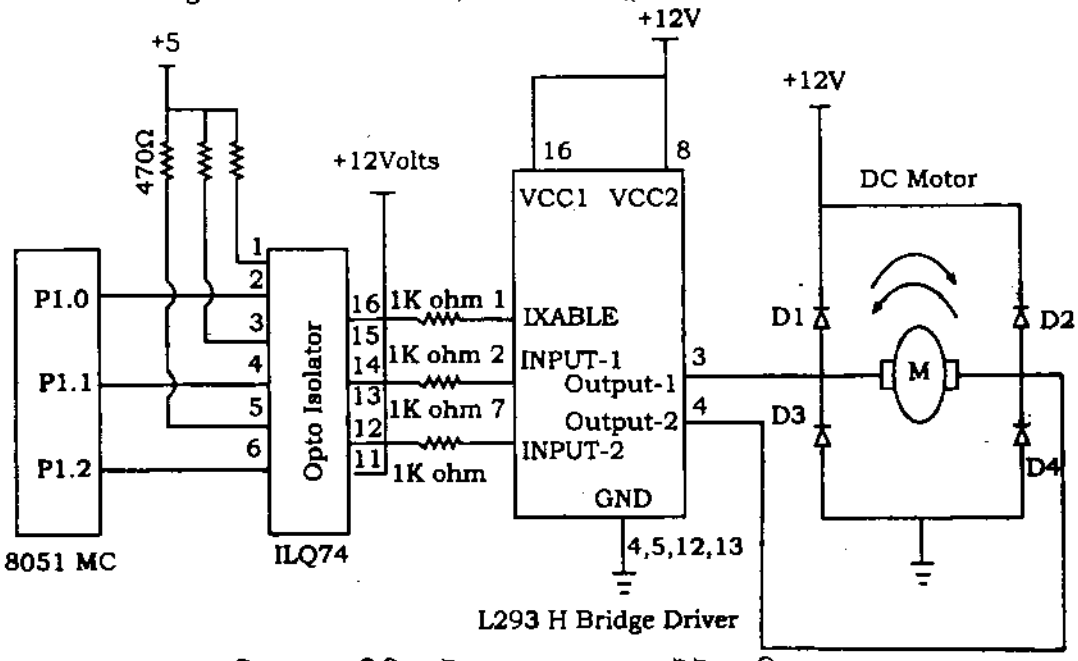
৬। **এলসিডি কমান্ড তালিকা বর্ণনা কর।**

**উত্তর :** অনুচ্ছেদ ১২.৩ নং দ্রষ্টব্য।

# অধ্যায় ১৩ বাস্তব জগতের সাথে ইন্টারফেসিং
# Understand Real World Interfacing

## ১৩.১ ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং বর্ণনা
## (Describe Interfacing the 8051 With a DC Motor)

DC কারেন্টে ডিসি মোটর চলে। এটি বৈদ্যুতিক ও চৌম্বক ক্ষেত্র ব্যবহার করে টর্ক উৎপন্ন করে। চলন্ত অবস্থার চেয়ে প্রাথমিক অবস্থায় টর্ক উৎপন্ন করার জন্য ডিসি মোটরের অধিক কারেন্ট দরকার হয়। 8051-এর সাথে ডিসি মোটরের ইন্টারফেসিং সরাসরি সম্ভব নয়। কারণ ডিসি মোটরকে চালনার জন্য অধিক কারেন্ট (ছোট ডিসি মোটরের জন্য এটি প্রায় 200–300 mA) দরকার। যদি এই কারেন্ট 8051 মাইক্রোকন্ট্রোলার দিয়ে প্রবাহিত হয়, তবে 8051 IC নষ্ট হয়ে যাবে। অধিকন্তু মোটরের গতি বিপরীতমুখী হওয়ার সময় Voltage spike উৎপন্ন করে, যা মাইক্রোকন্ট্রোলারকে খুব সহজেই নষ্ট করে ফেলে। সুতরাং, অপটোকাপলার ও Dual H-Bridge driver (L298) নিয়ে গঠিত একটি ড্রাইভিং সার্কিটের সাহায্যে 8051 এর সাথে ডিসি মোটরকে ইন্টারফেসিং করা হয়। অপটোকাপলার মাইক্রোকন্ট্রোলারকে অতিরিক্ত সুরক্ষা প্রদান করে। মোটরের অবিরাম অপারেশনের ফলে L293 Dual H-Bridge ড্রাইভার overheat হয়, তাই একটি উপযুক্ত heat sink অবশ্যই ব্যবহার করতে হয়।



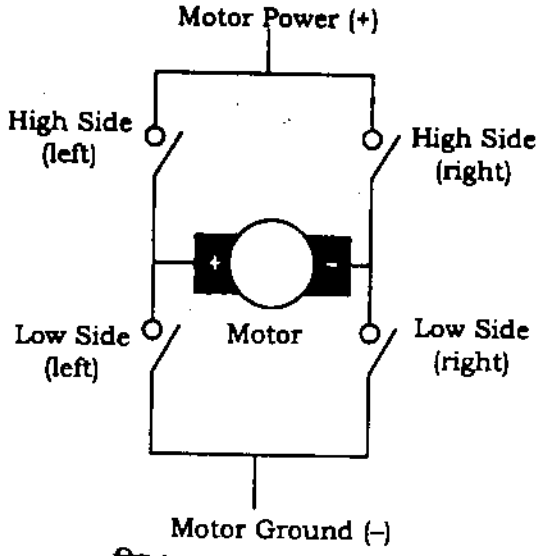

চিত্র-১৩.১ : ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং

**এইচ-ব্রিজ (H-Bridge) :** H-Bridge ট্রানজিস্টর বা MOSFET দ্বারা তৈরি একটি ব্রিজ। এটি মোটরের গতি নিয়ন্ত্রণ করে। এটিকে "Full Bridge"- ও বলে। এতে চারটি সুইচিং উপাদান রয়েছে। উপাদানগুলো নিচে দেওয়া হলো :

১। "High side left"
২। "High side right"
৩। "Low side right"
৪। "Low side left".

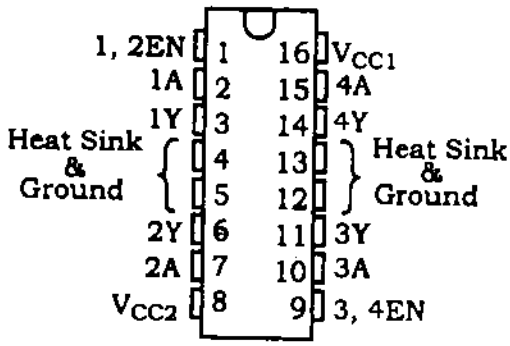
নিচে এর Truth table দেওয়া হলো :

| Truth Table | | | | |
|---|---|---|---|---|
| **High Left** | **High Right** | **Low Left** | **Low Right** | বর্ণনা |
| On | Off | Off | On | মোটর ক্লকওয়াইজ ঘুরে। |
| Off | On | On | Off | মোটর ক্লকওয়াইজের বিপরীত ঘুরে। |
| On | On | Off | Off | মোটর থেমে যায় বা গতি হ্রাস পায়। |
| Off | Off | On | On | মোটর থেমে যায় বা গতি হ্রাস পায়। |




চিত্র-১৩.২ : H-Bridge

**এল 293 মোটর ড্রাইভ (L293 Motor Driver) :** L293 একটি 16 পিনের quadruple Half H-Bridge মোটর ড্রাইভ IC। এর কারেন্ট বহন ক্ষমতা প্রতি চ্যানেল 600 mA এবং ভোল্টেজ রেঞ্জ 4.5–36 ভোল্ট (ডিসি)। ইন্ডাক্টিভ স্পাইক প্রতিরোধের জন্য এর অভ্যন্তরে উচ্চ গতির ক্ল্যাম্প ডায়োড আছে। তাছাড়া এর Noise immunity, ESD (Electro static discharge) প্রতিরোধ ক্ষমতা, Thermal shutdown অনেক বেশি এবং প্রতি চ্যানেলের পৃথক ইনপুট সাপ্লাই আছে। এর পিন ডায়াগ্রাম ও truth table চিত্রে দেখানো হলো :



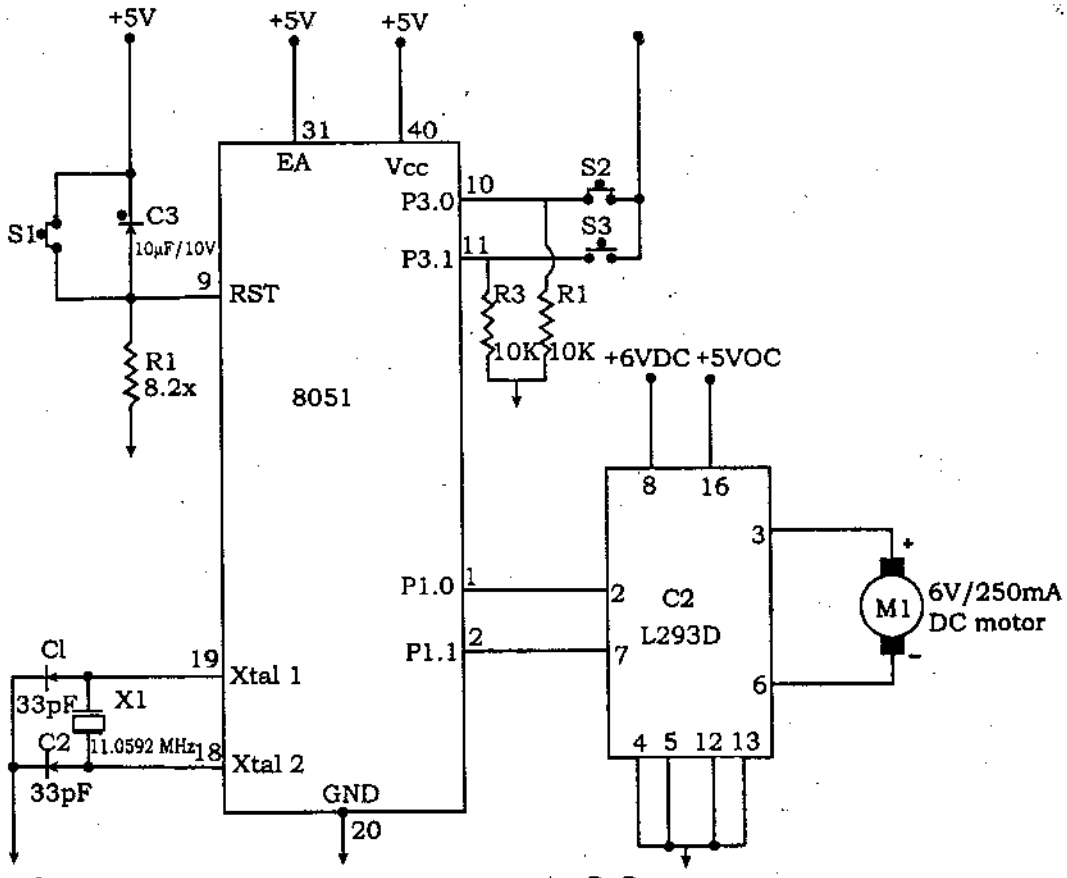

Function Table
(each driver)

| Inputs | | Output |
|---|---|---|
| A | EN | Y |
| H | H | H |
| L | H | L |
| X | L | Z |

H = high level, L = low level,
X = Irrelevent,
Z = High Impedance (Off)

চিত্র-১৩.৩ : L293 পিন ডায়াগ্রাম ও Truth table

**প্রোগ্রাম-১ :** 8051 ব্যবহার করে Push-button সুইচের মাধ্যমে ডিসি মোটরের গতির দিক পরিবর্তনের প্রোগ্রাম লিখ। গতির দিক পরিবর্তনের নিয়ন্ত্রণ সিগন্যালের জন্য পোর্ট পিন P1.0 ও P1.1 এবং Push-button সুইচের জন্য পোর্ট পিন P3.0 ও P3.1 ব্যবহারপূর্বক ইন্টারফেসিং সার্কিট দেখাও।




চিত্র-১৩.৪ : 8051 ব্যবহার করে Push-button সুইচ নিয়ন্ত্রিত Bi-directional DC motor

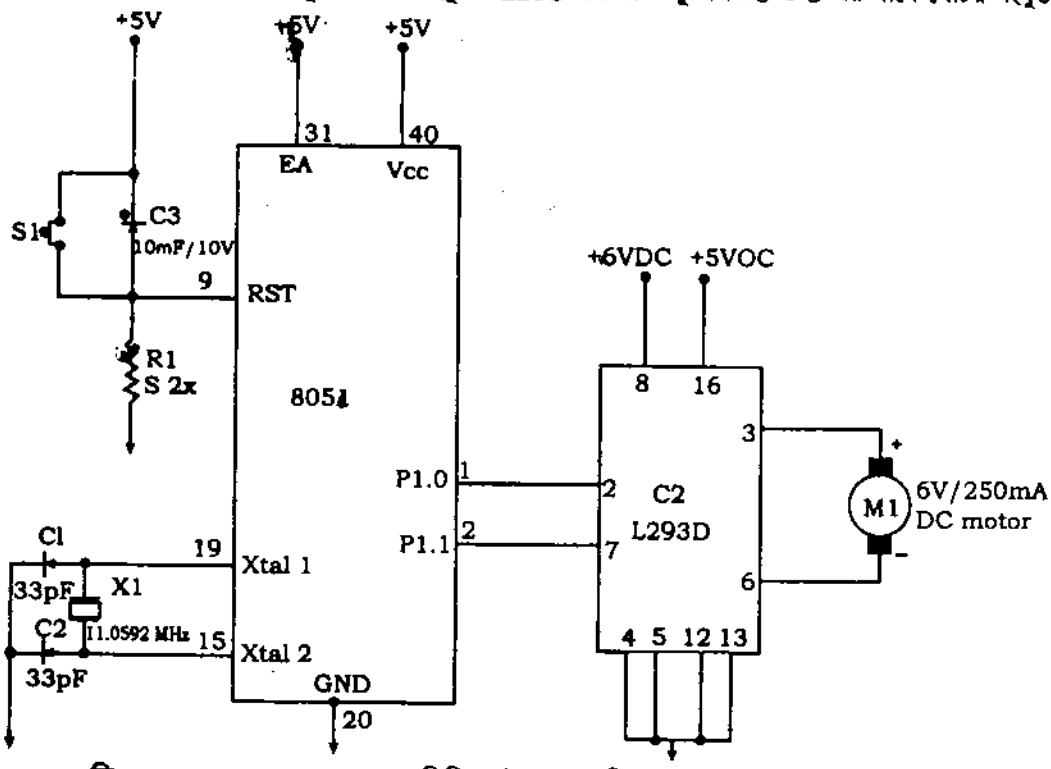
**সমাধান :** যখন S2 সুইচ চাপ দেওয়া হয় তখন P1.0 পিন High, P1.1 পিন Low হয় এবং মোটর ক্লকওয়াইজ ঘুরে। যতক্ষণ পর্যন্ত S2 সুইচ চাপ না দেওয়া হয় ততক্ষণ পর্যন্ত এই অবস্থা চলতে থাকে। S2 সুইচ চাপ দিলে মোটর অ্যান্টিক্লকওয়াইজ ঘুরে। R2 ও R3 রেজিস্টার যথাক্রমে P3.0 ও P3.1 পিনের Pull down রেজিস্টার।

**প্রোগ্রাম :**

```
        ORG 00H                          ; initial starting address
        MOV P3,#00000000B                ; initiates P3 as the push-button interface
        MOV P1,#00000000B                ; clears P1 for keeping the motor OFF initially
MAIN :  MOV A,P3                         ; moves the current state of P3 to Accumulator
        CJNE A, #00000001B, LABEL1       ; checks whether S2 is pressed
        MOV P1,#00000001B                ; makes the motor run clockwise
LABEL1 : CNJE A, #00000010B,LABEL2       ; checks wheter S3 is pressed
        MOV P1,#00000010B                ; makes the motor to run anticlockwise
LABEL2 : SJMP MAIN                       ; jumps back to the MAIN loop
        END
```

**প্রোগ্রাম-২ :** 8051 ব্যবহার করে একটি নির্দিষ্ট সময় পর পর ডিসি মোটরের স্বয়ংক্রিয়ভাবে গতির দিক পরিবর্তনের প্রোগ্রাম লিখ। গতির দিক পরিবর্তনের নিয়ন্ত্রণ সিগন্যালের জন্য পোর্ট পিন P1.0 ও P1.1 ব্যবহারপূর্বক ইন্টারফেসিং সার্কিট দেখাও।

**সমাধান :** ইন্টারফেসিং সার্কিট নিচে দেখানো হলো। এই সার্কিটে R1, S1 এবং C3 একত্রে একটি Debouncing reset সার্কিট গঠন করে। C1, C2 এবং X1 অসিলেটর সম্পর্কিত। P1.0 ও P1.1 পিন L293 মোটর ড্রাইভ এর ইনপুটের সাথে সংযুক্ত। L293 এর আউটপুট পিন 3 ও 6 এর সাথে মোটর সংযুক্ত।




চিত্র-১৩.৫ : 8051 ব্যবহার করে ডিসি মোটরের স্বয়ংক্রিয়ভাবে গতির দিক পরিবর্তন

**প্রোগ্রাম :**

```
        ORG 00H                 ; initial starting address
MAIN    MOV P1, #00000001B      ; motor runs clockwise
        ACALL DELAY             ; calls the 1S DELAY
        MOV P1,#00000010B       ; motor runs anticlockwise
        ACALL DELAY             ; calls the 1S DELAY
        SJMP MAIN               ; jumps to label MAIN for repeating the cycle
DELAY:  MOV R4,#0FH             ; 1s delay subroutine
WAIT1:  MOV R3,#00H
WAIT2:  MOV R2,#00H
WAIT3:  DJNZ R2, WAIT3
        DJNZ R3, WAIT2
        DJNZ R4, WAIT1
        RET
        END
```

## ১৩.২ 8051 এর সাথে স্টেপার মোটরের ইন্টারফেসিং

### (Describe Interfacing the 8051 With a Stepper Motor)

স্টেপার মোটর এমন একটি ইলেকট্রোমেকানিক্যাল ডিভাইস, যা একটি ব্রাসহীন, সিনক্রোনাস ইলেকট্রিক মোটর, এবং এটি ডিজিটাল পালসকে বিযুক্ত যান্ত্রিক গতিতে (discrete mechanical movement) রূপান্তর করে। যান্ত্রিক ঘূর্ণনে রূপান্তর করে যখন উপযুক্ত ক্রমে (Proper sequence) ইলেকট্রিক্যাল কমান্ড পালস প্রয়োগ করা হয়, তখন মোটরের প্রতিটি ঘূর্ণন কতকগুলো ধাপের বিযুক্ত (discrete) সংখ্যায় বিভক্ত হয়ে ক্রমবৃদ্ধিতে ঘুরে। একটি একক ধাপের কোণকে মোটরের স্টেপার কোণ বলে। এতে–

(১) প্রযুক্ত পালসের ক্রম বা সিকোয়েন্স মোটর Shaft-এর ঘূর্ণনের দিকের সাথে সরাসরি সম্পর্কিত।

(২) মোটর Shaft এর ঘূর্ণনগতি ইনপুট পালসের ফ্রিকুয়েন্সির সাথে সরাসরি সম্পর্কিত।

(৩) ঘূর্ণনের দৈর্ঘ্য প্রযুক্ত পালসের সংখ্যায় সাথে সম্পর্কিত।

স্টেপার মোটর সাধারণত তিন ধরনের হয়, যথা :

(i) Permanent Magnet Stepper
(ii) Variable Reluctance Stepper
(iii) Hybrid Step Motor.

