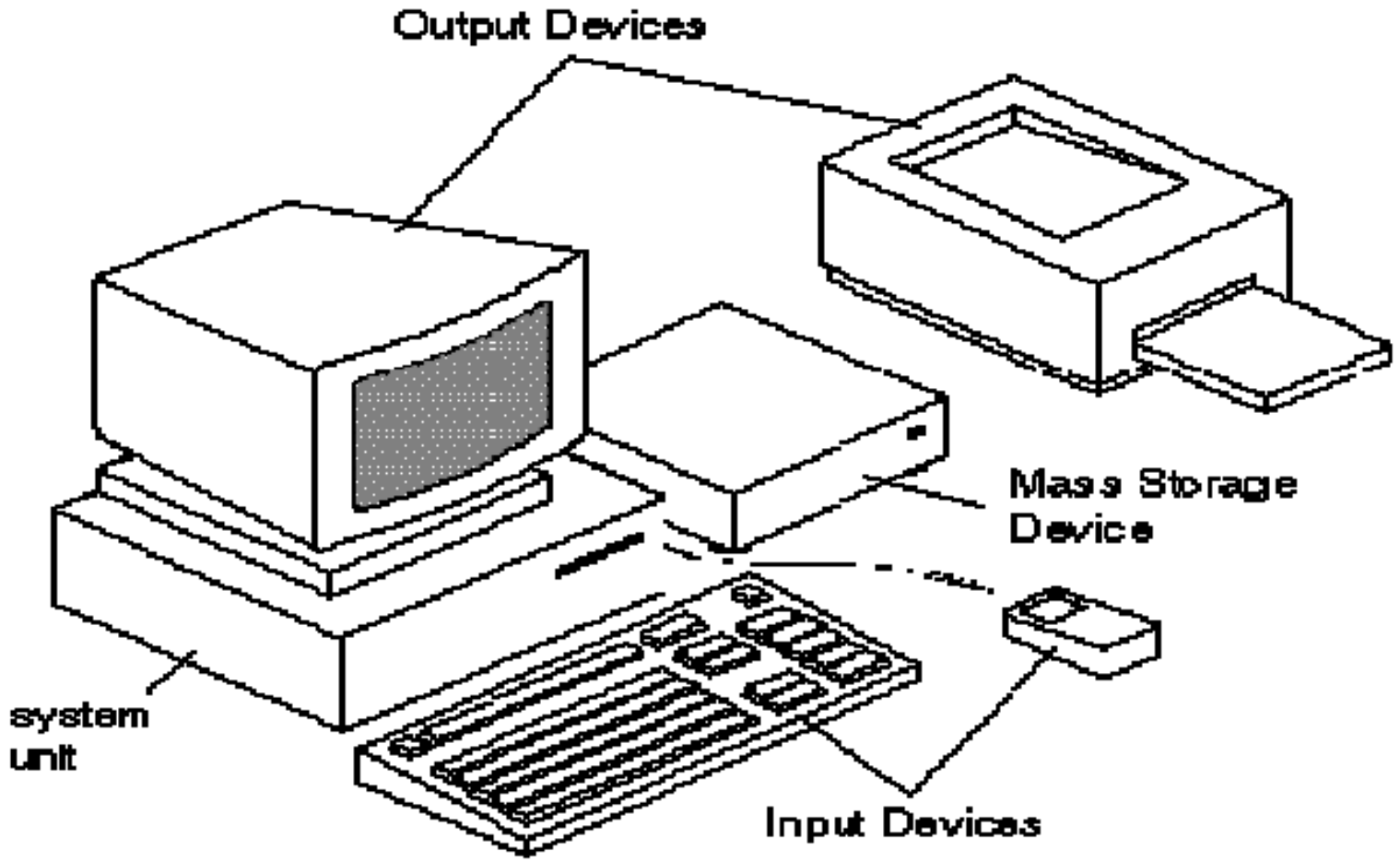
MICROPROCESSOR SYSTEMS

INTRODUCTION TO MICROPROCESSORS

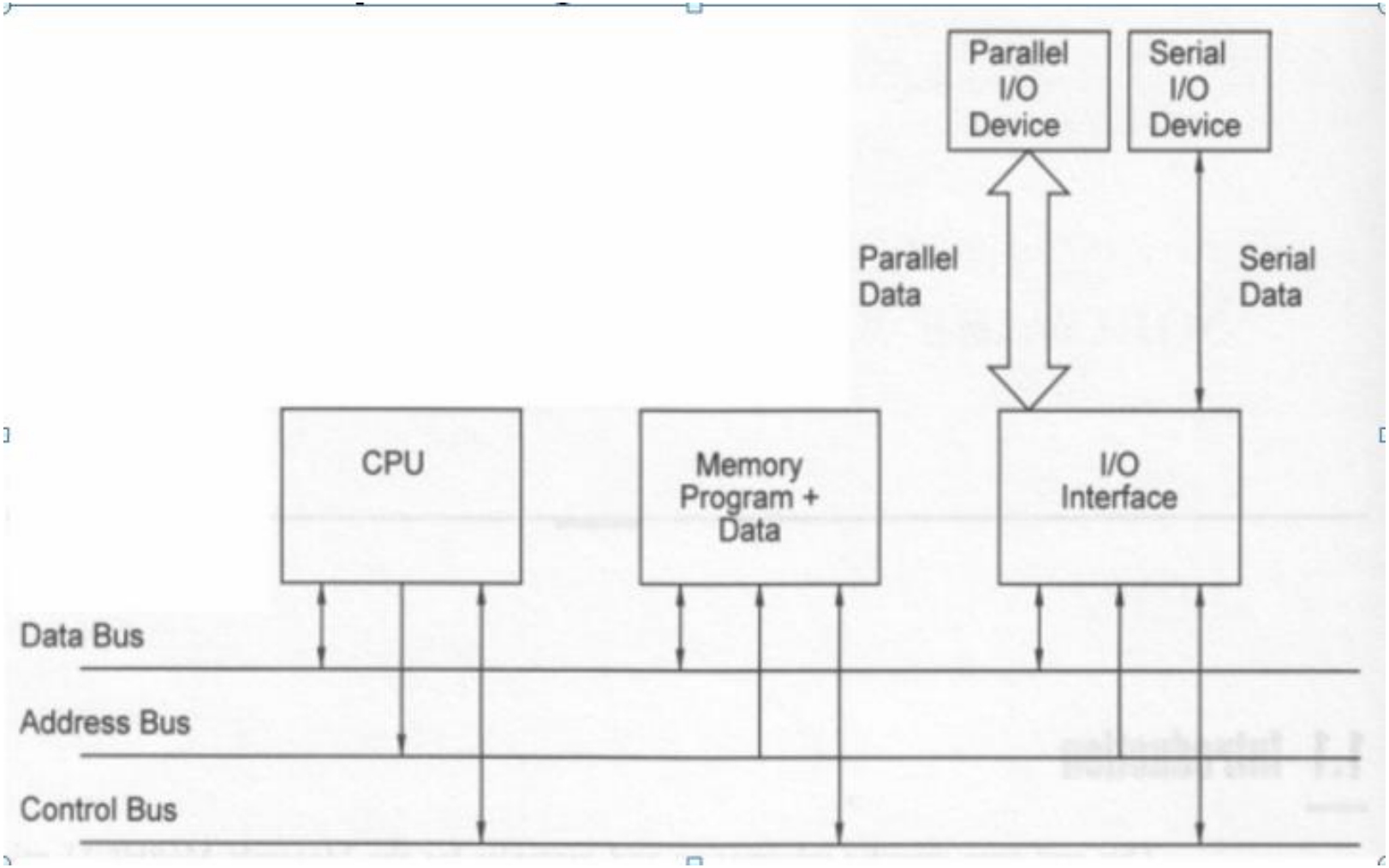
- A computer is a programmable electronic machine.
- Characteristics of a computer are:
  - It responds to a specific set of instructions in a well-defined manner.
  - It can execute a prerecorded list of instructions (a program).