Permanent Magnet Stepper-এ একটি স্ট্যাটর বেষ্টিত একটি রোটর বা shaft (স্থায়ী চুম্বক) থাকে। তবে variable reluctance স্টেপার মোটরে স্থায়ী চুম্বক রোটর থাকে না। অধিকাংশ স্টেপার মোটরে সাধারণত চারটি stator winding থাকে, যা একটি center-tap-এর সাথে জোড়া অবস্থায় থাকে। এ ধরনের স্টেপার মোটরকে চার ফেজ বা Unipolar stepper motor বলে বহুল ব্যবহৃত হয়। এটি প্রধানত পজিশন কন্ট্রোল এ ব্যবহৃত হয়। কোন ফিডব্যাক কৌশল ছাড়াই মোটর পজিশনকে খুব সূক্ষ্মভাবে নিয়ন্ত্রণ করা যায়।

স্টেপার মোটরের তিন step sequence আছে :

১। **Full step sequence :** দুই কয়েল একই সময়ে উত্তেজিত হয় এবং মোটর shaft ঘুরে।

২। **Half step sequence :** এক ও দুই কয়েল একই সময়ে উত্তেজিত হয়। মোটরের স্টেপার কোণ full mode-এর অর্ধেক হয়, ফলে angular resolution-ও বৃদ্ধি পায় এবং Full mode-এর দ্বিগুণ হয়। এতে ঘূর্ণনের ধাপ সংখ্যা full mode এর দ্বিগুণ। তাই এটি বেশি ব্যবহার করা হয়।

৩। **Micro-step Sequence :** একই সময়ে এক কয়েল উত্তেজিত হয়।

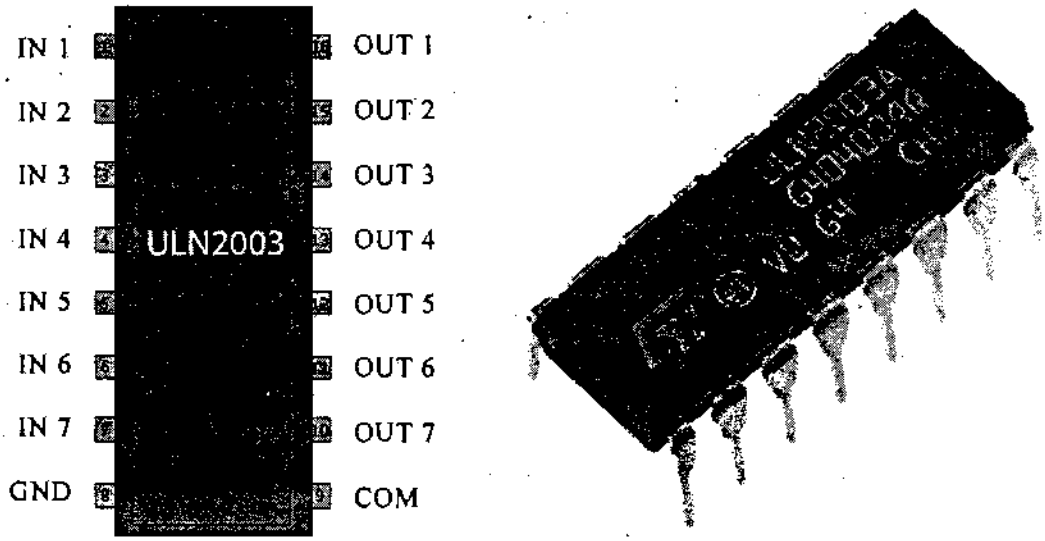
**Full mode sequence (4 steps)**

| Step | A | B | $\bar{A}$ | $\bar{B}$ |
|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 2 | 0 | 0 | 1 | 1 |
| 3 | 1 | 0 | 0 | 1 |

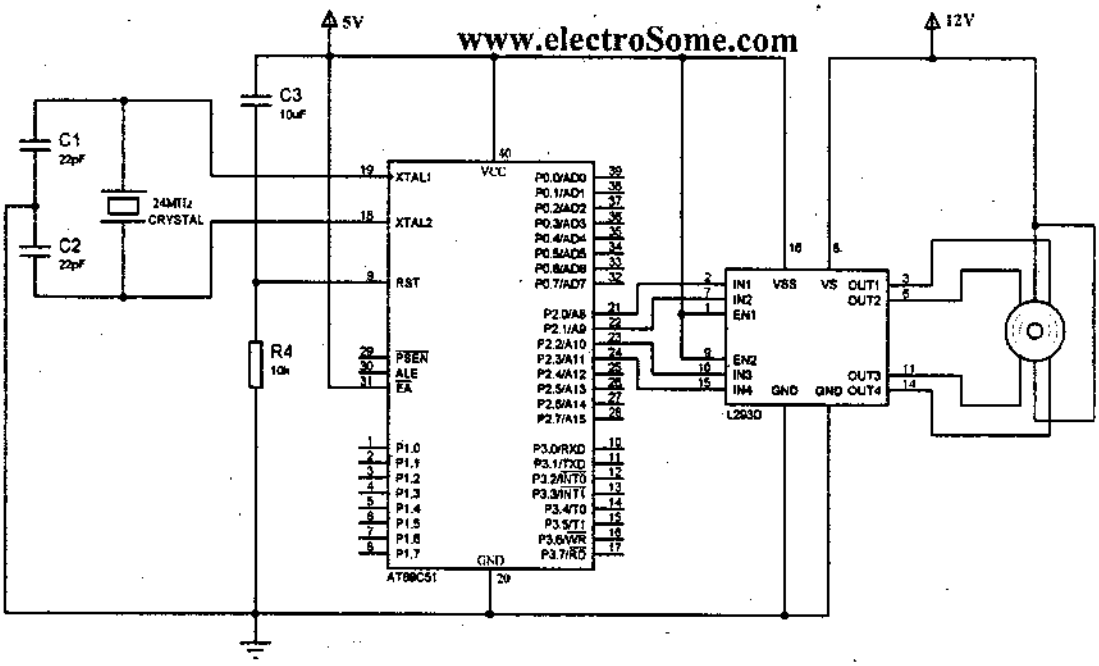
**Half mode sequence (8 steps)**

| Step | A | B | $\bar{A}$ | $\bar{B}$ |
|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 1 | 1 |
| 5 | 0 | 0 | 0 | 1 |
| 6 | 1 | 0 | 0 | 1 |
| 7 | 1 | 0 | 0 | 0 |

L293D বা ULN2003 ব্যবহার করে এটিকে সহজে 8051 এর সাথে ইন্টারফেস করা হয়। নিচের চিত্র ULN2003 ব্যবহার করে স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং দেখানো হলো। ULN2003 একটি উচ্চ কারেন্টর ডারলিংটন পেয়ার কারেন্ট ড্রাইভার আইসি। এতে সাতটি কমন ইমিটারের ওপেন কালেক্টরের ডারলিংটন পেয়ার আছে। এর কারেন্ট রেটিং 500 mA, তবে 600 mA পর্যন্ত সহ্য করতে পারে। ULN2003-এর পিন ডায়াগ্রাম চিত্র ১৩.৬-এ দেখানো হলো :




চিত্র-১৩.৬ : ডারলিংটন পেয়ার কারেন্ট ড্রাইভার




চিত্র-১৩.৭ : স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং

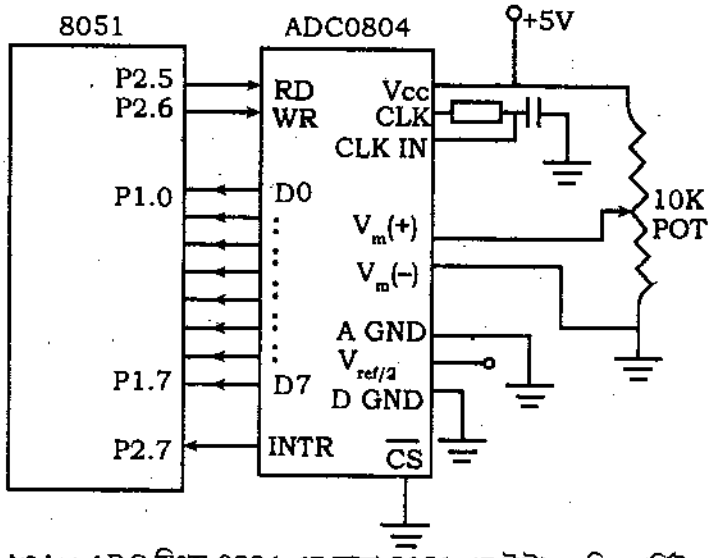
প্রোগ্রাম-৩ : নিচে Full step sequence-এ স্টেপারে মোটরের ইন্টারফেসিং প্রোগ্রাম উল্লেখ করা হলো।

```
        ORG 0H
        STEPPER EQUP1
MAIN :
        MOV STEPPER,#0CH        ; move the code to phase I into the port
        ACALL DELAY             ; Phase II code
        MOV STEPPER,#03H
        ACALL DELAY
        MOV STEPPER,#09H        : Phase IV code
        ACALL DELAY
        SJMP MAIN
DELAY : MOV R7,#4
WAIT1 : MOV R5,#0FFH
WAIT :  DJNZ R5, WAIT
        DJNZ R5, WAIT1
        DJNZ R7, WAIT2
        RET
        END
```

## ১৩.৩ 8051 এর এডিসি চিপ্‌স এর ইন্টারফেসিং বর্ণনা

## (Describe Interfacing ADC Chips to the 8051)

ADC চিপ্‌স 0804 একটি একক চ্যানেল analogue to digital converter. অর্থাৎ এটি শুধুমাত্র একটি অ্যানালগ সিগন্যাল গ্রহণ করে। এর রেজোলুশন 8 বিট, ফলে এর step size 19.53 ভোল্ট (5 ভোল্ট/255)। অ্যানালগ থেকে ডিজিটালে রূপান্তরে প্রয়োজনীয় সময় নির্ভর করে ক্লক ফ্রিকুয়েন্সির উপর। ADC চিপ্‌স 0804-এর রূপান্তর সময় প্রায় 110 মাইক্রোসেকেন্ড। আভ্যন্তরীণ ক্লক ব্যবহারের জন্য সার্কিটে একটি ক্যাপাসিটর ও রেজিস্টার ব্যবহার করা হয়। একটি রেগুলেটেড পাওয়ার সাপ্লাই ও একটি 10K পটেনশিওমিটার হতে ADC-তে ইনপুট দেয়া হয়। 8051 মাইক্রোকন্ট্রোলার ADC-তে কন্ট্রোল সিগন্যাল সরবরাহ করে। ADC এর $\overline{CT}$ (Chip select) পিন সরাসরি গ্রাউন্ড করা হয়। পিন P1.1, P1.0 ও P1.2 কে যথাক্রমে ADC এর WR, RD ও INTR সিগন্যালের সাথে যুক্ত করা হয়।




চিত্র-১৩.৮ : ADC চিপ্‌স 0804-এর সাথে 8051 এর ইন্টারফেসিং সার্কিট

এখানে, P1 পোর্ট ADC থেকে 8 বিটের ডিজিটাল ইনপুট গ্রহণ করে। P2.5 ও P2.6 পিনদ্বয় ADC এর SOC (Start of Conversion) এবং EOC (End of Conversion)-এর জন্য ব্যবহার করা হয়। যখন রূপান্তর প্রক্রিয়া শেষ হয় তখন ADC মাইক্রোকন্ট্রোলার P2 পিনের মাধ্যমে একটি ইন্টারাপ্ট সিগন্যাল প্রদান করে।

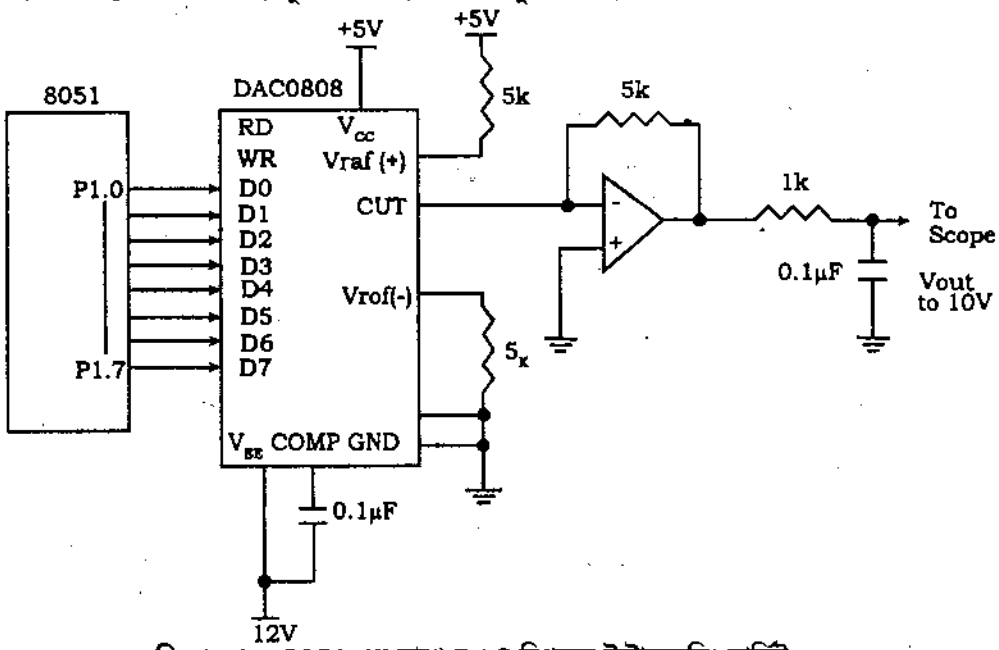
**প্রোগ্রাম- 8 :** P1 পোর্টে অ্যানালগ ইনপুট দিয়ে P0 পোর্টে ডিজিটাল আউটপুট পাওয়ার জন্য 8051-এর অ্যাসেম্বলি প্রোগ্রাম :

```
        ORG 00H
        MOV P1, #11111111B   ; initiates P1 as the input port
MAIN :
        SETB P2.5            ; makes RD high (P2.5)
        CLR P2.6             ; makes WR low P2.6
        SETB P2.6            ; low to high pulse to WR for starting conversion
WAIT :  JB P2.7, WAIT        ; polls until INTR = 0
        CLR P2.5             ; high to low pulse to RD for reading the data from ADC
        MOV A, P1            ; moves the digital data to accumulator
        CPL A                ; complements the digital data
        MOV P0, A            ; outputs the data to P0
        SJMP MAIN            ; jumps back to the MAIN program
        END
```

## ১৩.৪ 8051 এর সাথে ডিএসি চিপ্স এর ইন্টারফেসিং

## (Describe Interfacing the DAC Chips to the 8051)

DAC চিপ্স 0800 একটি সরল মনোলিথিক 8 বিটের D/A কনভার্টার। এর কনভার্শন টাইম খুব কম, মাত্র 100 ns। এটি 4.5 থেকে 18 V-এ চলে। এটি TTL, CMOS, PMOS বা অন্যান্য লজিকের সাথে সহজেই ইন্টারফেস হতে পারে। যেহেতু অ্যানালগ আউটপুটের লেবেল সংখ্যা $2^n$ (n হলো ডিজিটাল ইনপুট বিটের সংখ্যা) এর সমান তাই ডাটা ইনপুটের সংখ্যা DAC-এর রেজোলুশন নির্ধারণ করে। সুতরাং একটি 8 বিট ইনপুট DAC (DAC 0808) 256টি বিযুক্ত আউটপুট ভোল্টেজ বা কারেন্ট লেবেল উৎপন্ন করে। নিচের চিত্রে এ 8051 এর সাথে DAC ইন্টারফেসিং সার্কিট দেখানো হলো। এতে মাইক্রোকন্ট্রোলার হতে ডিজিটাল 8 বিট P1 পোর্টের মাধ্যমে DAC 0808-এর ইনপুট ডাটা লাইনের সাথে যুক্ত করা হয়।




চিত্র-১৩.৯ : 8051 এর সাথে DAC চিপ্সের ইন্টারফেসিং সার্কিট

**প্রোগ্রাম- ৫ :** ডিজিটাল সিগন্যালকে অ্যানালগ সিগন্যালে রূপান্তরের প্রোগ্রাম :

```
        MOV     A,#DATA; (A)    = #.data
START :
        MOV     90H, A          ; Port 1 = (A)
        INC     A
        LJMP    START           ; Repeat
```

## ১৩.৫ 8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা

## (Describe Interfacing Temperature Sensor to the 8051)

টেম্পারেচার সেন্সর হিসেবে বহুল ব্যবহৃত সেন্সরের তালিকা নিচে দেয়া হলো :

**(ক) অ্যানালগ টেম্পারেচার সেন্সর**

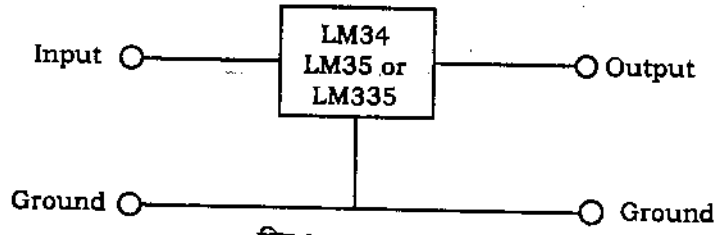
**ভোল্টেজ আউটপুট :**

১। LM34 : Fahrenheit sensor (10 mV/Fahrenheit)

২। LM35 : Celsius sensor (10 mV/Celsius)

৩। LM335 : Kelving sensor (10mV/Kelving)

এগুলো সাধারণত 3 পিনের ডিভাইস, যেমন– পাওয়ার, গ্রাউন্ড এবং আউটপুট।




চিত্র-১৩.১০ :

**কারেন্ট আউটপুট :**

* AD590 – Kelvin sensor (1μA/Kelvin) সাধারণত দুই পিনের ডিভাইস

**(খ) ডিজিটাল :**

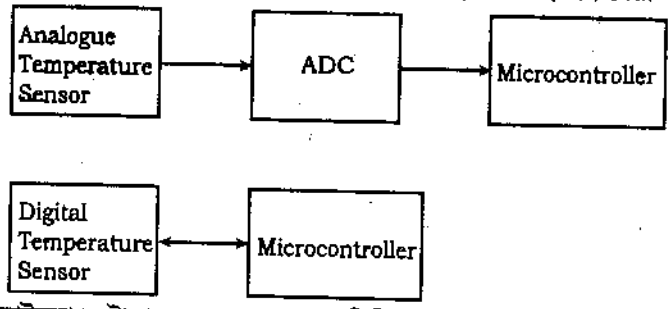
(a) ফ্রিকুয়েন্সি আউটপুট- MAX6576

(b) 1 Wire- DBS18B20

(c) 2 Wire/SMBUS – DS1621

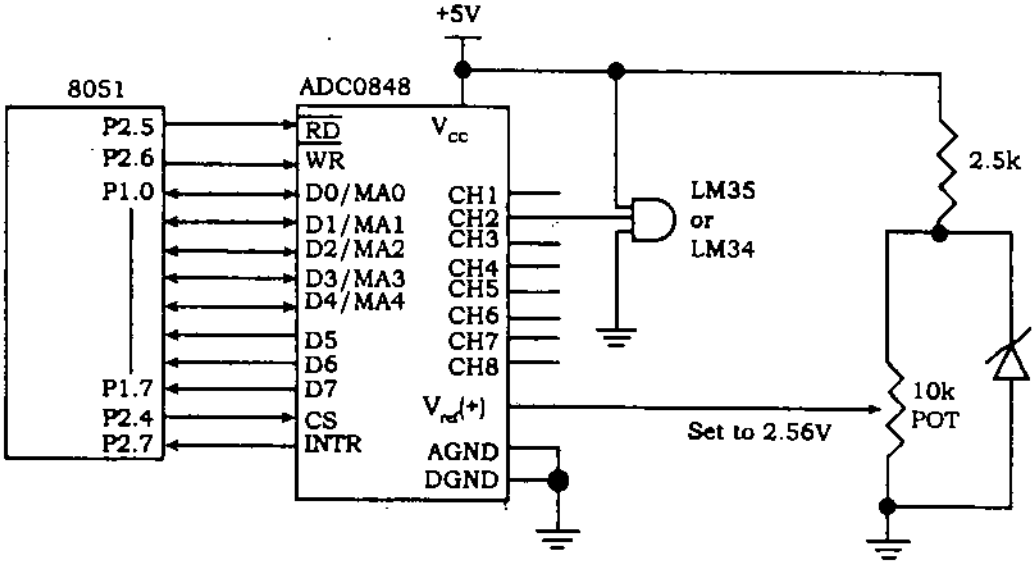
(d) 3 Wire – DS1620

**মাইক্রোকন্ট্রোলারের সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং :** অ্যানালগ টেম্পারেচার সেন্সর থেকে প্রাপ্ত অ্যানালগ সিগন্যালকে প্রথমে একটি ADC এর মাধ্যমে ডিজিটাল রূপান্তর করে মাইক্রোকন্ট্রোলারে প্রয়োগ করা হয়, তবে ডিজিটাল সেন্সরের ক্ষেত্রে সেন্সর আউটপুটকে সরাসরি মাইক্রোকন্ট্রোলারে সংযোগ দেয়া যায়।