Output Devices
Mass Storage Device
system unit
Input Devices

General classification of computers

- Computers can be generally classified by size, speed, technology, purpose and power as follows, though there is considerable overlap:
- • **Personal computer**: A small, single-user computer based on a microprocessor. In addition to the microprocessor, a personal computer has a keyboard for entering data, a monitor for displaying information, and a storage device for saving data.
- • **Workstation**: A powerful, single-user computer. A workstation is like a personal computer, but it has a more powerful microprocessor and a higher-quality monitor.
- • **Minicomputer**: A multi-user computer capable of supporting from 10 to hundreds of users simultaneously.
- • **Mainframe**: A powerful multi-user computer capable of supporting many hundreds or thousands of users simultaneously.
- • **Supercomputer:** An extremely fast computer that can perform hundreds of millions of instructions per second.

Common computer organization



Hardware components of a computer

- **Memory**: Enables a computer to store, at least temporarily, data and programs.
- **Mass storage device**: Allows a computer to permanently retain large amounts of data. Common mass storage devices include disk drives and tape drives.
- **Input device**: enable data and instructions to be enter in a computer e.g. keyboard and mouse
- **Output device**: Enables the user to understand what the computer has accomplished e.g. a display screen, printer.
- **Central processing unit (CPU):** It executes instructions. In addition to these components, many others make it possible for the basic components to work together efficiently. For example, every computer requires a bus that transmits data from one part of the computer to another.

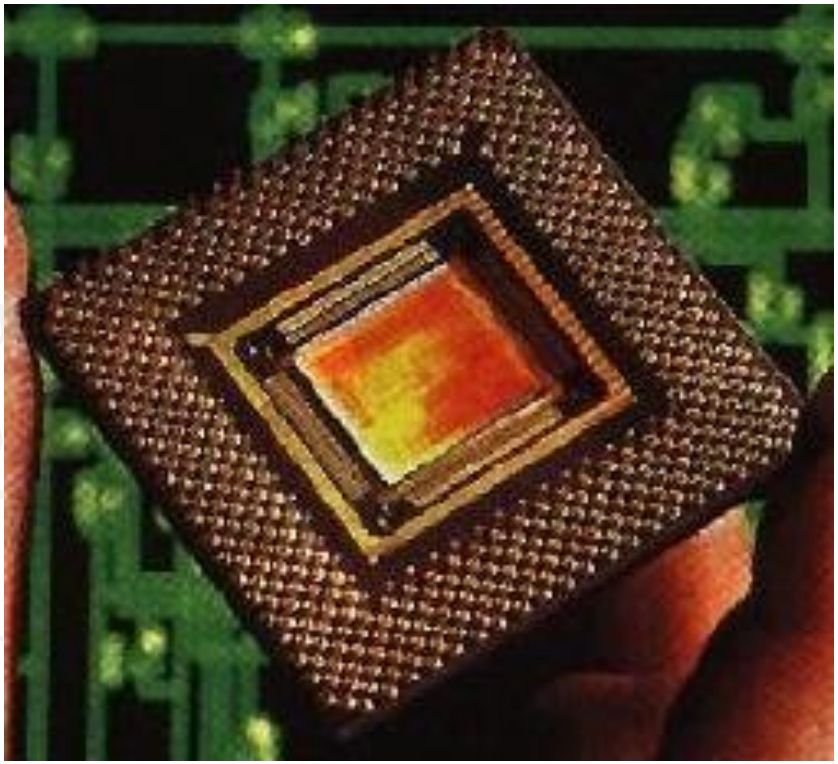
TASKs

- Distinguish between a microprocessor and a microcontroller.
- What is an Embedded system.

What is a microprocessor?

- Is a digital electronic component (in form of a silicon chip) with miniaturized electronic circuits that function as the central processing unit (CPU) of a computer, providing computational control.
- Are high performance, general purpose “brains” for PCs
- A **microprocessor** (sometimes abbreviated **µP**)
- They also control the logic of almost all digital devices, from clock radios, digital-wrist watches, programmable microwaves, digital video recorder to fuel-injection systems for automobiles.
- Are also used in advanced electronic systems, such as computer printers, automobiles and jet airliners.


P8085AH-2
INTEL © 1976



Most prominent features of a Microprocessor

- **Cost-effective** – The microprocessor chips are available at low prices and results its low cost.
- **Size** – The microprocessor is of small size chip, hence is portable.
- **Low Power Consumption** – Microprocessors are manufactured by using metal oxide semiconductor technology, which has low power consumption.
- **Versatility** – The microprocessors are versatile as we can use the same chip in a number of applications by configuring the software program.
- **Reliability** – The failure rate of an IC in microprocessors is very low, hence it is reliable.

Three basic characteristics that differentiate microprocessors

- **Instruction set**: The set of instructions that the microprocessor can execute.
- **Bandwidth**: The number of bits processed in a single instruction.
- **Clock speed**: Given in megahertz (MHz) or gigahertz (GHz), the clock speed determines how many instructions per second the processor can execute.

Classification of Microprocessor

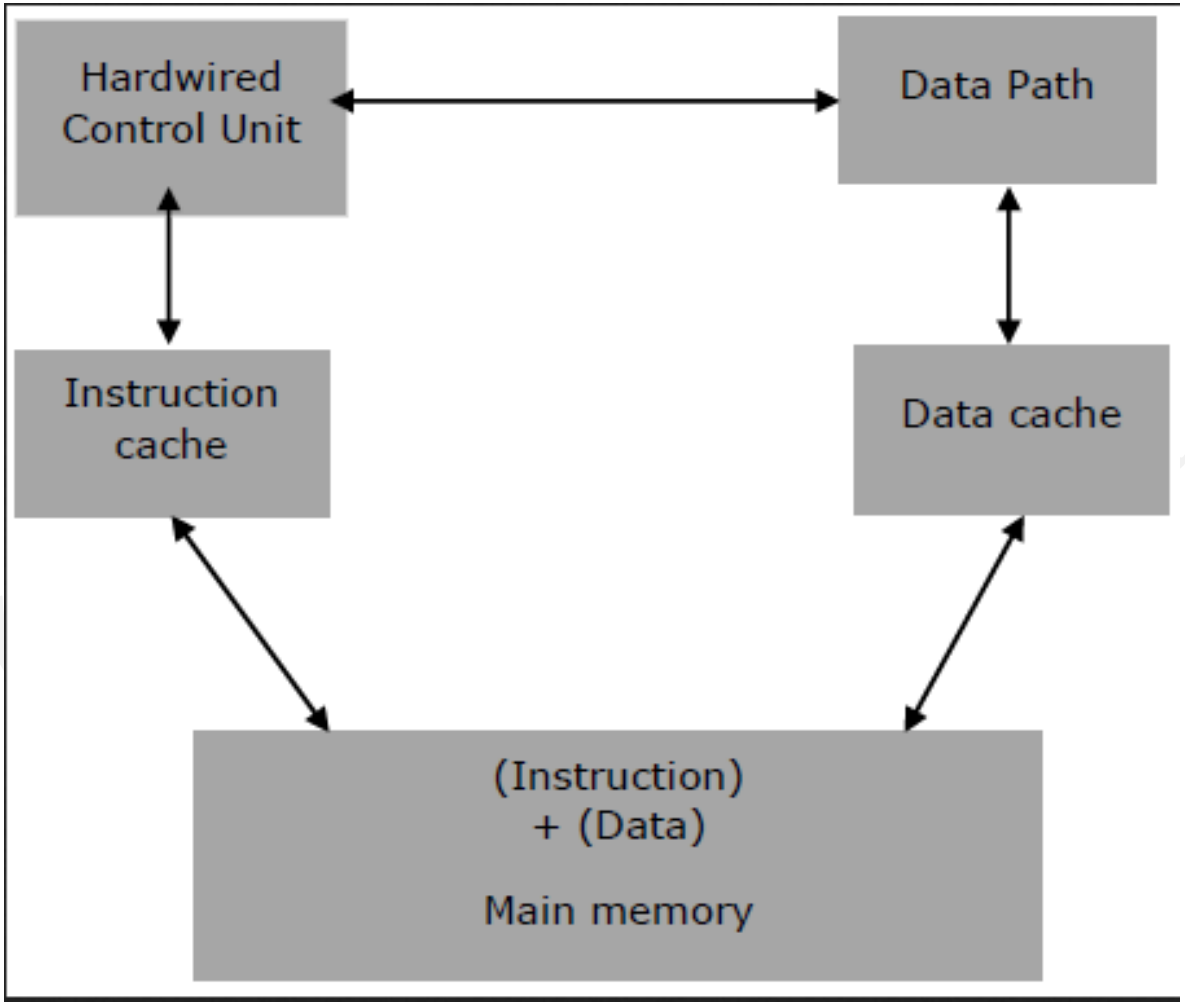
- Are classified by:
  - the semiconductor technology of their design (TTL, transistor-transistor logic; CMOS, complementary-metal-oxide semiconductor; or ECL, emitter-coupled logic),
  - the width of the data format (4-bit, 8-bit, 16-bit, 32-bit, or 64-bit) they process; and
  - their instruction set (CISC , RISC or special processors).