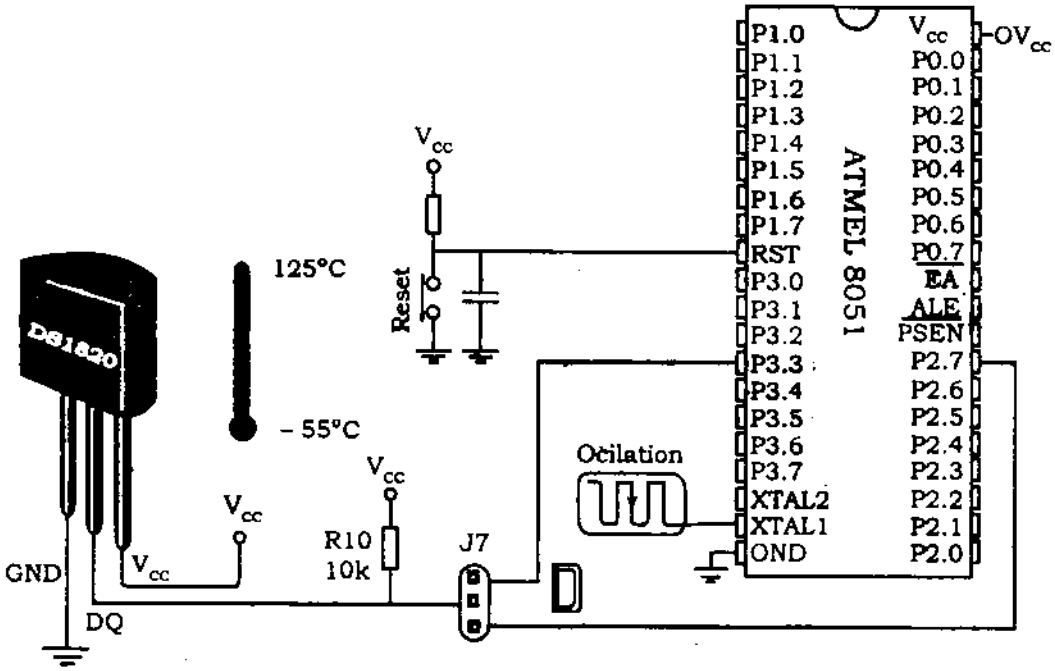



চিত্র-১৩.১১ : মাইক্রোকন্ট্রোলারের সাথে অ্যানালগ/ডিজিটাল টেম্পারেচার সেন্সর ইন্টারফেসিং

**LM35-এর সাথে 8051-এর ইন্টারফেসিং :** LM35 সিরিজ একটি টেম্পারেচার সেন্সর, যার আউটপুট ভোল্টেজ সেন্টিগ্রেড টেম্পারেচারের সাথে রৈখিক সমানুপাতিক। এর টেম্পারেচার রেঞ্জ – 55°C থেকে + 150°C। এর নিম্ন আউটপুট ইম্পিডেন্স, রৈখিক আউটপুট এবং সূক্ষ্ম ক্যালিব্রেশন একে ইন্টারফেসিং-এ সহজ করে। নিচে ADC 0848 ব্যবহার করে LM35-এর সাথে 8051 এর ইন্টারফেসিং দেখানো হলো। জিনার ডায়োড (LM336) 10k-এর আড়াআড়িতে ভোল্টেজকে 2.5V-এ স্থির রাখে। এটি পাওয়ার সাপ্লাই এর উঠানামা বন্ধ রাখে।




চিত্র-১৩.১২ : 8051 এর সাথে অ্যানালগ টেম্পারেচার সেন্সরের (LM34/35) ইন্টারফেসিং




চিত্র-১৩.১৩ : 8051 এর সাথে ডিজিটাল টেম্পারেচার সেন্সরের (DS1820) এর ইন্টারফেসিং

**প্রোগ্রাম- ৬ :** কিছু ডিলেসহ কোন টেম্পারেচার সেন্সরের পাঠ গ্রহণ ও P0 পোর্টে আউটপুট পাওয়ার জন্য 8051 প্রোগ্রাম : (চিত্র : ১৩.১১)

```
; Assembly code to read temperature, convert it and put it on P0 with some delay
        RD      BIT             P2.5    ; RD
        WR      BIT             P2.6    ; WR (Start conversion)
        INTR    BIT             P2.7    ; end-of-conversion (EOC)
        MYDATA  EQU             P1      ; P1.0–P1.7 = D0–D7 of the ADC0848
        MOV     P1.#0FFH                ; make P1 = input
        SETB    INTR
BACK :  CLR     WR                      ; WR = 0
        SETB    WR                      ; WR = 1, L-to-H to starting of
                                          conversion (SOC)
HERE :  JB      INTR, HERE              ; wait for end of conversion
        CLR     RD                      ; conversion tinished, enable RD
        MOV     A, MYDATA               ; read the data for ADC0848
        ACALL   CONVERSION              ; hexx-to-ASCII conversion
        ACALL   DATA_DISPLAY            ; display the data
        SETB    RD                      ; make RD = 1 for next round
        SJMP    BACK
CONVERSION :
 MOV            B, #10
 DIV            AB
 MOV            R7, B       ; least significant byte
 MOV            B, #10
 DIV            AB
 MOV            R6, B
 MOV            R5, A       ; most significant byte
 RET
DATA_DISPLAY :
 MOV            P0, R7
 ACALL          DELAY
 MOV            P0, R6
 ACALL          DELAY
 MOV            P0, R5
 ACALL          DELAY
 RET
```

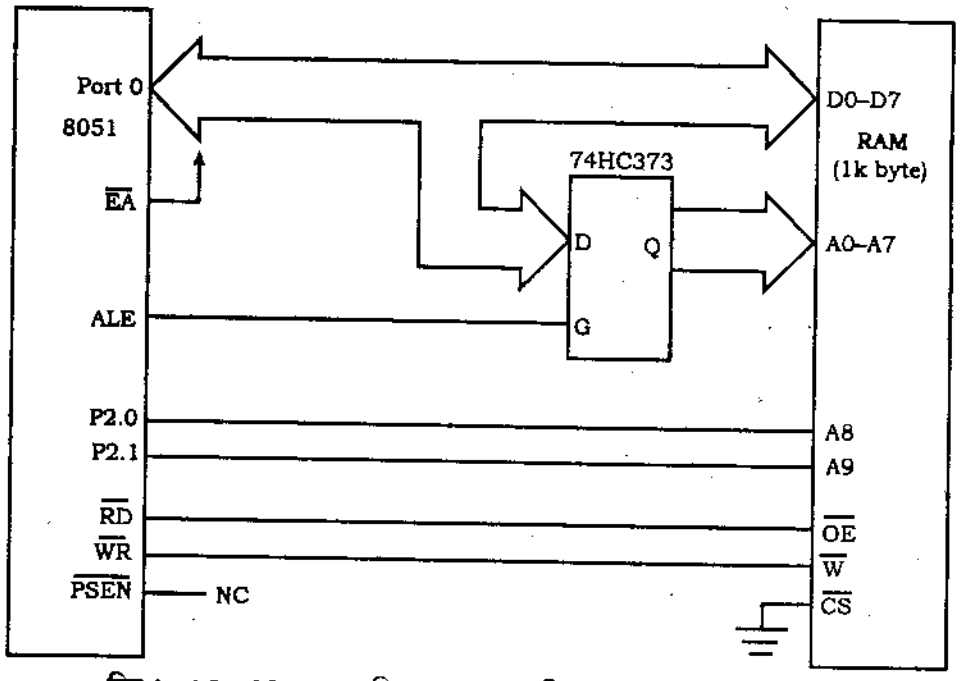
## ১৩.৬ 8051 এর সাথে বাহ্যিক মেমোরির ব্যবহার

### (Develop Program for Above Interfacing Circuits)

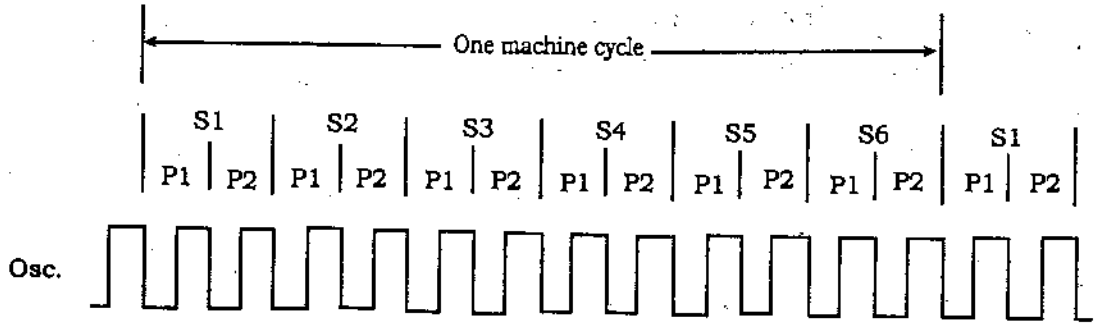
কখনো যদি অতিরিক্ত RAM ও ROM-এর দরকার হয়, তবে MCS-51 আর্কিটেকচারে বাহ্যিকভাবে 64 কিলোবাইট প্রোগ্রাম মেমোরি ও 64 কিলোবাইট ডাটা মেমোরি পর্যন্ত সংযোগের ব্যবস্থা রয়েছে। যখন বাহ্যিক মেমোরি ব্যবহার করা হয় তখন Port 0 ইনপুট/আউটপুট পোর্ট হিসেবে ব্যবহার হয় না। কেননা তখন এটি multiplexed address (A0–A7) এবং ডাটা (D0–D7) বাস হিসেবে ব্যবহার করা হয়। প্রত্যেক মেমোরি সাইকেলের শুরুতে ALE অ্যাড্রেসের Low byte কে Latching করে। পোর্ট P2 অ্যাড্রেস বাসের High byte

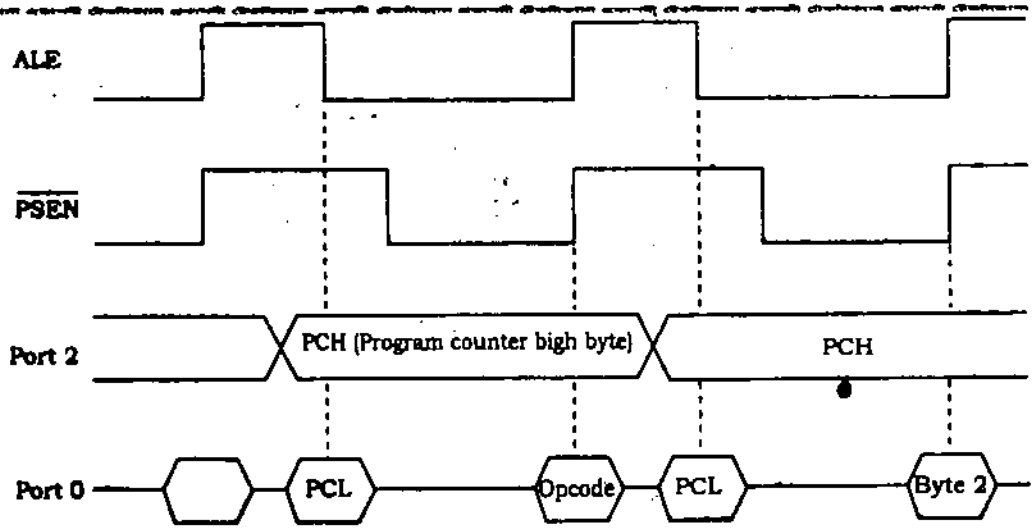
কে ধারণ করে। Multiplexed ব্যবস্থায় মেমোরি সাইকেলের প্রথম অর্ধে P0 তে অ্যাড্রেসের Low byte সরবরাহ করা হয় এবং ALE সিগন্যাল দ্বারা তাকে ধরে রাখা হয়। এখানে latch হিসেবে 74HC373 বা সমতুল্য ব্যবহার করা হয়। মেমোরি সাইকেলের দ্বিতীয় অর্ধে P0 ডাটা বাস হিসেবে ব্যবহার হয় এবং অপারেশনের ভিত্তিতে লিখন বা পঠন হয়।

**8051-এ বাহ্যিক কোড মেমোরির ব্যবহার :** 8051 এর $\overline{PSEN}$ সিগন্যাল দ্বারা বাহ্যিক কোড মেমোরি (EPROM) কার্যকরী হয়। যখন একটি বাহ্যিক EPROM ব্যবহার হয়, তখন পোর্ট P0 ও P2 কে ইনপুট/আউটপুট পোর্ট হিসেবে পাওয়া যায় না। বাহ্যিক EPROM-এর হার্ডওয়্যার সংযোগ নিচের চিত্রে দেখানো হলো। 8051 এর এক মেশিন সাইকেল হলো 12 অসিলেটর সাইকেল। যদি অন চিপ অসিলেটর 12 মে. হার্টসের ক্রিস্টাল দ্বারা পরিচালিত হয়, তবে এক মেশিন সাইকেলের সময়কাল 1 মাইক্রোসেকেন্ড। এক মেশিন সাইকেলে ALE দুবার পালস প্রদান করে এবং প্রোগ্রাম মেমোরি থেকে দুই বাইট পড়ে। এই অপারেশনের টাইমিং ডায়াগ্রাম চিত্র ১৩.১৫ তে দেখানো হলো :



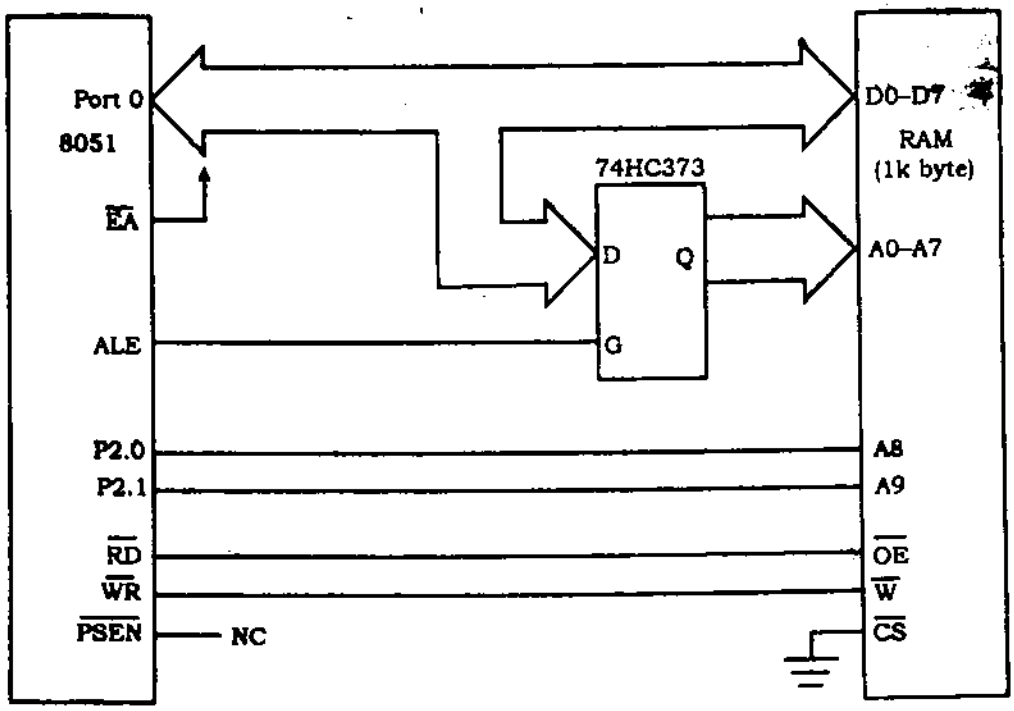

চিত্র-১৩.১৪ : 8051-এ বাহ্যিক কোড মেমোরির (EPROM) ব্যবহার



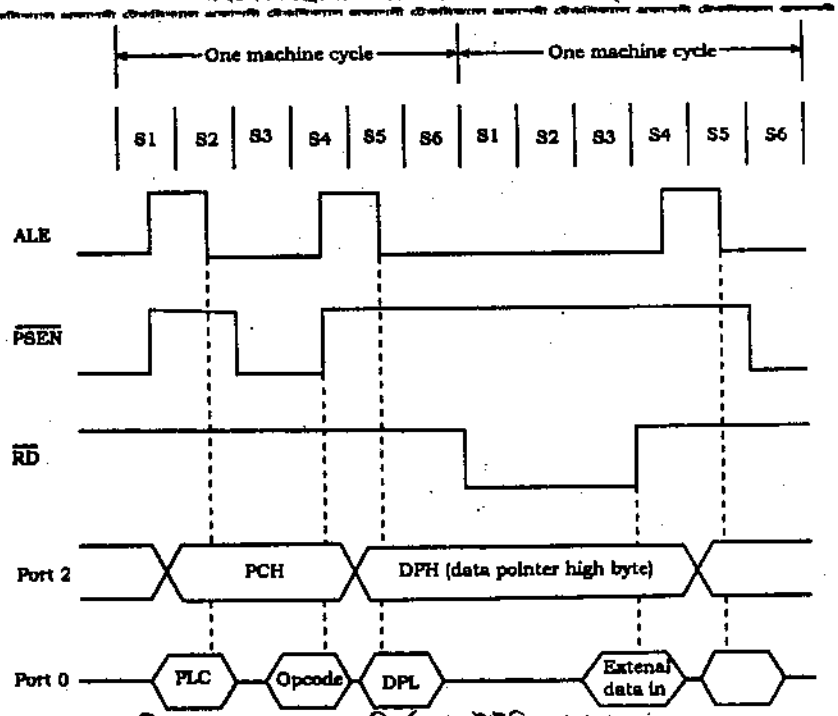



চিত্র-১৩.১৫ : 8051-এ বাহ্যিক কোড মেমোরির পঠনের টাইমিং ডায়াগ্রাম

**8051-এ বাহ্যিক ডাটা মেমোরির ব্যবহার :** বাহ্যিক ডাটা মেমোরি (RAM) হলো একটি লিখন/পঠন মেমোরি, যা 8051 এর পোর্ট পিন P3.7 ও P3.6-এর যথাক্রমে $\overline{RD}$ ও $\overline{WR}$ সিগন্যালকে যথাক্রমে RAM-এর $\overline{OE}$ ও $\overline{W}$ পিনের সাথে সংযোগ EPROM এর মতো। পোর্ট P0 ও P2 কে ব্যবহার করে 8051 এর সাথে 64 কিলোবাইট পর্যন্ত বাহ্যিক ডাটা মেমোরি সংযোগ দেয়া যায়। বাহ্যিক ডাটা মেমোরির সাথে 8051 সংযোগ নিচের চিত্রে এবং MOVX নির্দেশনার টাইমিং ডায়াগ্রাম ১৩.১৭ তে দেখানো হলো :



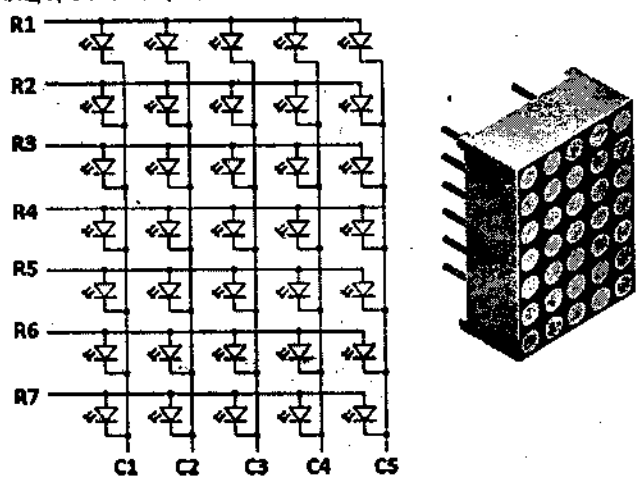

চিত্র-১৩.১৬ : 8051-এ বাহ্যিক ডাটা মেমোরির (RAM) ব্যবহার




চিত্র-১৩.১৭ : MOVX নির্দেশনার টাইমিং ডায়াগ্রাম

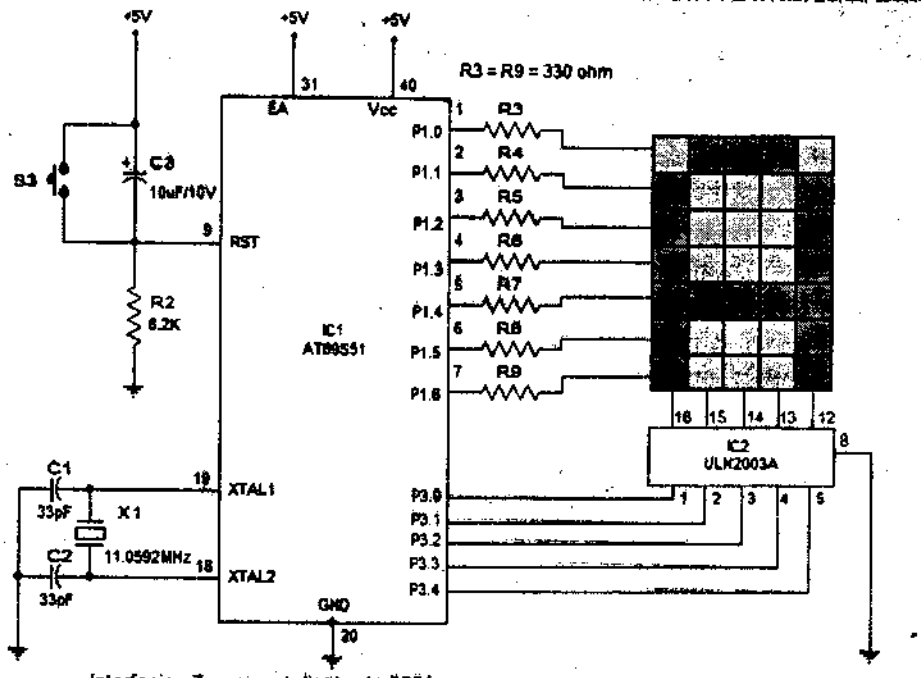
## ১৩.৭ ডট ম্যাট্রিক্স ডিসপ্লে এর সাথে 8051 এর ইন্টারফেসিং (Explain How to Use Off-chip Memory With the 8051)

একটি আয়তাকার LED matrix নিয়ে LED dot matrix গঠিত। আকাঙ্ক্ষিত অক্ষর বা গ্রাফিক্স LED-এর সজ্জাকে সুইচিং করে প্রদর্শন করা হয়। সাধারণ ডিসপ্লে সজ্জা হলো 7 × 5, 8 × 8, 7 × 15 ইত্যাদি। নিচে একটি 7 × 5 ডট ম্যাট্রিক্স দেখানো হলো :




চিত্র-১৩.১৮ : 7 × 5 Led dot matrix display

কোন স্বতন্ত্র LED বা LED এর একটি গ্রুপকে কার্যকরী করার জন্য প্রয়োজনীয় সংখ্যক সারি বা কলামকে সুইচিং করা হয়। যেমন– Row 1 কে High এবং Column 1 কে Low করলে উপরের LED (R1C1) জ্বলবে। এভাবে যে কোন অক্ষর বা গ্রাফিক্স ডিসপ্লে করা হয়। নিচের চিত্রে 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং দেখানো হলো :




Interfacing 7 segment display to 8051

চিত্র-১৩.১৯ : 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং

## ১৩.৮ ডট মেট্রিক ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং বর্ণনা

**(Describe Interfacing the 8051 With a Dot Matrix Display)**

প্রোগ্রাম- ৭ : 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং প্রোগ্রাম (উপরের চিত্র)

```
        ORG 00H
        MOV P3,#00000000B        ; initializes port 3 as output port
        MOV P1,#00000000B        ; initializes port 1 as output port
MAIN :  MOV P3,#01111110B
        MOV P1,#11111110B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1,#11111101B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1#11111011B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1,#11110111B
        ACALL DELAY
        MOV P3,#01111110B
        MOV P1,#11101111B
        ACALL DELAY
        SJMP MAIN                ; jumps back to the main loop
DELAY:  MOV R6,#255D             ; 1 ms delay subroutine
HERE :  DJNZ R6, HERE
        RET
        END
```

# অনুশীলনী-১৩

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

**১। একটি ছোট ডিসি মোটরের starting কারেন্ট কত?**

**উত্তর :** একটি ছোট ডিসি মোটরের জন্য starting কারেন্ট প্রায় 200–300 mA.

**২। একটি H-Bridge driver আইসি এর নাম লিখ।**

**উত্তর :** L298 একটি Dual H-Bridge driver আইসি।

**৩। কীসের সাহায্যে ডিসি মোটরকে 8051 এর সাথে ইন্টারফেস করা হয়?**

**উত্তর :** অপটোকাপলার ও Dual H-Bridge driver (L298) নিয়ে গঠিত একটি ড্রাইভিং সার্কিটের সাহায্যে 8051 এর সাথে ডিসি মোটরকে ইন্টারফেসিং করা হয়।

**৪। L293 Motor Driver কী?**

**উত্তর :** L293 একটি 16 পিনের quadruple half H-bridge মোটর ড্রাইভ IC। এর কারেন্ট বহন ক্ষমতা প্রতি চ্যানেলে 600 mA এবং ভোল্টেজ রেঞ্জ 4.5–36 ভোল্ট (ডিসি)। ইন্ডাক্টিভ স্পাইক প্রতিরোধের জন্য এর অভ্যন্তরে উচ্চ গতির ক্ল্যাম্প ডায়োড আছে। তাছাড়া এর Noise immunity, ESD (Electro static discharge) প্রতিরোধ ক্ষমতা, Thermal shutdown অনেক বেশি এবং প্রতি চ্যানেলের পৃথক ইনপুট সাপ্লাই আছে।

**৫। 8051 এর সাথে ডিসি মোটর ইন্টারফেসিং সার্কিটে heat sink-এর প্রয়োজনীয়তা কী?**

**উত্তর :** মোটরের অবিরাম অপারেশনের ফলে ইন্টারফেসিং সার্কিটে L293 Dual H-Bridge ড্রাইভার overheat হয়, তাই একটি উপযুক্ত heat sink অবশ্যই ব্যবহার করতে হয়।

**৬। স্টেপার মোটর কী?**

**উত্তর :** স্টেপার মোটর একটি ব্রাসহীন, সিনক্রোনাস ইলেকট্রিক মোটর (ইলেকট্রোমেকানিক্যাল ডিভাইস), যা ডিজিটাল পালসকে বিযুক্ত যান্ত্রিক গতিতে (discrete mechanical movement) রূপান্তর করে। যান্ত্রিক ঘূর্ণনে রূপান্তর করে যখন উপযুক্ত ক্রমে (Proper sequence) ইলেকট্রিক্যাল কমান্ড পালস প্রয়োগ করা হয়, তখন মোটরের প্রতিটি ঘূর্ণন কতকগুলো ধাপের বিযুক্ত (discrete) সংখ্যায় বিভক্ত হয়ে ক্রমবৃদ্ধিতে ঘুরে।

**৭। স্টেপার কোণ কাকে বলে?**

**উত্তর :** স্টেপার মোটরের প্রতি একক ধাপের কোণকে মোটরের স্টেপার কোণ বলে।

**৮। স্টেপার মোটর কয় ধরনের হয়?**

**উত্তর :** স্টেপার মোটর সাধারণত তিন ধরনের হয়, যথা :

১। Permanent magnet stepper
২। Variable reluctance stepper
৩। Hybrid step motor.

**৯। Permanent magnet ও variable reluctance স্টেপার মোটরের পার্থক্য কী?**

**উত্তর :** Permanent magnet stepper-এ একটি স্ট্যাটর বেষ্টিত রোটর বা shaft (স্থায়ী চুম্বক) আছে, কিন্তু variable reluctance স্টেপার মোটরে স্থায়ী চুম্বক রোটর থাকে না।

**১০। স্টেপার মোটরে কয়টি step sequence থাকে ও কী কী?**

**উত্তর :** স্টেপার মোটরের তিনটি step sequence আছে। যথা :

১। **Full step sequence :** দুই কয়েল একই সময়ে উত্তেজিত হয় এবং মোটর shaft ঘুরে।

২। **Half step sequence :** এক ও দুই কয়েল একই সময়ে উত্তেজিত হয়। মোটরের স্টেপার কোণ full mode-এর অর্ধেক হয়, ফলে angular resolution-ও বৃদ্ধি পায় এবং Full mode-এর দ্বিগুণ হয়। এতে ঘূর্ণনের ধাপ সংখ্যা Full mode এর দ্বিগুণ। তাই এটি বেশি ব্যবহৃত হয়।

৩। **Micro-step Sequence :** একই সময়ে এক কয়েল উত্তেজিত হয়।

**১১। স্টেপার মোটরকে কীভাবে 8051 এর সাথে ইন্টারফেস করা হয়?**

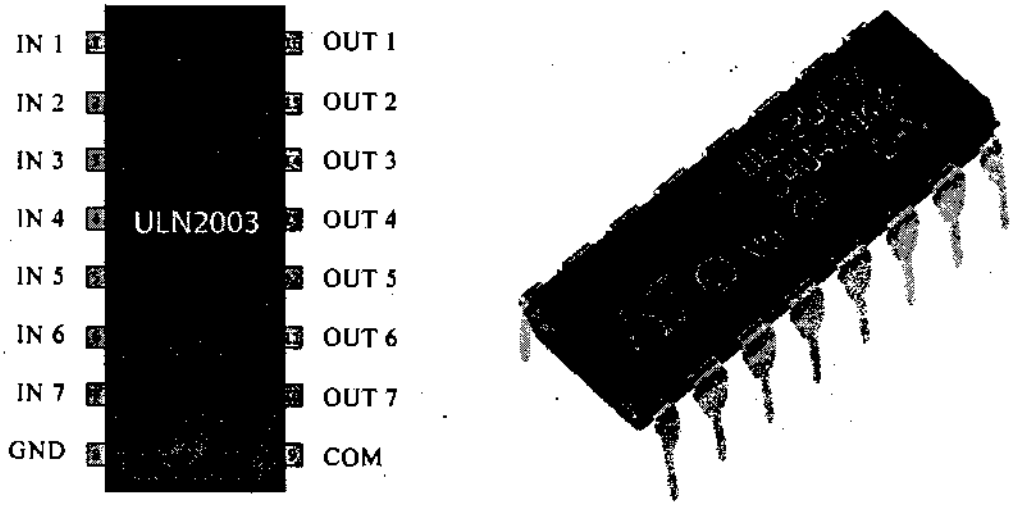
**উত্তর :** L293D বা ULN2003 ব্যবহার করে একে সহজে 8051 এর সাথে ইন্টারফেস করা হয়।

**১২। ULN2003 কী?**

**উত্তর :** ULN2003 একটি উচ্চ কারেন্টের ডারলিংটন পেয়ার কারেন্ট ড্রাইভার আইসি। এতে সাতটি কমন ইমিটারের ওপেন কালেক্টরের ডারলিংটন পেয়ার আছে। একে কারেন্ট রেটিং 500 mA এবং 600 mA পর্যন্ত সহ্য করতে হবে।

**১৩। ULN2003-এর পিন ডায়াগ্রাম অঙ্কন কর।**

**উত্তর :** ULN2003 এর পিন ডায়াগ্রাম নিচের চিত্রে দেখানো হলো :




চিত্র : ডারলিংটন পেয়ার কারেন্ট ড্রাইভার

**১৪। ডিজিটাল সিগন্যালকে অ্যানালগ সিগন্যালে রূপান্তরের প্রোগ্রাম লিখ।**

**উত্তর :** ডিজিটাল সিগন্যালকে অ্যানালগ সিগন্যালে রূপান্তরের প্রোগ্রাম :

```
        MOV     A, #DATA; (A)   = # data
START :
        MOV     90H, A          ; Port 1 = (A)
        INC     A
        LJMP    START           ; Repeat
```

**১৫। LM35 কী?**

**উত্তর :** LM35 সিরিজ একটি টেম্পারেচার সেন্সর, যার আউটপুট ভোল্টেজ সেন্টিগ্রেড টেম্পারেচারের সাথে রৈখিক সমানুপাতিক। এর টেম্পারেচার রেঞ্জ – 55°C থেকে + 150°C। এর নিম্ন আউটপুট ইম্পিডেন্স, রৈখিক আউটপুট এবং সূক্ষ্ম ক্যালিব্রেশন একে ইন্টারফেসিং-এ সহজ করে।

**১৬। অ্যানালগ টেম্পারেচার সেন্সরকে 8051 এর সাথে কীভাবে ইন্টারফেস করা যায়?**

**উত্তর :** একটি ADC (ADC0848) ব্যবহার করে টেম্পারেচার সেন্সর (LM35) কে 8051 এর সাথে ইন্টারফেস করা হয়।

**১৭। 8051 এর সাথে টেম্পারেচার ইন্টারফেসিং-এ জিনার ডায়োডের কাজ কী?**

**উত্তর :** 8051 এর সাথে টেম্পারেচার ইন্টারফেসিং-এ জিনার ডায়োড 10 কিলোওহম রেজিস্টারের আড়াআড়ি ভোল্টেজকে 2.5 ভোল্টে স্থির রাখে।

১৮। **ডিসি মোটরে কখন বেশি বিদ্যুৎ প্রয়োজন হয়?** *[বাকাশিবো : '১৫]*

**উত্তর :** চলন্ত অবস্থার চেয়ে প্রাথমিক অবস্থায় টর্ক উৎপন্ন করার জন্য ডিসি মোটরের অধিক কারেন্ট প্রয়োজন হয়।

১৯। **ডিসি মোটর কোন ধরনের কারেন্ট বলে?**

**উত্তর :** সিসি মোটর সিসি কারেন্টে চলে।

২০। **8051 এর সাথে ডিসি মোটরের ইন্টারফেসিং কিভাবে করা হয়।**

**উত্তর :** অপটোকাপলার ও Dual H-Bridge driver (L298) নিয়ে গঠিত একটি ড্রাইভিং সার্কিটের সাহায্যে 8051 এর সাথে ডিসি মোটরকে ইন্টারফেসিং করা হয়।

২১। **H-ব্রিক কি?**

**উত্তর :** H-Bridge ট্রানজিস্টর বা MOSFET দ্বারা তৈরি একটি ব্রিজ। এটি মোটরের গতি নিয়ন্ত্রণ করে। এটিকে "Full Bridge"- ও বলে। এতে চারটি সুইচিং উপাদান রয়েছে।

২২। **স্টেপার মোটর কি**

**উত্তর :** স্টেপার মোটর এমন একটি ইলেকট্রোমেকানিক্যাল ডিভাইস, যা একটি ব্রাসহীন, সিনক্রোনাস ইলেকট্রিক মোটর, এবং এটি ডিজিটাল পালসকে বিযুক্ত যান্ত্রিক গতিতে (discrete mechanical movement) রূপান্তর করে।

## সংক্ষিপ্ত প্রশ্নোত্তর

১। **বিভিন্ন প্রকার টেম্পারেচার সেন্সরের নাম লিখ।** *[বাকাশিবো : '১৪R, '১৫]*

**উত্তর :** টেম্পারেচার সেন্সর হিসেবে বহুল ব্যবহৃত সেন্সরের তালিকা নিচে দেয়া হলো :

(ক) **অ্যানালগ টেম্পারেচার সেন্সর**

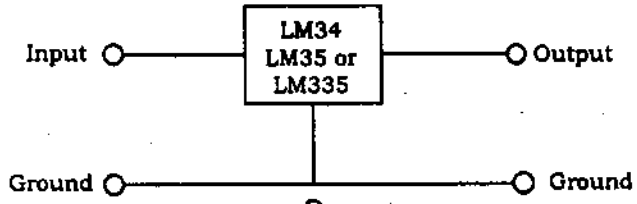
**ভোল্টেজ আউটপুট :**

১। LM34 : Fahrenheit sensor (10 mV/Fahrenheit)

২। LM35 : Celsius sensor (10 mV/Celsius)

৩। LM335 : Kelving sensor (10mV/Kelving)

এগুলো সাধারণত ৩ পিনের ডিভাইস, যেমন– পাওয়ার, গ্রাউন্ড এবং আউটপুট।




চিত্র

**কারেন্ট আউটপুট :**

* AD590 – Kelvin sensor (1μA/Kelvin) সাধারণত দুই পিনের ডিভাইস

(খ) **ডিজিটাল :**

(a) ফ্রিকুয়েন্সি আউটপুট- MAX6576

(b) 1 Wire – DBS18B20

(c) 2 Wire/SMBUS– DS1621

(d) 3 Wire – DS1620

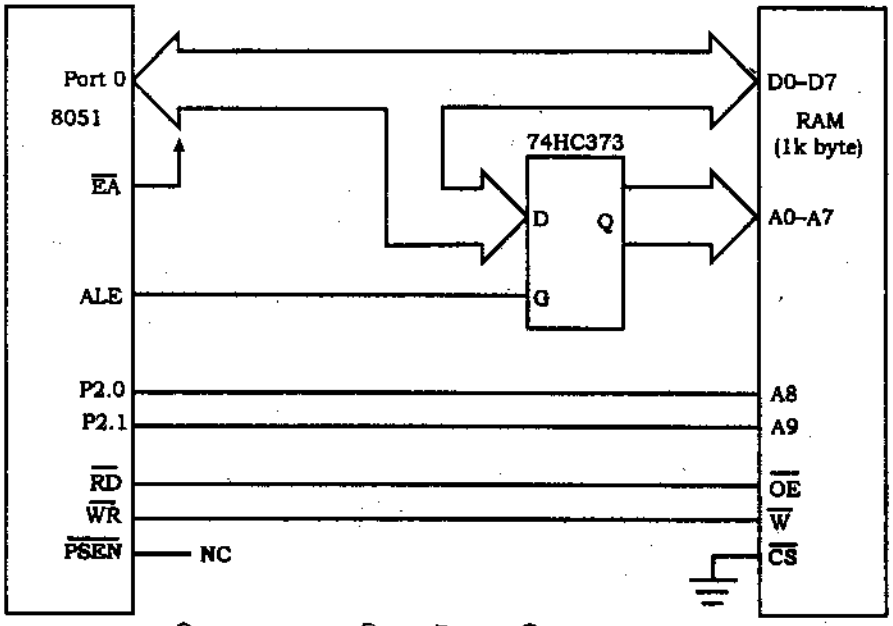
২। **8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা কর।** *[বাকাশিবো : '১৪, '১৪R, '১৫]*

**উত্তর :** অনুচ্ছেদ ১৩.৫ নং দ্রষ্টব্য।

৩। **8051 এর সাথে RAM-এর ইন্টারফেসিং দেখাও।** *[বাকাশিবো : '১৪]*

**উত্তর :** 8051 এর সাথে RAM-এর ইন্টারফেসিং দেখানো হলো :

বাহ্যিক ডাটা মেমোরি (RAM) হলো একটি লিখন/পঠন মেমোরি, যা 8051 এর পোর্ট পিন P3.7 ও P3.6-এর যথাক্রমে $\overline{RD}$ ও $\overline{WR}$ সিগন্যালকে যথাক্রমে RAM-এর $\overline{OE}$ ও $\overline{W}$ পিনের সাথে সংযোগ EPROM এর মতো। পোর্ট P0 ও P2 কে ব্যবহার করে 8051 এর সাথে 64 কিলোবাইট পর্যন্ত বাহ্যিক ডাটা মেমোরি সংযোগ দেয়া যায়। বাহ্যিক ডাটা মেমোরির সাথে 8051 সংযোগ নিচের চিত্রে এবং MOVX নির্দেশনার টাইমিং ডায়াগ্রাম ১৩.১৭ তে দেখানো হলো।




চিত্র : 8051-এ বাহ্যিক ডাটা মেমোরির (RAM) ব্যবহার

৪। **ডিসি মোটরকে সরাসরি 8051 এর সাথে ইন্টারফেস করা যায় না কেন?**

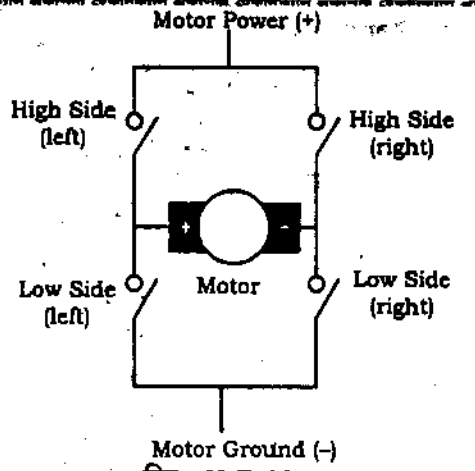
**উত্তর :** 8051 এর সাথে ডিসি মোটরের ইন্টারফেসিং সরাসরি সম্ভব নয়। কারণ ডিসি মোটরকে চালনার জন্য অধিক কারেন্ট (ছোট ডিসি মোটরের জন্য এর প্রায় 200–300 mA) প্রয়োজন। যদি এই কারেন্ট 8051 মাইক্রোকন্ট্রোলার দিয়ে প্রবাহিত হয় তবে 8051 IC নষ্ট হয়ে যাবে। অধিকন্তু মোটরের গতি বিপরতিমুখী হওয়ার সময় Voltage spike উৎপন্ন করে, যা মাইক্রোকন্ট্রোলারকে খুব সহজেই নষ্ট করে ফেলে। তাই ডিসি মোটরকে সরাসরি 8051 এর সাথে ইন্টারফেস করা যায় না।