- TTL technology is most commonly used because the technology is not as expensive
- CMOS is favoured for portable computers and other battery-powered devices because of its low power consumption, small size, high speed
- ECL is used where the need for its greater speed offsets the fact that it consumes the most power. Four-bit devices, while inexpensive, are good only for simple control applications; in general, the wider the data format, the faster and more expensive the device.
- CISC processors, which have 70 to several hundred instructions, are easier to program than RISC processors, but are slower and more expensive.

RISC Processor

- **Reduced Instruction Set Computer (**RISC). It is designed to reduce the execution time by simplifying the instruction set of the computer.
- In RISC processors, each instruction requires only one clock cycle to execute results in uniform execution time. This reduces the efficiency as there are more lines of code, hence more RAM is needed to store the instructions.
- The compiler also has to work more to convert high-level language instructions into machine code.
- Some of the RISC processors are –
  - Power PC: 601, 604, 615, 620
  - DEC Alpha: 210642, 211066, 21068, 21164
  - MIPS: TS (R10000) RISC Processor
  - PA-RISC: HP 7100LC

Architecture of RISC



Characteristics of RISC

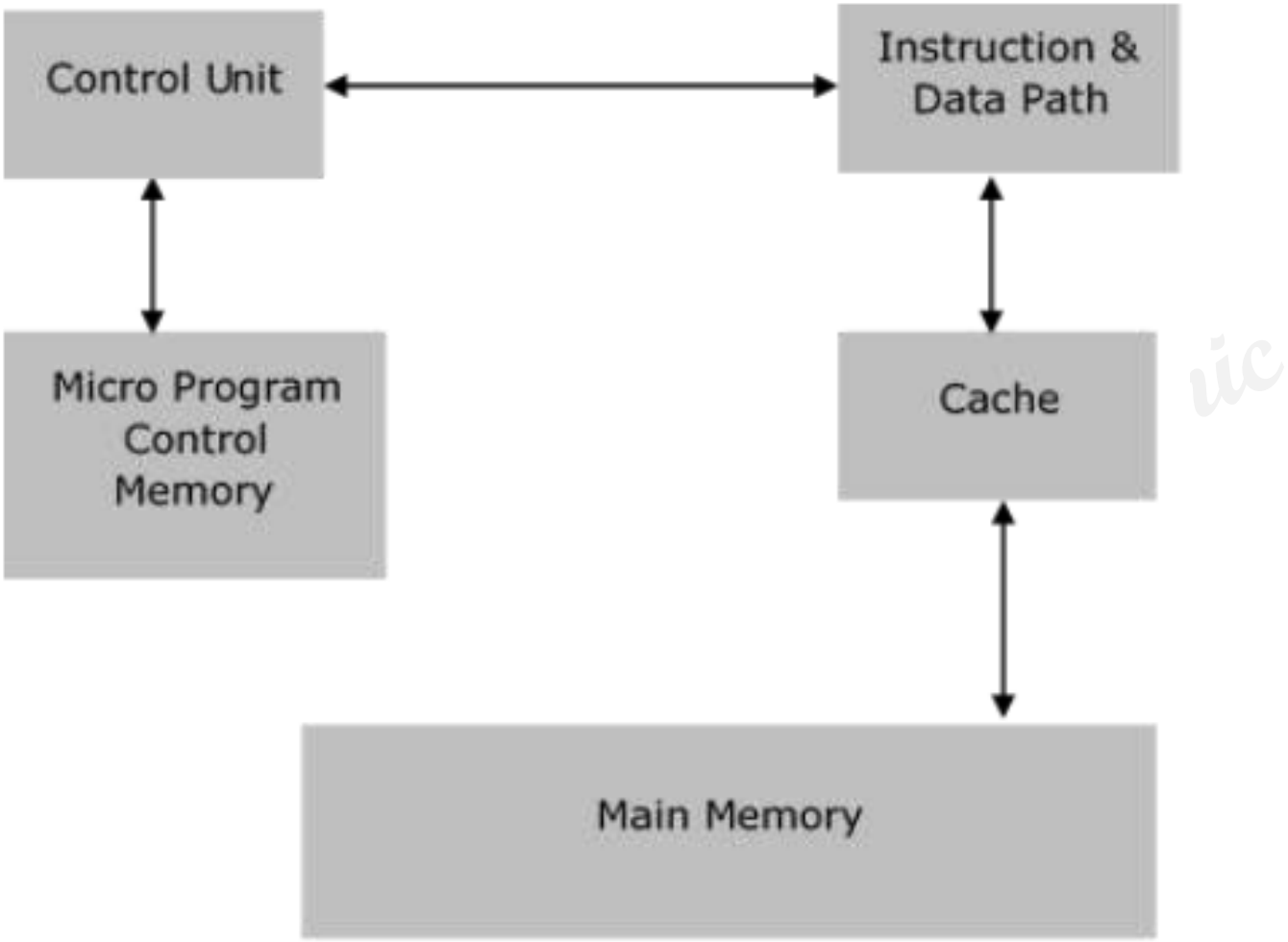
- It consists of simple instructions.
- It supports various data-type formats.
- It utilizes simple addressing modes and fixed length instructions for pipelining.
- It supports register to use in any context.
- One cycle execution time.
- “LOAD” and “STORE” instructions are used to access the memory location.
- It consists of larger number of registers.
- It consists of less number of transistors.