৫। **Truth table সহ H-Bridge-এর Schematic ডায়াগ্রাম অঙ্কন কর।**

**উত্তর :** নিচে Truth table সহ H-Bridge-এর Schematic ডায়াগ্রাম অঙ্কন করা হলো :

**Truth Table**

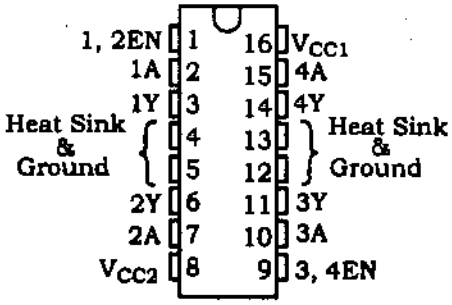
| High Left | High Right | Low Left | Low Right | বর্ণনা |
|---|---|---|---|---|
| On | Off | Off | On | মোটর ক্লকওয়াইজ ঘুরে। |
| Off | On | On | Off | মোটর ক্লকওয়াইজের বিপরীত ঘুরে। |
| On | On | Off | Off | মোটর থেমে যায় বা গতি হ্রাস পায়। |
| Off | Off | On | On | মোটর থেমে যায় বা গতি হ্রাস পায়। |




চিত্র : H-Bridge

৬। **L293-এর পিন ডায়াগ্রাম ও Truth table অঙ্কন কর।**

**উত্তর :** নিচে L293-এর পিন ডায়াগ্রাম ও Truth table অঙ্কন করা হলো :




Function Table (each driver)

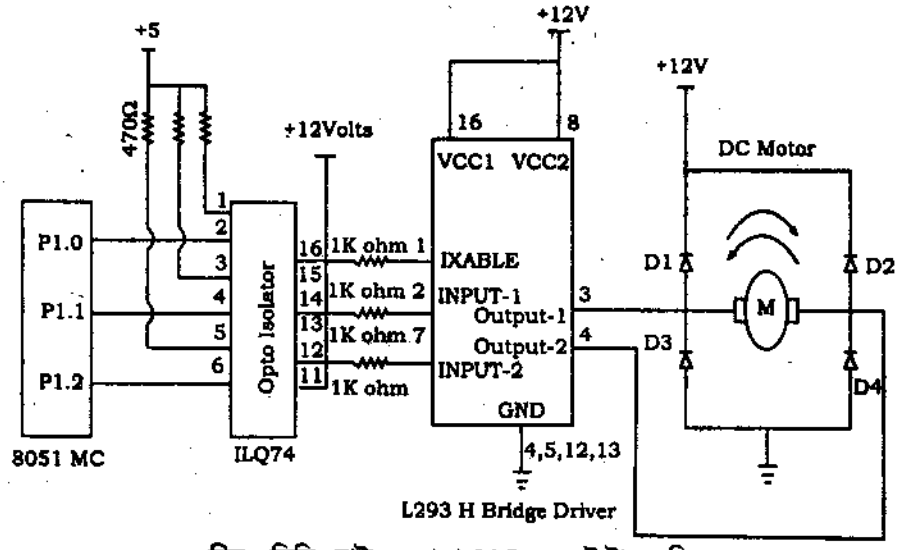
| Inputs | | Output |
|---|---|---|
| A | EN | Y |
| H | H | H |
| L | H | L |
| X | L | Z |

H = high level, L = low level,
X = Irrelevent,
Z = High Impedance (Off)

চিত্র : L293 পিন ডায়াগ্রাম ও Truth table

৭। **ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন কর।**

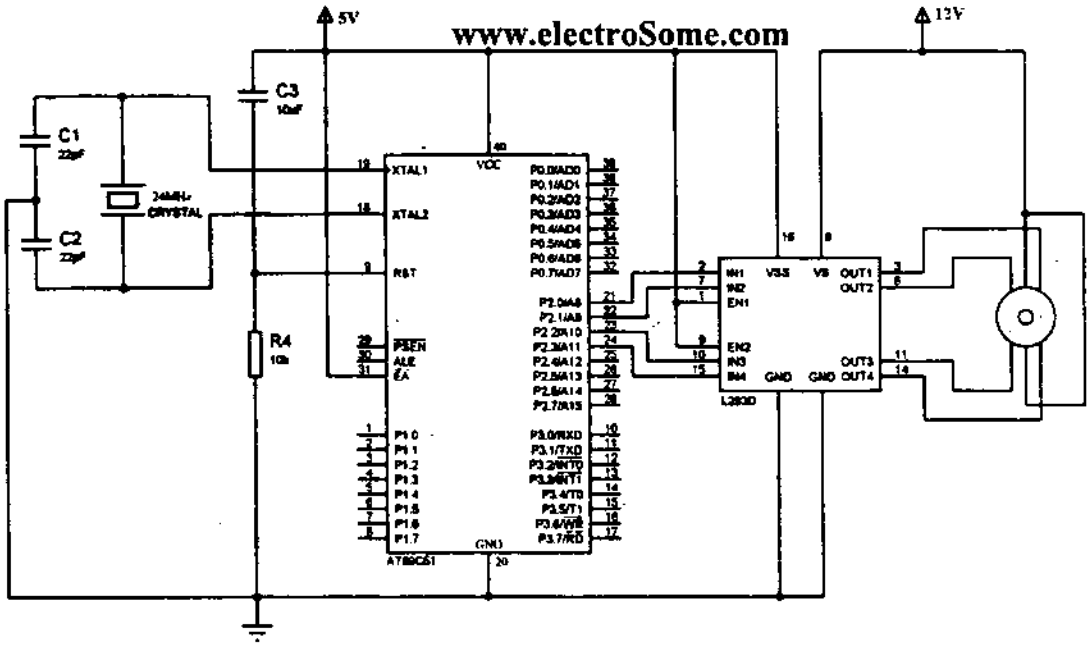
**উত্তর :** নিচে ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন করা হলো :




চিত্র : ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং

**৮। স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন কর।**

**উত্তর :** নিচে স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন করা হলো :




**চিত্র :** স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং

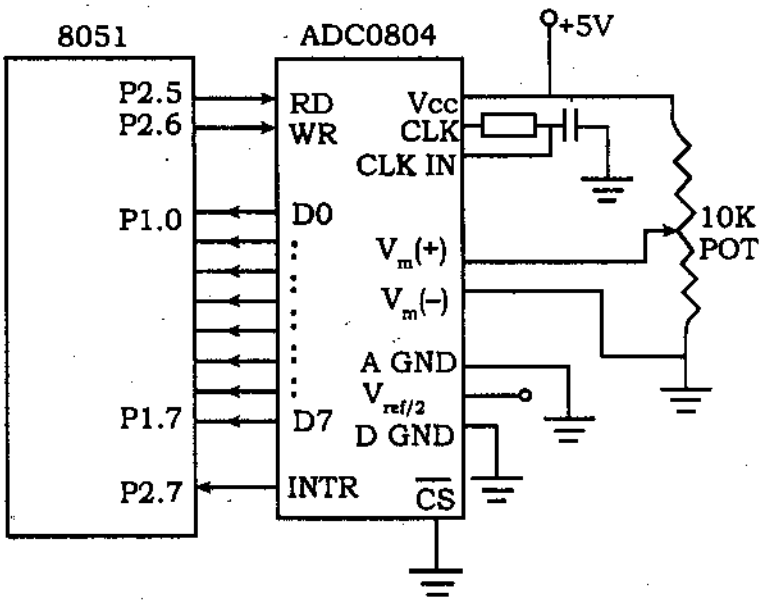
**৯। 8051 এর সাথে Full step sequence-এ স্টেপার মোটরের ইন্টারফেসিং প্রোগ্রাম উল্লেখ কর।**

**উত্তর :** নিচে Full step sequence-এ স্টেপারে মোটরের ইন্টারফেসিং প্রোগ্রাম উল্লেখ করা হলো।

```
            ORG 0H
            STEPPER EQUP1
MAIN :
            MOV STEPPER,#0CH      ; move the code to phase I into the port
            ACALL DELAY           ; Phase II code
            MOV STEPPER,#03H
            ACALL DELAY
            MOV STEPPER,#09H      : Phase IV code
            ACALL DELAY
            SJMP MAIN
DELAY :     MOV R7,#4
WAIT1 :     MOV R5,#0FFH
WAIT :      DJNZ R5, WAIT
            DJNZ R5, WAIT1
            DJNZ R7, WAIT2
            RET
            END
```

**১০। ADC এর সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন কর।**

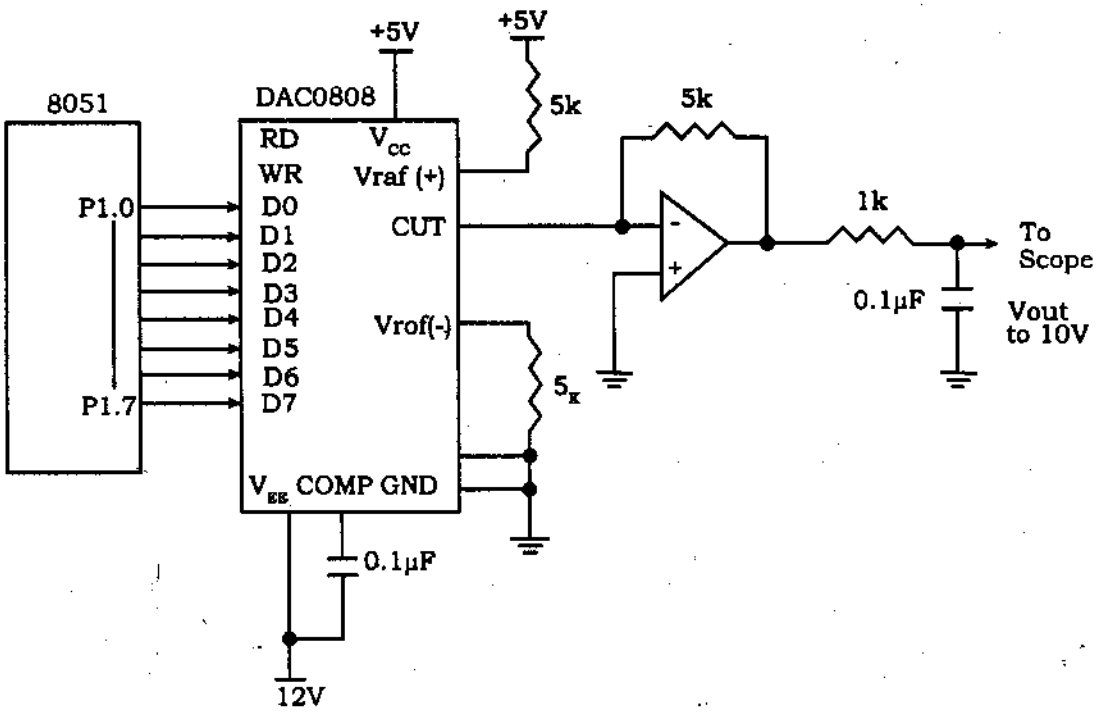
**উত্তর :** ADC এর সাথে 8051 এর ইন্টারফেসিং সার্কিট অঙ্কন করা হলো :




চিত্র : ADC 0804-এর সাথে 8051 এর ইন্টারফেসিং সার্কিট

**১১। DAC-এর সাথে 8051 এর ইন্টারফেসিং দেখাও।**

**উত্তর :** নিচে DAC-এর সাথে 8051 এর ইন্টারফেসিং দেখানো হলো :




চিত্র : 8051 এর সাথে DAC IC-এর ইন্টারফেসিং সার্কিট

**১২। P1 পোর্টে অ্যানালগ ইনপুট নিয়ে P0 পোর্টে ডিজিটাল আউটপুট পাওয়ার জন্য 8051-এর অ্যাসেম্বলি প্রোগ্রাম লিখ।**

**উত্তর :** P1 পোর্টে অ্যানালগ ইনপুট দিয়ে P0 পোর্টে ডিজিটাল আউটপুট পাওয়ার জন্য 8051-এর অ্যাসেম্বলি প্রোগ্রাম :

```
        ORG 00H
        MOV P1, #11111111B      ; initiates P1 as the input port
MAIN :
        SETB P2.5               ; makes RD high (P2.5)
        CLR P2.6                ; makes WR low P2.6
        SETB P2.6               ; low to high pulse to WR for starting conversion
WAIT :  JB P2.7, WAIT           ; polls until INTR = 0
        CLR P2.5                ; high to low pulse to RD for reading the data
        MOV A, P1                 from ADC
        CPL A                   ; moves the digital data to accumulator
        MOV P0, A               ; complements the digital data
        SJMP MAIN               ; outputs the data to P0
        END                     ; jumps back to the MAIN program
```

**১৩। P1 পোর্টে অ্যানালগ ইনপুট নিয়ে P0 পোর্টে ডিজিটাল আউটপুট পাওয়ার জন্য 8051 এর অ্যাসেম্বলি প্রোগ্রাম লিখ।**

**উত্তর :** P1 পোর্টে অ্যানালগ ইনপুট নিয়ে P0 পোর্টে ডিজিটাল আউটপুট পাওয়ার জন্য 8051 এর অ্যাসেম্বলি প্রোগ্রাম লিখা হলো :

```
        ORG 00H
        MOV P1, #11111111B      ; initiates P1 as the input port
MAIN :
        SETB P2.5               ; makes RD high (P2.5)
        CLR P2.6                ; makes WR low P2.6
        SETB P2.6               ; low to high pulse to WR for starting conversion
WAIT :  JB P2.7, WAIT           ; polls until INTR = 0
        CLR P2.5                ; high to low pulse to RD for reading the data from ADC
        MOV A, P1               ; moves the digital data to accumulator
        CPL A                   ; complements the digital data
        MOV P0, A               ; outputs the data to P0
        SJMP MAIN               ; jumps back to the MAIN program
        END
```

**১৪। স্টেপার মোটরের Full step sequence ও Half step sequence-এর step-গুলো লিখ।**

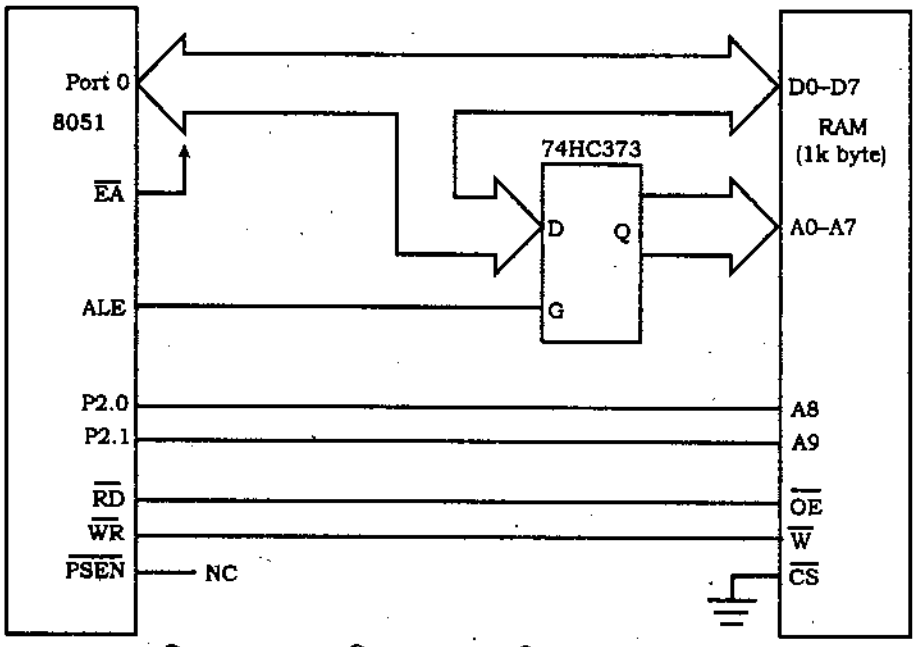
**উত্তর :** স্টেপার মোটরের Full step sequence ও Half step sequence-এর step-গুলো নিম্নরূপ :

| Full mode sequence (4 steps) | | | | |
|---|---|---|---|---|
| Step | A | B | $\bar{A}$ | $\bar{B}$ |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 2 | 0 | 0 | 1 | 1 |
| 3 | 1 | 0 | 0 | 1 |

| Half mode sequence (8 steps) | | | | |
|---|---|---|---|---|
| Step | A | B | $\overline{A}$ | $\overline{B}$ |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 2 | 0 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 1 | 1 |
| 5 | 0 | 0 | 0 | 1 |
| 6 | 1 | 0 | 0 | 1 |
| 7 | 1 | 0 | 0 | 0 |

**১৫। 8051 এর সাথে EPROM-এর ইন্টারফেসিং দেখাও।**

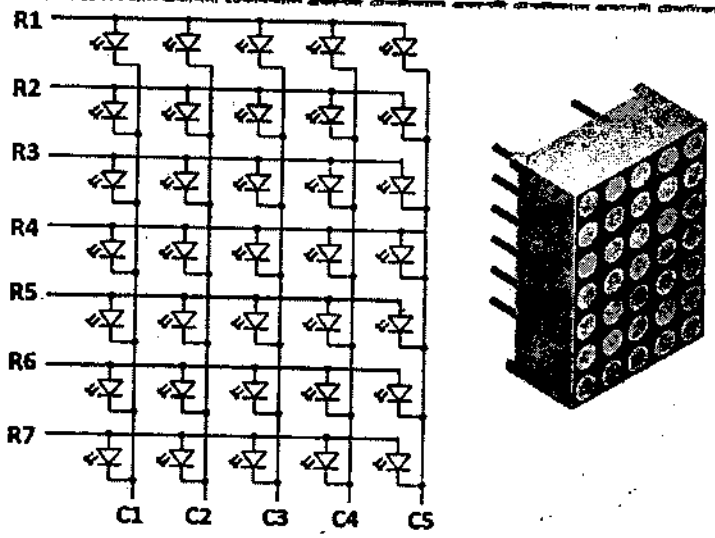
**উত্তর :** 8051 এর $\overline{PSEN}$ সিগন্যাল দ্বারা বাহ্যিক কোড মেমোরি (EPROM) কার্যকরী হয়। যখন একটি বাহ্যিক EPROM ব্যবহার হয়, তখন পোর্ট P0 ও P2 কে ইনপুট/আউটপুট পোর্ট হিসেবে পাওয়া যায় না। বাহ্যিক EPROM-এর হার্ডওয়্যার সংযোগ নিচের চিত্রে দেখানো হলো। 8051 এর এক মেশিন সাইকেল হলো 12 অসিলেটর সাইকেল। যদি অন চিপ অসিলেটর 12 মে. হার্টসের ক্রিস্টাল দ্বারা পরিচালিত হয়, তবে এক মেশিন সাইকেলের সময়কাল 1 মাইক্রোসেকেন্ড। এক মেশিন সাইকেলে ALE দুবার পালস প্রদান করে এবং প্রোগ্রাম মেমোরি থেকে দুই বাইট পড়ে। এই অপারেশনের টাইমিং ডায়াগ্রাম চিত্র ১৩.১৫ তে দেখানো হলো :




চিত্র : 8051-এ বাহ্যিক কোড মেমোরির (EPROM) ব্যবহার

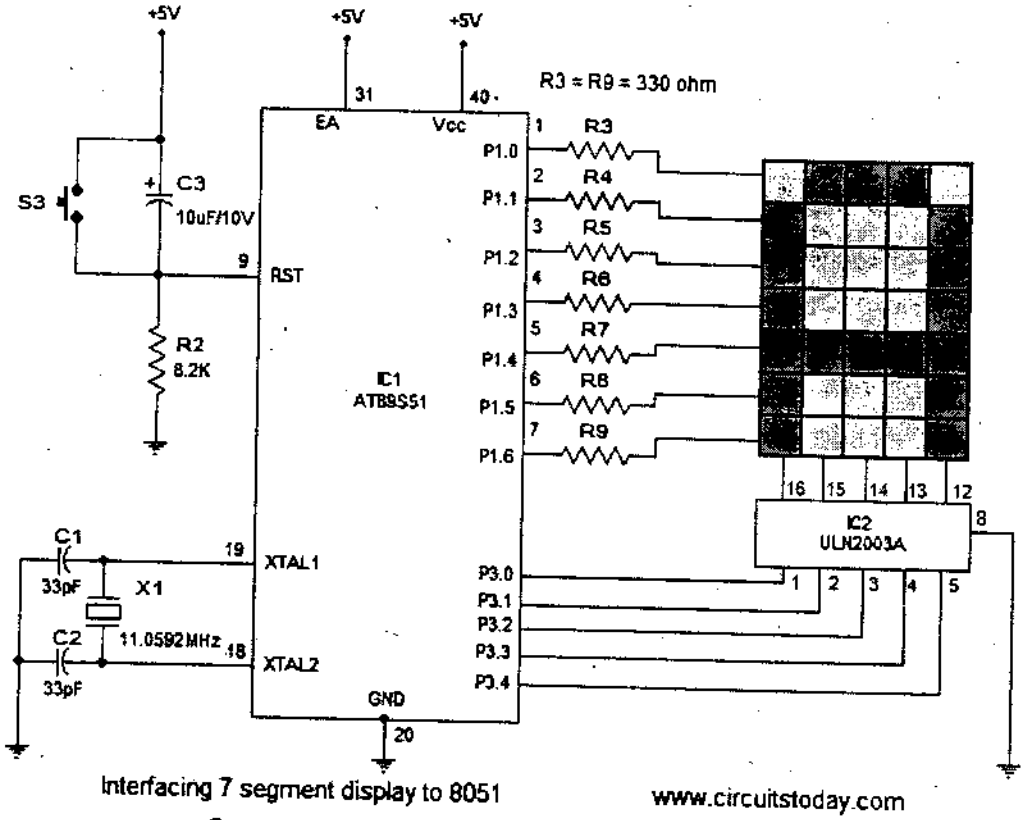
**১৬। 8051 এর সাথে Dot matrix display-এর ইন্টারফেসিং দেখাও।**

**উত্তর :** একটি আয়তাকার LED matrix নিয়ে LED dot matrix গঠিত। আকাঙ্ক্ষিত অক্ষর বা গ্রাফিক্স LED-এর সজ্জাকে সুইচিং করে প্রদর্শন করা হয়। সাধারণ ডিসপ্লে সজ্জা হলো 7 × 5, 8 × 8, 7 × 15 ইত্যাদি। নিচে একটি 7 × 5 ডট ম্যাট্রিক্স দেখানো হলো :




চিত্র : 7 × 5 Led dot matrix display

কোন স্বতন্ত্র LED বা LED এর একটি গ্রুপকে কার্যকরী করার জন্য প্রয়োজনীয় সংখ্যক সারি বা কলামকে সুইচিং করা হয়। যেমন– Row 1 কে High এবং Column 1 কে Low করলে উপরের LED (R1C1) জ্বলবে। এভাবে যে কোন অক্ষর বা গ্রাফিক্স ডিসপ্লে করা হয়। নিচের 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং দেখানো হলো :




Interfacing 7 segment display to 8051 www.circuitstoday.com

চিত্র : 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং

## রচনামূলক প্রশ্নোত্তর

১। **8051 এর সাথে 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লে এর ইন্টারফেসিং সার্কিটসহ প্রোগ্রাম লিখ।** *[ঢাকাশিবো : '১৫]*

উত্তর : প্রোগ্রাম-৭ ও চিত্র ১৩.১৮ নং দ্রষ্টব্য।

২। **ADC ও DAC-এর সাথে 8051 এর ইন্টারফেসিং বর্ণনা কর।** *[ঢাকাশিবো : '১৪]*

উত্তর : অনুচ্ছেদ ১৩.৩ ও ১৩.৪ নং দ্রষ্টব্য।

৩। **8051-এ বাহ্যিক ডাটা মেমোরি (RAM)-এর ব্যবহার ইন্টারফেসিং সার্কিট ও টাইমিং ডায়াগ্রামসহ বর্ণনা কর।** *[ঢাকাশিবো : '১৪R]*

উত্তর : অনুচ্ছেদ ১৩.৬ নং দ্রষ্টব্য।

৪। **ডিসি মোটরের সাথে 8051 এর ইন্টারফেসিং বর্ণনা কর।**

উত্তর : ১৩.১ নং অনুচ্ছেদ দ্রষ্টব্য।

৫। **একটি নির্দিষ্ট সময় পর পর ডিসি মোটরের স্বয়ংক্রিয়ভাবেতর দিক পরিবর্তনের প্রোগ্রাম লিখ। গতির দিক পরিবর্তনের নিয়ন্ত্রণ সিগন্যালের জন্য পোর্ট পিন P1.0 ও P1.1 ব্যবহারপূর্বক ইন্টারফেসিং সার্কিট দেখাও।**

উত্তর : অনুচ্ছেদ ১৩.১ নং এর প্রোগ্রাম- ১ নং দ্রষ্টব্য।

৬। **Push-button সুইচের মাধ্যমে ডিসি মোটরের গতির দিক পরিবর্তনের প্রোগ্রাম লিখ। গতির দিক পরিবর্তন নিয়ন্ত্রণ সিগন্যালের জন্য পোর্ট পিন P1.0 ও P1.1 এবং Push-Button সুইচের জন্য পোর্ট পিন P3.0 ও P3.1 ব্যবহারপূর্বক ইন্টারফেসিং সার্কিট দেখাও।**

উত্তর : অনুচ্ছেদ ১৩.১ নং এর প্রোগ্রাম- ২ নং দ্রষ্টব্য।

৭। **স্টেপার মোটরের সাথে 8051 এর ইন্টারফেসিং বর্ণনা কর।**

উত্তর : ১৩.২ নং অনুচ্ছেদ দ্রষ্টব্য।

৮। **8051 এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা কর।**

উত্তর : অনুচ্ছেদ ১৩.৫ নং দ্রষ্টব্য।

৯। **8051 এর সাথে ডট ম্যাট্রিক্স ডিসপ্লে এর ইন্টারফেসিং বর্ণনা কর।**

উত্তর : অনুচ্ছেদ ১৩.৭ নং দ্রষ্টব্য।

১০। **8051-এ বাহ্যিক কোড মেমোরি (EPROM)-এর ব্যবহার ইন্টারফেসিং সার্কিট ও টাইমিং ডায়াগ্রামসহ বর্ণনা কর।**

উত্তর : অনুচ্ছেদ ১৩.৬ নং দ্রষ্টব্য।

১১। **8051 এর সাথে স্টেপার মোটরের ইন্টারফেসিং বর্ণনা কর।**

উত্তর : অনুচ্ছেদ ১৩.২ নং দ্রষ্টব্য।

১২। **ডট মেট্রিক ডিসপ্লের সাথে 8051 এর ইন্টারফেসিং বর্ণনা কর।**

উত্তর : অনুচ্ছেদ ১৩.৮ নং দ্রষ্টব্য।

# ব্যবহারিক

# Practical

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

পরীক্ষণ নাম : **P1 ও P2 পোর্টে ডাটা পাঠানোর প্রোগ্রাম।**

**Experiment Name : Test a Program to Send a Data to Port P1 and p2**

```
BACK :   MOV     A, #55H     ; data#55h sends to accumulator
         MOV     P1, A       ; Data send to P1
         ACALL   DELAY
         MOV     A, #AAH     ; data#AAh sends to accumulator
         MOV     P2, A       ; data send to P2
         SJMP    BACK        ; Keep doing it
```

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

পরীক্ষণ নাম : **RAM-এর 30H স্থান থেকে 8 বাইটকে ডাটা 50H স্থানে কপি করার প্রোগ্রাম।**

**Experiment Name : Test a Program to Copy 8 Bytes of Data From RAM Location Staring at 30H to RAM Location Starting 50H**

```
         MOV R0, #35H      ; source pointer
         MOV R1, #60H      ; destination pointer
         MOV R3, #10       ; counter
BACK :   MOV A, @R0        ; get a byte from source
         MOV @R1, A        ; copy it to destination
         INC R0            ; increment source pointer
         INC R1            ; increment destination pointer
         DJNZ R3, BACK     ; keep doing for ten bytes
```

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

পরীক্ষণ নাম : **RAM-এর 30H স্থান থেকে 16 বাইট ডাটাকে 50H স্থানে স্থানান্তর করার প্রোগ্রাম।**

**Experiment Name : Test a Program to Transfer the 16 Bytes of Data from ROM Location Staring at 30H to RAM Location Starting 50H**

```
         MOV     A, #0FFH    ; A = FF hex
         MOV     P0, A       ; make P0 an i/p port by writing it all 1's
Back :   MOV     A, P0       ; get data from P0
         MOV     P1, A       ; send it port 1
         SJMP    BACK        ; keep doing it
```

| প্রোগ্রাম নং : ০৪ | তারিখ : ..... | সময় :..... |
|---|---|---|

পরীক্ষণ নাম : **RAM-এর 50H স্থান থেকে 55H পর্যন্ত যোগ করার প্রোগ্রাম।**

**Experiment Name : Test a Program to Find the Sum of the Values of RAM Locations from 50H to 55H. Register A Should Contain the Low Byte and R7 the High Byte**

```
MOV     A,  #25H      ; load 25H into A
MOV     R2, #34H      ; load 34H into R2
ADD     A, R2         ; add R2 to Accumulator
                      ; (A = A + R2)
```

| প্রোগ্রাম নং : ০৫ | তারিখ : ..... | সময় :..... |
|---|---|---|

পরীক্ষণ নাম : **P1 থেকে X-এর মান নিয়ে P2 তে $X^2$-এর মান নির্ণয়ের প্রোগ্রাম।**

**Experiment Name : Test a Program to Get the Vale of x and Send $x^2$ to p2 Continuously**

```
            ORG0                          ; LOAD TABLE ADDRESS
            MOV DPTR,#300H
            MOV A,#0FFH; A = FF
            MOV P1,A                      ; CONFIGURE P1 INPUT PORT
BACK :      MOV A,P1                      ; GET X
            MOV A,@A + DPTR               ; GET X SAURE FROM TABLE
            MOV P2,A                      ; ISSUE IT TO P2
            SJMP BACK                     ; KEEP DOINT IT
            ORG 300H
XSQR_TABLE :
            DB 0, 1, 4, 9, 16, 25, 36, 49, 64, 81
            END
```

| প্রোগ্রাম নং : ০৬ | তারিখ : ..... | সময় :..... |
|---|---|---|

পরীক্ষণ নাম : **ROM থেকে কোন তথ্য নিয়ে এটিকে RAM (upper memory space)-এর 80h থেকে কপি কর এবং পরে এটিকে P0 পোর্টে পাঠানোর প্রোগ্রাম।**

**Experiment Name : Test a Program to Get Hex Data in the Range of 00-FFH from the Port**

```
            ORG     0H
            MOV     DPTR,#MYDATA
            MOV     R1,#80H       ; access the upper memory
B1 :        CKR     A
            MOVC    A@+DPTR       ; copy from code ROM
            MOV     @R1, A        ; store in upper memory
            MOV     P0, A         ; give a copy to P0
```

```
        JZ      EXIT        ; exit if last byte
        INC     DPTR        ; increment DPTR
        INC     R1          ; increment R1
        SJMP    B1          ; repeat until last byte
EXIT    SJMP    $           ; stay here when finished
;-----------------------------------
        ORG     300H
MYDATA          DB      "Dhaka Mohila Polytechnic Institute", 0
        END
```

| | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : LED Flashing-এর প্রোগ্রাম লিখ।**

**Experiment Name : Develop and Test a Program for Flashing LEDs**

**সমাধান :**

```
        ORG 00H
        CLR P3.0
        CLR P2.0
AGAIN :JB P3.0, ON
        SETB P2.0
        ACALL DELAY
        CLR P2.0
        ACALL DELAY
        SJMP AGAIN
ON :    SETB P2.0
        SJMP AGAIN
DELAY : MOV R0,#05H
GO :    MOV R1,#0FFH
HERE :  DJNZ R1,HERE
        DJNZ R0,GO
        RET
        END
```

| | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : 0 হতে 9 পর্যন্ত 7-segment ডিসপ্লে এর জন্য প্রোগ্রাম।**

**Experiment Name : Develop and Test a Program for Displaying 0 to 9 on 7-Segment Display**

```
        ORG 000H            ; initial starting address
START : MOV A, #00001001B   ; initial value of accumulator
        MOV B, A
        MOV R0,#0ah         ; Register R0 initialized as counter which counts
                              from 10 to 0
LABEL : MOV A, B
        INC A
        MOV B, A
```

```
        MOVC A, @A + PC     ; adds the byte in A to the program counters
                              address
        MOV P1, A
        ACALL DELAY         ; calls the delay of the timer
        DEC R0              ; counter R0 decremented by 1
        MOV A, R0           ; R0 moved to accumulator to check if it is zero in
                              next instruction.
        JZ START            ; checks accumulator for zero and jumps to START.
                            ; done to check if counting has been finished.
        SJMP LABEL
        DB 3FH              ; digit drive pattern for 0
        DB 06H              ; digit drive pattern for 1
        DB 5BH              ; digit drive pattern for 2
        DB 4FH              ; digit drive pattern for 3
        DB 66H              ; digit drive pattern for 4
        DB 6DH              ; digit drive pattern for 5
        DB 7DH              ; digit drive pattern for 6
        DB 07H              ; digit drive pattern for 7
        DB 7FH              ; digit drive pattern for 8
        DB 6FH              ; digit drive pattern for 9
DELAY :  MOV R4,#05H        ; subroutine for delay
WAIT1 :  MOV R3,#00H
WAIT2 :  MOV R2,#00H
WAIT3:   DJNZ R2, WAIT3
         DJNZ R3, WAIT2
         DJNZ R4, WAIT1
         RET
         END
```

| প্রোগ্রাম নং ০১ | তারিখ : ..... | সময় :..... |
|---|---|---|

পরীক্ষণ নাম : স্কয়ার ওয়েভ উৎপাদনের জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।

**Experiment Name : Develop and Test a Program for Generating Square Wave**

```
        MOV   TMOD   ,#02h   ; 8 bit auto-reload mode
        MOV   TH0    ,#-50   ; – 50 reload value in TH0
        SETB  TR0            ; start timer
LOOP :  JNB   TF0,   LOOP    ; wait for overflow
        CLR   TF0            ; clear timer overflow flag
        CPL   P1.0           ; toggle port bit
        SJMP  Loop           ; repeat
        END
```

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

পরীক্ষণ নাম : কী-বোর্ড ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।

**Experiment Name : Develop and Test a Program for Interfacing Keyboard**

```
        ORG 00H
        MOV DPTR, #LUT          ; moves starting address of LUT to DPTR
        MOV A, #11111111B       ; loads A with all 1's
        MOV P0, #00000000B      ; initializes P0 as output port

BACK :  MOV P1, #11111111B      ; loads P1 with all 1's
        CLR P1.0                ; makes row 1 low
        JB P1.4, NEXT1          ; checks whether column 1 is low and
                                ; jumps to NEXT1 if now low
        MOV A, #0D              ; loads A with 0D if column is low
                                ; (that means key 1 is pressed)
        ACALL DISPLAY           ; calls DISPLAY subroutine
NEXT1 : JB P1.5, NEXT2          ; checks whether column 2 is low and so on.....
        MOV A, #1D
        ACALL DISPLAY
NEXT2 : JB P1.6, NEXT3
        MOV A, #2D
        ACALL DISPLAY
NEXT3 : JB P1.7, NEXT4
        MOV A, #3D
        ACALL DISPLAY
NEXT4 : SETB P1.0
        CLR P1.1
        JB P1.4, NEXT5
        MOV A, #4D
        ACALL DISPLAY
NEXT5 : JB P1.5, NEXT6
        MOV A, #5D
        ACAL DISPLAY
NEXT6 : JB P1.6, NEXT7
        MOV A, #6D
        ACALL DISPLAY
NEXT7 : JB P1.7, NEXT8
        MOV A, #7D
        ACALL DISPLAY
NEXT8 : SETB P1.1
        CLR P1.2
        JB P1.4, NEXT9
        MOV A, #8D
        ACALL DISPLAY
```

```
NEXT9 :   JB P1.5, NEXT10
          MOV A, #9D
          ACALL DISPLAY
NEXT10    JB P1.6, NEXT11
   :      MOV A, #10D
          ACALL DISPLAY
NEXT11    JB P1.7, NEXT12
   :      MOV A, #11D
          ACALL DISPLAY
NEXT12    SETB P1.2
   :      CLR P1.3
          JB P1.4, NEXT13
          MOV A, #12D
          ACALL DISPLAY
NEXT13    JB P1.5, NEXT14
   :      MOV A, #13D
          ACALL DISPLAY
NEXT14    JB P1.6, NEXT15
   :      MOV A, #14D
          ACALL DISPLAY
NEXT15    JB P1.7, BACK
   :      MOV A, #15D
          ACALL DISPLAY
          LJMP BACK
DISPLAY:  MOVC A, @A + DPTR   ; gets digit drive pattern for the current key from LUT
          MOV P0, A           ; puts corresponding digit drive pattern into P0
          RET
LUT :     DB 01100000B        ; Look up table starts here
          DB 11011010B
          DB 11110010B
          DB 11101110B
          DB 01100110B
          DB 10110110B
          DB 10111110B
          DB 00111110B
          DB 11100000B
          DB 11111110B
          DB 11110110B
          DB 10011100B
          DB 10011110B
          DB 11111100B
          DB 10001110B
          DB 01111010B
          END
```

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

**পরীক্ষণ নাম : LCD ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

**Experiment Name : Develop and Test a Program for Interfacing LCD**

```
MOV A,#38H        ; Use 2 lines and 5 × 7 matrix
ACALL CMND
MOV A,#0FH        ; LCD ON, curson ON, cursor blinking ON
ACALL CMND
MOV A,#01H        ; Clear screen
ACALL CMND
MOV A,#06H        ; Increment cursor
ACALL CMND
MOV A,#82H        ; Cursor line one, position 2
ACALL CMND
MOV A,#3CH        ; Activate second line
ACALL CMND
MOV A,#3CH        ; Activate second line
ACALL CMND
MOV A,#49D
ACALL DISP
MOV A,#54D
ACALL DISP
MOV A,#88D
ACALL DISP
MOV A,#50D
ACALL DISP
MOV A,#532D
ACALL DISP
MOV A,#76D
ACALL DISP
MOV A,#67D
ACALL DISP
MOV A,#68D
ACALL DISP
MOV A,#0C1H       ; Jump to second line, position 1
ACALL CMND
MOV A,#67D
ACALL DISP
MOV A,#73D
ACALL DISP
MOV A,#82D
ACALL DISP
MOV A,#67D
ACALL DISP
MOV A,#85D
ACALL DISP
MOV A,#73D
ACALL DISP
MOV A,#84D
```

```
ACALL DISP
MOV A,#83D
ACALL DISP
MOV A,#84D
ACALL DISP
MOV A,#79D
ACALL DISP
MOV A,#68D
ACALL DISP
MOV A,#65D
ACALL DISP
MOV A,#89D
ACALL DISP
HERE : SJMP HERE
CMND : MOV P1, A
CLR P3.5
CLR P3.4
SETB P3.3
CLR P3.3
ACALL DELAY
RET;
DISP:MOV P1, A
SETB P3.5
CLR P3.4
SETB P3.3
CLR P3.3
ACALL DELAY
RET;
DELAY : CLR P3.3
CLR P3.5
SETB P3.4
MOV P1, #0FFh
SETB P3.3
MOV A, P1
JB ACC.7, DELAY
CLR P3.3
CLR P3.4
RET;
END
```

| | | |
|---|---|---|
| [redacted] | তারিখ : ..... | সময় :..... |

**পরীক্ষণ নাম : DC মোটর ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

## Experiment Name : Develop and Test a Program for Interfacing DC Motor

```
        ORG 00H              ; initial starting address
MAIN :  MOV P1, #00000001B   ; motor runs clockwise
        ACALL DELAY          ; calls the 1S DELAY
        MOV P1, #00000010B   ; motor runs anticlockwise
        ACALL DELAY          ; calls the 1S DELAY
        SJMP MAIN            ; jumps to label MAIN for repeating the cycle
```

```
DELAY :     MOV R4, #0FH
WAIT1 :     MOV R3, #00H
WAIT2 :     MOV R2, #00H
WAIT3 :     DJNZ R2, WAIT3
        DJNZ R3, WAIT2
        DJNZ R4, WAIT1
        RET
        END
```

| প্রোগ্রাম নং : ১৩ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : স্টেপার মোটর ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