CISC Processor

- CISC stands for **Complex Instruction Set Computer**. It is designed to minimize the number of instructions per program, ignoring the number of cycles per instruction. The emphasis is on building complex instructions directly into the hardware.
- The compiler has to do very little work to translate a high-level language into assembly level language/machine code because the length of the code is relatively short, so very little RAM is required to store the instructions.
- Some of the CISC Processors are −
  - IBM 370/168
  - VAX 11/780
  - Intel 80486

Architecture of CISC

- Its architecture is designed to decrease the memory cost because more storage is needed in larger programs resulting in higher memory cost. To resolve this, the number of instructions per program can be reduced by embedding the number of operations in a single instruction.


Control Unit
Instruction & Data Path
Micro Program Control Memory
Cache
Main Memory

Characteristics of CISC

- Variety of addressing modes.
- Larger number of instructions.
- Variable length of instruction formats.
- Several cycles may be required to execute one instruction.
- Instruction-decoding logic is complex.
- One instruction is required to support multiple addressing modes.

Special Processors

- These are the processors which are designed for some special purposes. E.g.
  - **Coprocessor:** is a specially designed microprocessor, which can handle its particular function many times faster than the ordinary microprocessor. Like math coprocessor from Intel are 8087-used with 8086, 80287-used with 80286, 80387-used with 80386
  - **Input/Output Processor:** It is a specially designed microprocessor having a local memory of its own, which is used to control I/O devices with minimum CPU involvement. E.g. DMA (direct Memory Access) controller, Keyboard/mouse controller, Graphic display controller, SCSI port controller

– **DSP (Digital Signal Processor):** This processor is specially designed to process the analog signals into a digital form. This is done by sampling the voltage level at regular time intervals and converting the voltage at that instant into a digital form. This process is performed by a circuit called an analogue to digital converter, A to D converter or ADC.

- A DSP contains the following components –
- **Program Memory** – It stores the programs that DSP will use to process data.
- **Data Memory** – It stores the information to be processed.
- **Compute Engine** – It performs the mathematical processing, accessing the program from the program memory and the data from the data memory.
- **Input/Output** – It connects to the outside world.