**Experiment Name : Develop and Test a Program for Interfacing Stepper Motor**

```
            ORG 0H
            STEPPER EQU P1
MAIN :
            MOV STEPPER, #0CH     ; move the code to phase I into the port
            ACALL DELAY
            MOV STEPPER, #06H     ; phase II code
            ACALL DELAY
            MOV STEPPER,#03H      ; phase II code
            ACALL DELAY
            MOV STEPPER,#09H      ; phase IV code
            ACALL DELAY
            SJMP MAIN
DELAY :     MOV R7, #4
WAIT2 :     MOV R6,#0FFH
WAIT1 :     MOV R5,#0FFH
WAIT :      DJNZ R5, WAIT
            DJNZ R6, WAIT1
            DJNZ R7, WAIT2
            RET
            END
```

| প্রোগ্রাম নং : ১৪ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : DAC ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

**Experiment Name : Develop and Test a Program for Interfacing DAC**

```
            MOV     A,#DATA; (A)    = # data
START :     MOV     90H, A          ; Port 1 = (A)
            INC     A
            LJMP    START           ; Repeat
```

| প্রোগ্রাম নং : ১৫ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : ADC ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম লিখ।**

**Experiment Name : Develop and Test a Program for Interfacing ADC**

```
        ORG 00H
        MOV P1, #11111111B      ; initiates P1 as the input port
MAIN :
        SETB P2.5               ; makes RD high (P2.5)
        CLR P2.6                ; makes WR low P2.6
        SETB P2.6               ; low to high pulse to WR for starting conversion
WAIT :  JB P2.7, WAIT           ; polls until INTR = 0
        CLR P2.5                ; high to low pulse to RD for reading the data from ADC
        MOV A, P1               ; moves the digital data to accumulator
        CPL A                   ; complements the digital data
        MOV P0, A               ; outputs the data to P0
        SJMP MAIN               ; jumps back to the MAIN program
        END
```

| প্রোগ্রাম নং : ১৬ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : টেম্পারেচার ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

**Experiment Name : Develop and Test a Program for Interfacing Temperature Sensor**

```
; Assembly code to read temperature, convert it and put it on P0 with some delay
        RD      BIT     P2.5        ; RD
        WR      BIT     P2.6        ; WR (Start conversion)
        INTR    BIT     P2.7        ; end-of-conversion (EOC)
        MYDATA  EQU     P1          ; P1.0-P1.7 = D0-D7 of the ADC0848
        MOV     P1.#0FFH            ; make P1 = input
        SETB    INTR
BACK :  CLR     WR                  ; WR = 0
        SETB    WR                  ; WR = 1, L-to-H to starting of
                                        conversion (SOC)
HERE :  JB      INTR, HERE          ; wait for end of conversion
        CLR     RD                  ; conversion finished enable RD
        MOV     A, MYDATA           ; read the data for ADC0848
        ACALL   CONVERSION          ; hexx-to-ASCII conversion
        ACALL   DATA_DISPLAY        ; display the data
        SETB    RD                  ; make RD = 1 for next round
        SJMP    BACK
CONVERSION :
        MOV     B, #10
        DIV     AB
        MOV     R7, B               ; least significant byte
        MOV     B, #10
        DIV     AB
        MOV     R6, B
        MOV     R5, A               ; most significant byte
        RET
```

```
DATA_DISPLAY :
MOV     P0, R7
ACALL   DELAY
MOV     P0, R6
ACALL   DELAY
MOV     P0, R5
ACALL   DELAY
RET
```

| প্রোগ্রাম নং ২১ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : Dot matrix display ইন্টারফেসিং এর জন্য একটি অ্যাসেম্বলি প্রোগ্রাম।**

## Experiment Name : Develop and Test a Program for Dot Matrix Display

সমাধান :

```
        ORG 00H
        MOV P3,#00000000B       ; initializes port 3 as output port
        MOV P1,#00000000B       ; initializes port 1 as output port
MAIN :  MOV P3,#01111110B
        MOV P1,#11111110B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1,#11111101B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1#11111Q11B
        ACALL DELAY
        MOV P3,#00010001B
        MOV P1,#11110111B
        ACALL DELAY
        MOV P3,#01111110B
        MOV P1,#11101111B
        ACALL DELAY
        SJMP MAIN               ; jumps back to the main loop
DELAY:  MOV R6,#255D            ; 1 ms delay subroutine
HERE :  DJNZ R6, HERE
        RET
        END
```

| প্রোগ্রাম নং ২২ | তারিখ : ..... | সময় :..... |
|---|---|---|

**পরীক্ষণ নাম : P1 পোর্টের P1.0 পিনের সংযুক্ত সুইচের অবস্থা (অন/অফ) P2.7 পিনের সাথে সংযুক্ত LED মাধ্যমে জানার প্রোগ্রাম।**

## Experiment Name : Test a Program to Copy 8 Bytes of Data from RAM Location Staring at 30H to RAM Location Starting 50H

```
        SETB P1.7       ; make P1.7 an input
AGAIN : MOV C,P1.0      ; read SW status into CF
        MOV P2.7, C     ; send SW status to LED
        SJMP AGAIN      ; keep repeating
```

টেকনিক্যাল প্রকাশনীর বই পড়ুন
রেফার্ডমুক্ত জীবন গড়ুন।

# সাজেশন

১। অতি সংক্ষিপ্ত প্রশ্নোত্তর
২। সংক্ষিপ্ত প্রশ্নোত্তর
৩। রচনামূলক প্রশ্নোত্তর

অধিক কমনের নিশ্চয়তা
দিচ্ছে টেকনিক্যাল প্রকাশনী

ফ্লেক্সিলোড অথবা বিকাশযোগে মূল্য পরিশোধে টেকনিক্যাল প্রকাশনী
থেকে আজই আপনার কপিটি সংগ্রহ করুন।

# [illegible]

## অতি সংক্ষিপ্ত প্রশ্নোত্তর

★★★ ১। **মাইক্রোকন্ট্রোলার কাকে বলে?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২। **ইমবেডেড মাইক্রোকন্ট্রোলার কাকে বলে?**
**অথবা, ইমবেডেড মাইক্রোকন্ট্রোলার কী?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৩। **ইমবেডেড সিস্টেম বলতে কী বুঝায়?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৪। **MCS-51 পরিবার কী?**
**অথবা, 8051 পরিবার কাকে বলে?**
**উত্তর :** অনুশীলনী ২ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৫। **Long range কী?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৬। **সাবরুটিন কাকে বলে?**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৭। **Embedded controller কাকে বলে?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৮। **Microcontroller শব্দের অর্থ কী?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৯। **কয়েকটি জনপ্রিয় মাইক্রোকন্ট্রোলারের নাম লিখ।**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১০। **মাইক্রোকন্ট্রোলারের গতি (Speed) বলতে কী বুঝ?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১১। **মাইক্রোকন্ট্রোলারের প্যাকেজিং (Packaging) বলতে কী বুঝ?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১২। **মাইক্রোকন্ট্রোলারের মেমোরি বলতে কী বুঝ?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৯ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৩। **Microcontroller এর ব্যবহার লেখ।**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১০ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৪। **PIC কী?**
**উত্তর :** অনুশীলনী ২ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৫। **MCS-51 পরিবারের ৫টি মাইকোকন্ট্রোলারের নাম লিখ।**
**উত্তর :** অনুশীলনী ২ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৬। **Sepcial function registers (SFRs) কী?**
**উত্তর :** অনুশীলনী ৩ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৭। **DJNZ R5 ইন্সট্রাকশন কী?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৮। **নির্দেশনা সেট বলতে কী বুঝ?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৯। **RLA নির্দেশনা দ্বারা কী বুঝায়?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২০। **Absolute value কী?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২১। **কাজের বৈশিষ্ট্যের উপর ভিত্তি করে 8051 মাইক্রোকন্ট্রোলের নির্দেশনাকে কয় ভাগে ভাগ করা হয় ও কী কী?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২২। **IDE কী?**
**উত্তর :** অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২৩। **অ্যাসেম্বলি প্রোগ্রামে list file এর কাজ কী?**
**উত্তর :** অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২৪। **ACALL এর কাজ লেখ।**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৫। **STMP এর কাজ কী?**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৬। **8051-এ স্ট্যাক পয়েন্টার রেজিস্টার কীভাবে ব্যবহৃত হয়?**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৭। **8051 মাইক্রোকন্ট্রোলারে কী কী CALL নির্দেশনা আছে?**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৮। **DB ডাইরেক্টিভের ব্যবহার বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৯ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৯। **ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার লেখ।**
**উত্তর :** অনুশীলনী ৯ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩০। **SBUF রেজিস্টারের কাজ কী?**
**উত্তর :** অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩১। **Baud rate কীসের উপর নির্ভর করে?**
**উত্তর :** অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩২। **সিরিয়াল কমিউনিকেশন বলতে কী বুঝায়?**
**উত্তর :** অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৩। **8051 মাইক্রোকন্ট্রোলারের বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের কাজ কী?**
**উত্তর :** অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৪। **TI ও RI কখন সেট হয়?**
**উত্তর :** অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৫। **Busy Flag কী?**
**উত্তর :** অনুশীলনী ১২ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৬। **ইনস্ট্রাকশন বা নির্দেশনা বলতে কী বুঝ?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।।

৩৭। **আকারের উপর ভিত্তি করে নির্দেশনাকে কয় ভাগে ভাগ করা যায় ও কী কী?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৮। **অ্যাসেম্বলি ভাষায় মাইক্রোকন্ট্রোলার প্রোগ্রাম রচনায় অসুবিধা কী?**
উত্তর : অনুশীলনী ৭ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৯। **মাইক্রোকন্ট্রোলার প্রোগ্রামিং-এ সি ভাষার অসুবিধা কী?**
উত্তর : অনুশীলনী ৭ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪০। **Pull-up রেজিস্টরের মান নির্ণয়ের নিয়ম কী?**
উত্তর : অনুশীলনী ৮ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪১। **8051 এ কোন পোর্টে অভ্যন্তরীণ Pull-up রেজিস্টার নেই।**
উত্তর : অনুশীলনী ৮ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪২। **Interrupt service routine (ISR) বা Interrupt handler বলতে কী বুঝায়?**
উত্তর : অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৩। **16 × 2 LCD দ্বারা কী বুঝায়?**
উত্তর : অনুশীলনী ১২ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৪। **একটি বাণিজ্যিক LCD-এর নাম লিখ।**
উত্তর : অনুশীলনী ১২ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৫। **একটি ছোট ডিসি মোটরের starting কারেন্ট কত?**
উত্তর : অনুশীলনী ১৩ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৬। **একটি H-Bridge driver আইসি এর নাম লিখ।**
উত্তর : অনুশীলনী ১৩ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৭। **কীসের সাহায্যে ডিসি মোটরকে 8051 এর সাথে ইন্টারফেস করা হয়?**
উত্তর : অনুশীলনী ১৩ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

## সংক্ষিপ্ত প্রশ্নোত্তর

★★★ ১। **C কে Mid level language বলা হয় কেন?**
উত্তর : অনুশীলনী ৭ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ২। **C কে Structured language বলা হয় কেন?**
উত্তর : অনুশীলনী ৭ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৩। **C কে পোর্টেবল ল্যাংগুয়েজ বলা হয় কেন?**
উত্তর : অনুশীলনী ৭ এর সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৪। **সাবরুটিনের সুবিধাগুলো লিখ।**
উত্তর : অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৫। **8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা কর।**
উত্তর : অনুশীলনী ১৩ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৬। **মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসরের পার্থক্য লিখ।**
উত্তর : অনুশীলনী ১ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৭। **কয়েকটি High level language এর নাম লেখ।**
**অথবা, কয়েকটি OOP Language এর নাম লেখ।**
উত্তর : অনুশীলনী ৭ এর সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৮। **টাইমারের কাজ উল্লেখ কর।**
উত্তর : অনুশীলনী ৯ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ৯। **C-প্রোগ্রাম কখন, কোথায় এবং কে তৈরি করেন?**
**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১০। **একটি ইন্টারাপ্ট সম্পাদনের ধাপগুলো লিখ।**
**উত্তর :** অনুশীলনী ১১ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১১। **LCD, HD44780-এর পিন ডায়াগ্রাম অঙ্কন কর।**
**অথবা, LCD-এর পিন ডায়াগ্রাম অঙ্কন কর।**
**উত্তর :** অনুশীলনী ১২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১২। **বিভিন্ন প্রকার টেম্পারেচার সেন্সরের নাম লিখ।**
**উত্তর :** অনুশীলনী ১৩ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৩। **মাইক্রোকন্ট্রোলারের শ্রেণিবিভাগ উল্লেখ কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৪। **হার্ভার্ড আর্কিটেকচারের সুবিধা/অসুবিধা লিখ।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৫। **একটি মাইক্রোকন্ট্রোলার নির্বাচনের শর্তসমূহ আলোচনা কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৬। **মাইক্রোকন্ট্রোলারের মৌলিক ব্লক চিত্র অঙ্কন কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৭। **মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৮। **মাইক্রোপ্রসেসরের ব্লক ডায়াগ্রাম অঙ্কন কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

★★★ ১৯। **মাইক্রোকন্ট্রোলারের ব্যবহার বা সুবিধাসমূহ লিখ।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

২০। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৯ নং প্রশ্নোত্তর দ্রষ্টব্য।

২১। **8051-এর মূল বৈশিষ্ট্য লিখ।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ১০ নং প্রশ্নোত্তর দ্রষ্টব্য।

২২। **মাইক্রোকন্ট্রোলার বলতে কী বুঝায়?**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ১১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৩। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লিখ।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৪। **PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লিখ।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৫। **MCS-51 মাইক্রোকন্ট্রোলারের নামারিং পদ্ধতি উল্লেখ কর।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৬। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো কী কী?**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৭। **8051-এর মূল বৈশিষ্ট্য লিখ।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৮। **8051 মাইক্রোকন্ট্রোলারের মেমোরির গঠন চিত্র অঙ্কন কর ও বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৯। **8051 এর কন্ট্রোল SFR-গুলোর নাম লিখ।**
**অথবা, অ্যাড্রেসহ স্পেশাল ফাংশন রেজিস্টারগুলোর নাম লেখ।**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩০। **8051 মাইক্রোকন্ট্রোলারের রিসেট সার্কিট বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩১। **PSW রেজিস্টারের বিটগুলো বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩২। **8051 ইনস্ট্রাকশনের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৪ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৩। **CPU Instruction কাকে বলে?**
**উত্তর :** অনুশীলনী ৫ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৪। **8051 এর জন্য সি ভাষার bitwise operator গুলোর নাম লিখ।**
**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৫। **C কে Case sensitive language বলা হয় কেন?**
**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৬। **C-প্রোগ্রাম কম্পাইলেশন ও নির্বাহ প্রক্রিয়ার ধাপগুলো লিখ।**
**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৭। **একটি 16 × 2 LCD (HD44780)-এর পিনগুলোর কাজ লিখ।**
**অথবা, একটি 14 পিন LCD এর পিনগুলোর কাজ লেখ।**
**উত্তর :** অনুশীলনী ১২ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৮। **8051 এর সাথে RAM-এর ইন্টারফেসিং দেখাও।**
**উত্তর :** অনুশীলনী ১৩ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩৯। **8051 মাইক্রোকন্ট্রোলারের বিভিন্ন Family-এর মধ্যে পার্থক্য দেখাও।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪০। **8051 এ টাইমার/কাউন্টারের কাজ কী?**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪১। **8051 মাইক্রোকন্ট্রোলার কী কী অ্যাড্রেসিং মুড সমর্থন করে নাম লিখ।**
**উত্তর :** অনুশীলনী ৪ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪২। **অ্যাসেম্বলি ভাষা প্রোগ্রামের ফাইলগুলোর নাম লিখ।**
**উত্তর :** অনুশীলনী ৫ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৩। **12 মে. হার্টস ক্লক পাল চালিত 8051 মাইক্রোকন্ট্রোলারে একটি ক্ষুদ্রতম নির্দেশনা সম্পন্ন হতে কত সময় লাগে?**
**উত্তর :** অনুশীলনী ৬ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৪। **Pull-up রেজিস্টরের উদ্দেশ্য লিখ।**
**উত্তর :** অনুশীলনী ৮ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪৫। **SCON রেজিস্টারের বিটগুলোর কাজ লিখ।**
**উত্তর :** অনুশীলনী ১০ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

## রচনামূলক প্রশ্নোত্তর

*** ১। **8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন ব্লকের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ২। **মাইক্রোকন্ট্রোলার নির্বাচনে বিবেচ্য বিষয়গুলো উল্লেখ কর।**
**উত্তর :** অনুশীলনী ১ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৩। **একটি সাধারণ মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রামসহ এর বিভিন্ন ব্লকের বর্ণনা দাও।**
**অথবা, বেসিক মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন অংশের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ১ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৪। **8051 মাইক্রোকন্ট্রোলারের পিন ডায়াগ্রাম অঙ্কন করে পিনগুলোর কাজের সংক্ষিপ্ত বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৫। **8051 মাইক্রোকন্ট্রোলারের অ্যাড্রেসিং মুডগুলো উদাহরণসহসংক্ষেপে বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৪ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৬। **উদাহরণসহকারে 8051 মাইক্রোকন্ট্রোলারে বিদ্যুৎশক্তি প্রয়োগের পর ধাপে ধাপে এর কাজের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৫ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৭। **একটি সম্পাদনযোগ্য অ্যাসেম্বলি ভাষার প্রোগ্রাম রচনার ধাপগুলো বর্ণনা কর।**
**অথবা, অ্যাসেম্বলি ভাষার প্রোগ্রাম তৈরি ও সম্পাদনের ধাপ (Steps) লেখ।**
**উত্তর :** অনুশীলনী ৫ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

*** ৮। **8051 এর টাইমার অপারেশনের মুডগুলো বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৯ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৯। **সিরিয়াল ডাটা প্রেরণ ও গ্রহণের জন্য 8051 প্রোগ্রাম লিখ।**
**উত্তর :** অনুশীলনী ১০ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১০। **একটি মেট্রিক্স কী-বোর্ডের গঠনপ্রণালী বর্ণনা কর।**
**উত্তর :** অনুশীলনী ১২ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১১। **8051 এর সাথে 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লে এর ইন্টারফেসিং সার্কিটসহ প্রোগ্রাম লিখ।**
**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১২। **ADC ও DAC-এর সাথে 8051 এর ইন্টারফেসিং বর্ণনা কর।**
**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৩। **8051-এ বাহ্যিক ডাটা মেমোরি (RAM)-এর ব্যবহার ইন্টারফেসিং সার্কিট ও টাইমিং ডায়াগ্রামসহ বর্ণনা কর।**
**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৪। **উদাহরণসহ সাবরুটিন আহ্বান ও প্রত্যাবর্তন প্রক্রিয়া সংক্ষেপে বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৬ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৫। **8051 এর জন্য সি ভাষায় টাইম ডিলে তৈরির প্রক্রিয়া উদাহরণসহ বর্ণনা কর।**
**উত্তর :** অনুশীলনী ৭ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৬। **মাইক্রোকন্ট্রোলারে Pull-up রেজিস্টরের উদ্দেশ্য আলোচনা কর।**
**উত্তর :** অনুশীলনী ৮ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৭। **8051 এর বাহ্যিক হার্ডওয়্যার টাইমার ও সিরিয়াল কমিউনিকেশন ইন্টারাপ্ট বর্ণনা কর।**
**উত্তর :** অনুশীলনী ১১ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

# বিগত বছরের বোর্ড পরীক্ষার প্রশ্ন ও উত্তর

# বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৭ম পর্ব পরিপূরক পরীক্ষা-২০১৪

টেকনোলজি : ইলেকট্রনিক্স, ইলেকট্রোমেডিক্যাল ও মেকাট্রনিক (২০১০ প্রবিধান)

বিষয় : মাইক্রোকন্ট্রোলার অ্যান্ড ইমবেডেড সিস্টেম (বিষয় কোড : ৬৮৭১)