- Applications of DSP processors are:
  - Sound and music synthesis
  - Audio and video compression
  - Video signal processing
  - 2D and 3d graphics acceleration.
- **For example** – Texas Instrument's TMS 320 series, e.g., TMS 320C40, TMS320C50.

Central Processing Unit

- The CPU (Central Processing Unit) is the combination of the control logic, associated registers and the arithmetic logic unit. The main functions of the CPU are:
- data transfer
- perform arithmetic and logic operations
- control and perform decision making (instructional flow control)
- The register array consists of at least one accumulator, program counter and stack pointer.
- The control unit controls all the operations in a CPU and basically it puts the CPU in one of the fetch and execution phases

Control Unit

- CU generates signals within uP that carry out the instruction, which has been decoded.
- In reality causes certain connections between blocks of the uP to be opened or closed, so that data goes where it is required, and so that ALU operations occur

Arithmetic Logic Unit

- The ALU is a circuitry in the CPU that performs the actual numerical and logic operation such as ‘add’, ‘subtract’, ‘AND’, ‘OR’, ‘XOR’, ‘increment’, ‘decrement’, ‘shift’, ‘clear’, etc, on certain on-chip registers.
- It uses data from memory and from Accumulator to perform arithmetic. Always stores result of operation in Accumulator

Registers

- The CPU includes different types of registers e.g.
  - an accumulator,
  - flag register,
  - stack pointer,
  - program counter,
  - Instruction register
  - Memory Address Register
  - Memory Data Register
  - general purpose registers e.t.c..

- **Control Generator**
- Generates signals within uP to carry out the instruction which has been decoded. In reality causes certain connections between blocks of the uP to be opened or closed, so that data goes where it is required, and so that ALU operations occur.
- **Register Selector**
- This block controls the use of the register stack in the example. Just a logic circuit which switches between different registers in the set will receive instructions from Control Unit.

System buses

- A bus is a group of parallel conduction paths, lines or conductors carrying digital information, in binary form, between digital devices in a system.
- There are three main types of system buses:
  - the data bus
  - the control bus
  - the address bus

Data bus:

- carries data bits in binary form, which is stored in memory (or an I/O device) to the CPU or from the CPU to the memory (or other I/O devices).
- Data bus is bi-directional.
- Size of the data bus determines what arithmetic can be done. If only 8 bits wide then largest number is 11111111 (255 in decimal).
- Therefore, larger number has to be broken down into chunks of 255. This slows microprocessor.
- Data Bus also carries instructions from memory to the microprocessor. A 16-bit size data bus would limits the number of possible instructions to 65536, each specified by a separate number.

Control bus:

- is a collection of control signals that coordinate and synchronize the order of operation of the whole system.
- Control Bus are various lines which have specific functions for coordinating and controlling µP operations.
- Eg: Read/Write line, single binary digit. Controls whether memory is being ‘written to’ (data stored in mem) or ‘read from’ (data taken out of mem) 1 = Read, 0 = Write. May also include clock line(s) for timing/synchronising, ‘interrupts’, ‘reset’ etc.
- Control Bus carries control signals partly unidirectional, partly bi-directional.
- Larger buses allow larger number of instructions, more memory locations, and faster arithmetic.
- In the microprocessor the three buses are external to the chip (except for the internal data bus). The chip connects to the buses via buffers.