সময় : ৩ ঘন্টা পূর্ণমান : ৮০

ক ও খ-বিভাগের সকল প্রশ্নের এবং গ-বিভাগের যে কোনো ৫ (পাঁচ) টি প্রশ্নের উত্তর দাও।

## ক-বিভাগ (মান : ২ × ১০ = ২০)

১। **Embeded controller কাকে বলে?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২। **কয়েকটি জনপ্রিয় মাইক্রোকন্ট্রোলারের নাম লেখ।**
উত্তর : অনুশীলনী ২ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩। **PIC কী?**
উত্তর : অনুশীলনী ২ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪। **Absolute value কী?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

৫। **RLA নির্দেশনা দিয়ে কী বোঝায়?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

৬। **Long range কী?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

৭। **8051 মাইক্রোকন্ট্রোলারে কী কী CALL নির্দেশনা আছে?**
উত্তর : অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

৮। **Band rate কীসের ওপর নির্ভর করে?**
উত্তর : অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৯। **Special function register কী?**
উত্তর : অনুশীলনী ৩ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১০। **অ্যাসেম্বলি প্রোগ্রাম List file-এর কাজ কী?**
উত্তর : অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

## খ-বিভাগ (মান : ৩ × ১০ = ৩০)

১১। **মাইক্রোকন্ট্রোলার ও মইক্রোপ্রসেসরের মাঝে পার্থক্য লেখ।**
উত্তর : অনুশীলনী ১ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১২। **PIC মাইক্রোকন্ট্রোলারের বৈশিষ্ট্য লেখ।**
উত্তর : অনুশীলনী ২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৩। **8051 মাইক্রোকন্ট্রোলারের নাম্বারিং পদ্ধতি উল্লেখ কর।**
উত্তর : অনুশীলনী ২ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৪। **PSW রেজিস্টারের বিটগুলোর নাম লেখ।**
উত্তর : অনুশীলনী ৩ এর সংক্ষিপ্ত [illegible] নং প্রশ্নোত্তর দ্রষ্টব্য।

১৫। **সাবরুটিনের সুবিধাগুলো লেখ।**
উত্তর : অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৬। একটি ইন্টারাপ্ট সম্পাদনের ধাপগুলো লেখ।**

**উত্তর :** অনুশীলনী ১১ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৭। বিভিন্ন প্রকার টেম্পারেচার Sensor-এর নাম লেখ।**

**উত্তর :** অনুশীলনী ১৩ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৮। 8051-এর সাথে RAM-এর ইন্টারফেসিং দেখাও।**

**উত্তর :** অনুশীলনী ১৩ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৯। হার্ভার্ড আর্কিটেকচারের সুবিধা-অসুবিধা লেখ।**

**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২০। টাইমারের কাজগুলো লেখ।**

**উত্তর :** অনুশীলনী ৯ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

গ–বিভাগ (মান : ৬ × ৫ = ৩০)

**২১। একটি সাধারণ মাইকক্রোকন্ট্রোলারের ব্লক চিত্র অঙ্কন করে বিভিন্ন অংশের বর্ণনা দাও।**

**উত্তর :** অনুশীলনী ১ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২২। একটি সম্পাদনযোগ্য অ্যাসেম্বলি ভাষার প্রোগ্রাম রচনার ধাপগুলো বর্ণনা কর।**

**উত্তর :** অনুশীলনী ৫ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৩। 8051-এর ব্যাহ্যিক ডাটা মেমোরি ইন্টারফেসিং সার্কিট ও টাইমিং ডায়াগ্রাম বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৪। Port $P_1$ ও Port $P_2$-এ পর্যায়ক্রমিক ডাটা পাঠানোর মাধ্যমে ফ্লাশিং লাইন তৈরির একটি প্রোগ্রাম লেখ।**

**উত্তর :** ব্যবহারিক – ১ নং দ্রষ্টব্য।

**২৫। 8051 মাইক্রোকন্ট্রোলারের পিন ডায়াগ্রাম অঙ্কন করে কন্ট্রোল পিনগুলোর কাজ লেখ।**

**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৬। একটি মেট্রিক্স 'কী'-বোর্ডের গঠন প্রণালি বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১২ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

# বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

**ডিপ্লোমা ইন ইঞ্জিনিয়ারিং**

**৭ম পর্ব সমাপনী পরীক্ষা-২০১৪**

**টেকনোলজি : ইলেকট্রনিক্স, ইলেকট্রোমেডিক্যাল ও মেকাট্রনিক (২০১০ প্রবিধান)**

**বিষয় : মাইক্রোকন্ট্রোলার অ্যান্ড ইমবেডেড সিস্টেম (বিষয় কোড : ৬৮৭১)**

**সময় : ৩ ঘণ্টা** **পূর্ণমান : ৮০**

**ক ও খ-বিভাগের সকল প্রশ্নের এবং গ-বিভাগের যে কোন ৫ (পাঁচ)টি প্রশ্নের উত্তর দাও।**

**ক-বিভাগ (মান : ২ × ১০ = ২০)**

১। **মাইক্রোকন্ট্রোলার কাকে বলে?**
**উত্তর :** অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২। **8051 পরিবার কাকে বলে?**
**উত্তর :** অনুশীলনী ২ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩। **ইনস্ট্রাকশন সেট বলতে কী বোঝায়?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪। **IDE বলতে কী বোঝায়?**
**উত্তর :** অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৫। **সাবরুটিন কাকে বলে?**
**উত্তর :** অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৬। **Long range বলতে কী বোঝায়?**
**উত্তর :** অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

৭। **টাইমারের কাজ কী?**
**উত্তর :** অনুশীলনী ৯ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৮। **ইভেন্ট কাউন্টার হিসেবে টাইমারের ব্যবহার লেখ।**
**উত্তর :** অনুশীলনী ৯ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৯। **TI ও RI কখন সেট হয়?**
**উত্তর :** অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১০। **Busy flag কী?**
**উত্তর :** অনুশীলনী ১২ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**খ-বিভাগ (মান : ৩ × ১০ = ৩০)**

১১। **মাইক্রোকন্ট্রোলার ও মাইক্রোপ্রসেসর-এর মাঝে তুলনা কর।**
**উত্তর :** অনুশীলনী ১ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১২। **ইন্টেল 8051 মাইক্রোকন্ট্রোলারের বৈশিষ্ট্যগুলো লেখ।**
**উত্তর :** অনুশীলনী ২ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৩। **8051 রিসেট সার্কিটের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৩ এর সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৪। **8051 ইনস্ট্রাকশনের বর্ণনা দাও।**
**উত্তর :** অনুশীলনী ৪ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৫। **CUP ইনস্ট্রাকশন বলতে কী বোঝায়?**
**উত্তর :** অনুশীলনী ৫ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৬। সাবরুটিনের সুবিধাগুলো লেখ।**

**উত্তর :** অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৭। টাইমারের কাজ লেখ।**

**উত্তর :** অনুশীলনী ৯ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৮। একটি ইন্টারাপ্ট সম্পাদনের ধাপগুলো লেখ।**

**উত্তর :** অনুশীলনী ১১ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**১৯। LCD-এর পিন ডায়াগ্রাম অঙ্কন কর।**

**উত্তর :** অনুশীলনী ১২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২০। 8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১৩ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

গ-বিভাগ (মান : ৫ × ৬ = ৩০)

**২১। বেসিক মাইক্রোকন্ট্রোলারের ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন অংশের বর্ণনা দাও।**

**উত্তর :** অনুশীলনী ১ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২২। অ্যাসেম্বলি ভাষায় প্রোগ্রাম তৈরি ও সম্পাদনের ধাপগুলো (Steps) লেখ।**

**উত্তর :** অনুশীলনী ৫ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৩। ইন্টেল 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন অংশের বর্ণনা দাও।**

**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৪। ADC ও DAC-এর সাথে 8051-এর ইন্টারফেসিং বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৫। 8051-এর সিরিয়াল পোর্টের মাধ্যমে ডাটা প্রেরণ ও গ্রহণের প্রোগ্রাম লেখ।**

**উত্তর :** অনুশীলনী ১০ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

**২৬। একটি ম্যাট্রিক্স কী-বোর্ডের গঠনপ্রণালি বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১২ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

# বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা ইন ইঞ্জিনিয়ারিং

৭ম পর্ব সমাপনী পরীক্ষা-২০১৫

টেকনোলজি : ইলেকট্রনিক্স, ইলেকট্রোমেডিক্যাল ও মেকাট্রনিক (২০১০ প্রবিধান)

বিষয় : মাইক্রোকন্ট্রোলার অ্যান্ড ইমবেডেড সিস্টেম (বিষয় কোড : ৬৮৭১)

সময় : ৩ ঘণ্টা পূর্ণমান : ৮০

ক ও খ-বিভাগের সকল প্রশ্নের এবং গ-বিভাগের যে কোন ৫ (পাঁচ)টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান : ২ × ১০ = ২০)

১। **8051 পরিবার বলতে কী বোঝায়?**
উত্তর : অনুশীলনী ২ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২। **8051-এ স্ট্যাক পয়েন্টার রেজিস্টার কীভাবে ব্যবহৃত হয়?**
উত্তর : অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩। **AGAIN : DJNZ R5, AGAIN ইনস্ট্রাকশন কী?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪। **ACALL ও SJMP-এর কাজ কী?**
উত্তর : অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ২ ও ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

৫। **মাইক্রোকন্ট্রোলার কী?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৬। **সাবরুটিন কী?**
উত্তর : অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৭। **DB ডাইরেক্টিভের ব্যবহার বর্ণনা কর।**
উত্তর : অনুশীলনী ৯ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৮। **Embeded system কাকে বলে?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

৯। **SBUF রেজিস্টারের কাজ কী?**
উত্তর : অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১০। **8051 মাইক্রোকন্ট্রোলারের বাহ্যিক হার্ডওয়্যার ইন্টারাপ্টের কাজ কী?**
উত্তর : অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

খ-বিভাগ (মান : ৩ × ১০ = ৩০)

১১। **মাইক্রোকন্ট্রোলারের শ্রেণিবিন্যাস কর।**
উত্তর : অনুশীলনী ১ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১২। **8051 মাইক্রোকন্ট্রোলারের মেমোরি অর্গানাইজেশন বর্ণনা কর।**
উত্তর : অনুশীলনী ৩ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৩। **সাবরুটিনের সুবিধাগুলো লেখ।**
উত্তর : অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৪। **8051 এর জন্য 'সি' ভাষার Bitwise operatorগুলো সংক্ষেপে বর্ণনা কর।**
উত্তর : অনুশীলনী ৭ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৫। **LCD-এর পিন ডায়াগ্রাম অঙ্কন কর।**
উত্তর : অনুশীলনী ১২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৬। বিভিন্ন প্রকার টেম্পারেচার সেন্সর-এর নাম লেখ।

উত্তর : অনুশীলনী ১৩ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৭। একটি 14 পিন LCD -এর পিনগুলোর কাজ লেখ।

উত্তর : অনুশীলনী ১২ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৮। 8051-এর সাথে টেম্পারেচার সেন্সরের ইন্টারফেসিং বর্ণনা কর।

উত্তর : অনুশীলনী ১৩ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৯। অ্যাড্রেসসহ স্পেশাল ফাংশন রেজিস্টারগুলোর নাম লেখ।

উত্তর : অনুশীলনী ৩ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২০। 8051 মাইক্রোকন্ট্রোলারের P1 থেকে x-এর মান নিয়ে P2-তে $x^2$-এর মান পাঠানোর জন্য একটি অ্যাসেম্বলি ভাষায় প্রোগ্রাম লেখ।

উত্তর : ব্যবহারিক ৫ নং দ্রষ্টব্য।

গ-বিভাগ (মান : ৫ × ৬ = ৩০)

২১। 8051 মাইক্রোকন্ট্রোলারের সরল ব্লক ডায়াগ্রাম অঙ্কন করে বিভিন্ন ব্লকের বর্ণনা দাও।

উত্তর : অনুশীলনী ৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২২। 8051 মাইক্রোকন্ট্রোলারের অ্যাড্রেসিং মোডগুলোর উদাহরণসহ বর্ণনা দাও।

উত্তর : অনুশীলনী ৪ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৩। 8051 মাইক্রোকন্ট্রোলারের বিদ্যুৎ শক্তি প্রয়োগের পর ধাপে ধাপে কাজ উদাহরণসহ ব্যাখ্যা কর।

উত্তর : অনুশীলনী ৫ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৪। 8051 মাইক্রোকন্ট্রোলারে সাথে 7 × 5 ডট ম্যাট্রিক্স ডিসপ্লে-এর ইন্টারফেসিংসহ অ্যাসেম্বলি অথবা 'সি' ভাষায় 'A' অক্ষর প্রদর্শনেরজন্য একটি প্রগ্রাম লেখ।

উত্তর : অনুশীলনী ১৩ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৫। 8081 এর টাইমার অপারেশনের মোডগুলো বর্ণনা কর।

উত্তর : অনুশীলনী ৯ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৬। 8051 মাইক্রোকন্ট্রোলারের সাহায্যে 7-সেগমেন্ট ডিসপ্লেতে 0 – 9 পর্যন্ত সংখ্যা প্রদর্শন করার জন্য ইন্টারফেসিংসহ অ্যাসেম্বলি ভাষায় একটি প্রোগ্রাম লেখ।

উত্তর : ব্যবহারিক ৮ নং দ্রষ্টব্য।

# বাংলাদেশ কারিগরি শিক্ষা বোর্ড, ঢাকা

ডিপ্লোমা-ইন-ইঞ্জিনিয়ারিং

৭ম পর্ব সমাপনী পরীক্ষা-২০১৬

টেকনোলজি : ইলেকট্রনিক্স, ইলেকট্রো-মেডিক্যাল ও মেকাট্রনিক (২০১০ প্রবিধান)

বিষয় : মাইক্রোকন্ট্রোলার অ্যান্ড ইমবেডেড সিস্টেম (বিষয় কোড : ৬৮৭১)

সময় : ৩ ঘণ্টা পূর্ণমান : ৮০

ক ও খ-বিভাগের সকল প্রশ্নের এবং গ-বিভাগের যে কোনো ৫ (পাঁচ)টি প্রশ্নের উত্তর দাও।

ক-বিভাগ (মান : ২ × ১০ = ২০)

১। **মাইক্রোকন্ট্রোলার কাকে বলে?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২। **ক্যাশ মেমরি বলতে কী বোঝায়?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ২১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৩। **IDE কী?**
উত্তর : অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

৪। **Addressing mode কাকে বলে?**
উত্তর : অনুশীলনী ৪ এর অতি সংক্ষিপ্ত ১৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

৫। **RET নির্দেশনার কাজ কী?**
উত্তর : অনুচ্ছেদ ৪.৩ নং দ্রষ্টব্য।

৬। **CALL নির্দেশনা কাজ কী?**
উত্তর : অনুশীলনী ৬ এর অতি সংক্ষিপ্ত ৭ নং প্রশ্নোত্তর দ্রষ্টব্য।

৭। **ইমবেডেড মাইক্রোকন্ট্রোলার কী?**
উত্তর : অনুশীলনী ১ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

৮। **Line driver এর কাজ লেখ।**
উত্তর : অনুশীলনী ১০ এর অতি সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

৯। **Interrupt service routine বলতে কী বোঝায়?**
উত্তর : অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ৩ নং প্রশ্নোত্তর দ্রষ্টব্য।

১০। **TI ও RI কখন Set হয়?**
উত্তর : অনুশীলনী ১১ এর অতি সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

খ-বিভাগ (মান : ৩ × ১০ = ৩০)

১১। **Microprocessor এবং Microcontroller এর মাঝে পার্থক্য লেখ।**
উত্তর : অনুশীলনী ১ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১২। **PIC Micro controller এর বৈশিষ্ট্যগুলো লেখ।**
উত্তর : অনুশীলনী ২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৩। **8051 Microcontroller এর বৈশিষ্ট্যগুলো লেখ।**
উত্তর : অনুশীলনী ২ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৪। **Commonly used Assembler directives এর তালিকা লেখ।**
উত্তর : অনুশীলনী ৫ এর অতি সংক্ষিপ্ত ৮ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৫। **24 H এবং 35 H ডাটা দুটি যোগ করে যোগফল 2121 H লোকেশনে জমা রাখার জন্য Assembly language program লেখ।**
উত্তর : অনুশীলনী ৫ এর সংক্ষিপ্ত ৫ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৬। **'C' language এর সুবিধাগুলো লেখ।**

**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ৯ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৭। **Serial এবং parallel communication এর ব্যাখ্যা দাও।**

**উত্তর :** অনুচ্ছেদ ১০.১ নং দ্রষ্টব্য।

১৮। **I/O port 8051 এর তালিকা তৈরি কর।**

**উত্তর :** অনুশীলনী ৮ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৯। **LCD এর pin diagram এঁকে চিহ্নিত কর।**

**উত্তর :** অনুশীলনী ১২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২০। **Subroutine এ সুবিধাগুলো লেখ।**

**উত্তর :** অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

গ-বিভাগ (মান : ৬ × ৫ = ৩০)

২১। **Micro controller এর Basic block diagram অঙ্কনপূর্বক প্রত্যেকটি block এর কাজ লেখ।**

**উত্তর :** অনুশীলনী ১ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২২। **Micro controller এর pin diagram এঁকে প্রত্যেকটি pin এর কাজ সংক্ষেপে লেখ।**

**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৩। **8051 Microprocessor এর Addressing mode কয়টি? প্রত্যেকটি Addressing mode উদাহরণসহ সংক্ষিপ্ত বর্ণনা কর।**

**উত্তর :** অনুশীলনী ৪ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৪। **'C' language এ Time delay উৎপন্ন করার প্রক্রিয়া বর্ণনা কর।**

**উত্তর :** অনুশীলনী ৭ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৫। **প্রয়োজনীয় চিত্রসহ Interrupt Enable (IE) register-এর প্রত্যেকটি bit এর বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১১ এর রচনামূলক ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৬। **DC motor সহযোগে 8051 Micro controller এর interfacing প্রক্রিয়া চিত্রসহ বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৬। **'C' language এর সুবিধাগুলো লেখ।**

**উত্তর :** অনুশীলনী ৭ এর সংক্ষিপ্ত ৯ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৭। **Serial এবং parallel communication এর ব্যাখ্যা দাও।**

**উত্তর :** অনুচ্ছেদ ১০.১ নং দ্রষ্টব্য।

১৮। **I/O port 8051 এর তালিকা তৈরি কর।**

**উত্তর :** অনুশীলনী ৮ এর অতি সংক্ষিপ্ত ৬ নং প্রশ্নোত্তর দ্রষ্টব্য।

১৯। **LCD এর pin diagram এঁকে চিহ্নিত কর।**

**উত্তর :** অনুশীলনী ১২ এর সংক্ষিপ্ত ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২০। **Subroutine এ সুবিধাগুলো লেখ।**

**উত্তর :** অনুশীলনী ৬ এর সংক্ষিপ্ত ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

গ-বিভাগ (মান : ৬ × ৫ = ৩০)

২১। **Micro controller এর Basic block diagram অঙ্কনপূর্বক প্রত্যেকটি block এর কাজ লেখ।**

**উত্তর :** অনুশীলনী ১ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২২। **Micro controller এর pin diagram এঁকে প্রত্যেকটি pin এর কাজ সংক্ষেপে লেখ।**

**উত্তর :** অনুশীলনী ৩ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৩। **8051 Microprocessor এর Addressing mode কয়টি? প্রত্যেকটি Addressing mode উদাহরণসহ সংক্ষিপ্ত বর্ণনা কর।**

**উত্তর :** অনুশীলনী ৪ এর রচনামূলক ১ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৪। **'C' language এ Time delay উৎপন্ন করার প্রক্রিয়া বর্ণনা কর।**

**উত্তর :** অনুশীলনী ৭ এর রচনামূলক ২ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৫। **প্রয়োজনীয় চিত্রসহ Interrupt Enable (IE) register-এর প্রত্যেকটি bit এর বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১১ এর রচনামূলক ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

২৬। **DC motor সহযোগে 8051 Micro controller এর interfacing প্রক্রিয়া চিত্রসহ বর্ণনা কর।**

**উত্তর :** অনুশীলনী ১৩ এর রচনামূলক ৪ নং প্রশ্নোত্তর দ্রষ্টব্য।

BDeBooks.com

E-BOOK

www.BDeBooks.com
FB.com/BDeBooksCom
BDeBooks.Com@gmail.com