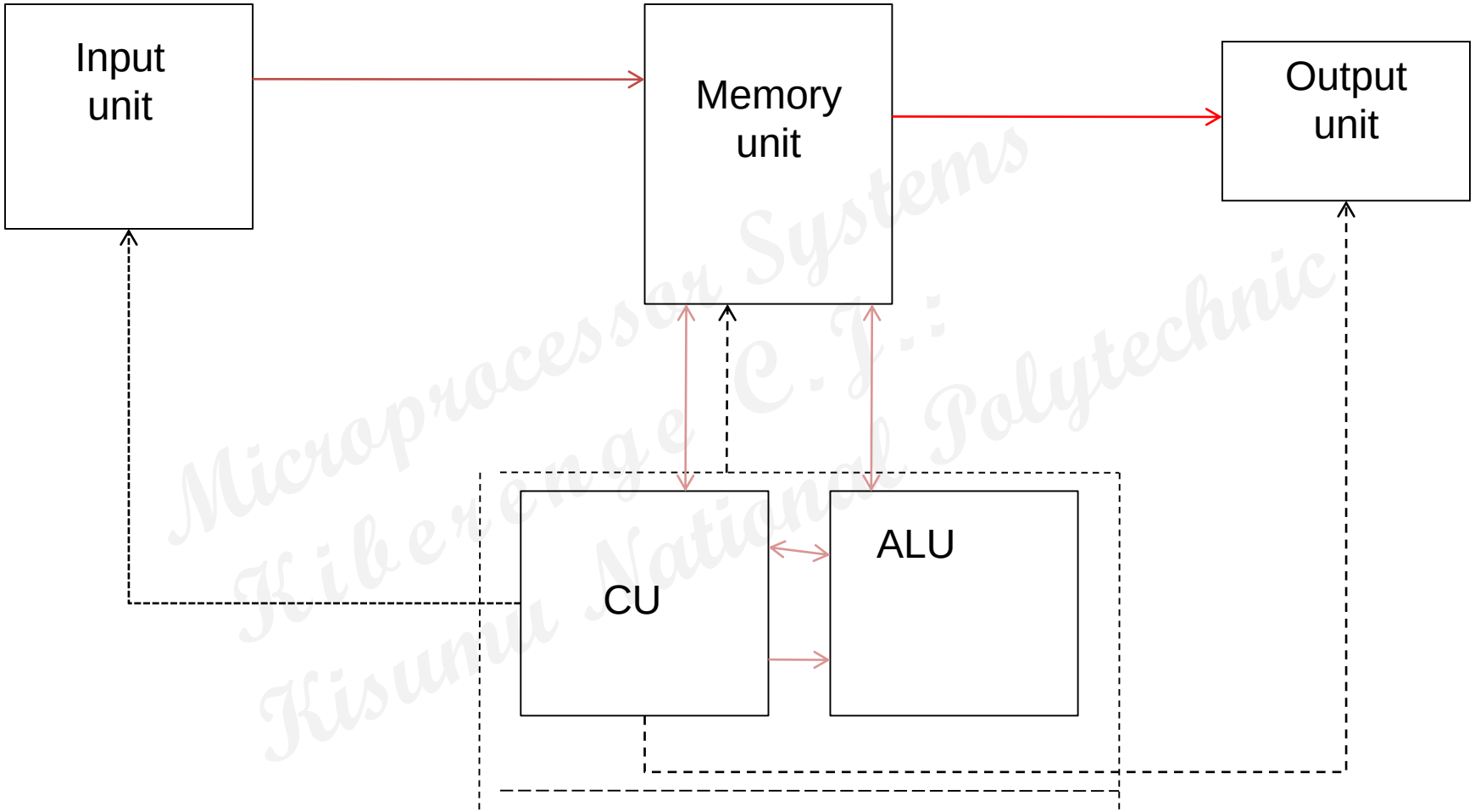
Address bus:

- It carries device location (address) information of a unique memory or input/output (I/O) device.
- It is **unidirectional**, i.e. one-way. Address bits are only sent from microprocessor to memory (or I/O device), not the other way.
- A 16-bit Address Bus consists of 16 wires, one wire per bit. Its "width" is therefore 16 bits. A 16 bit binary number allows 2^{16} different numbers, i.e. 0000000000000000 up to 1111111111111111.
- Because memory consists of boxes, each with a unique address, the size of the address bus determines the size of memory, which can be used.
- To communicate with memory the microprocessor sends an address on the address bus, eg 0000000000000011 (3 in decimal), to the memory. The memory logic then selects box number 3 for reading or writing data.

- Question?: If you have a memory chip of size 256 kilobytes (256 x 1024 x 8 bits), how many wires does the address bus need, in order to be able to specify an address in this memory? **Note:** the memory is organized in groups of 8 bits per location, therefore, how many locations must you be able to specify?

Micro-processor based system

consist of five different types of functional units



The Intel 8085 uP

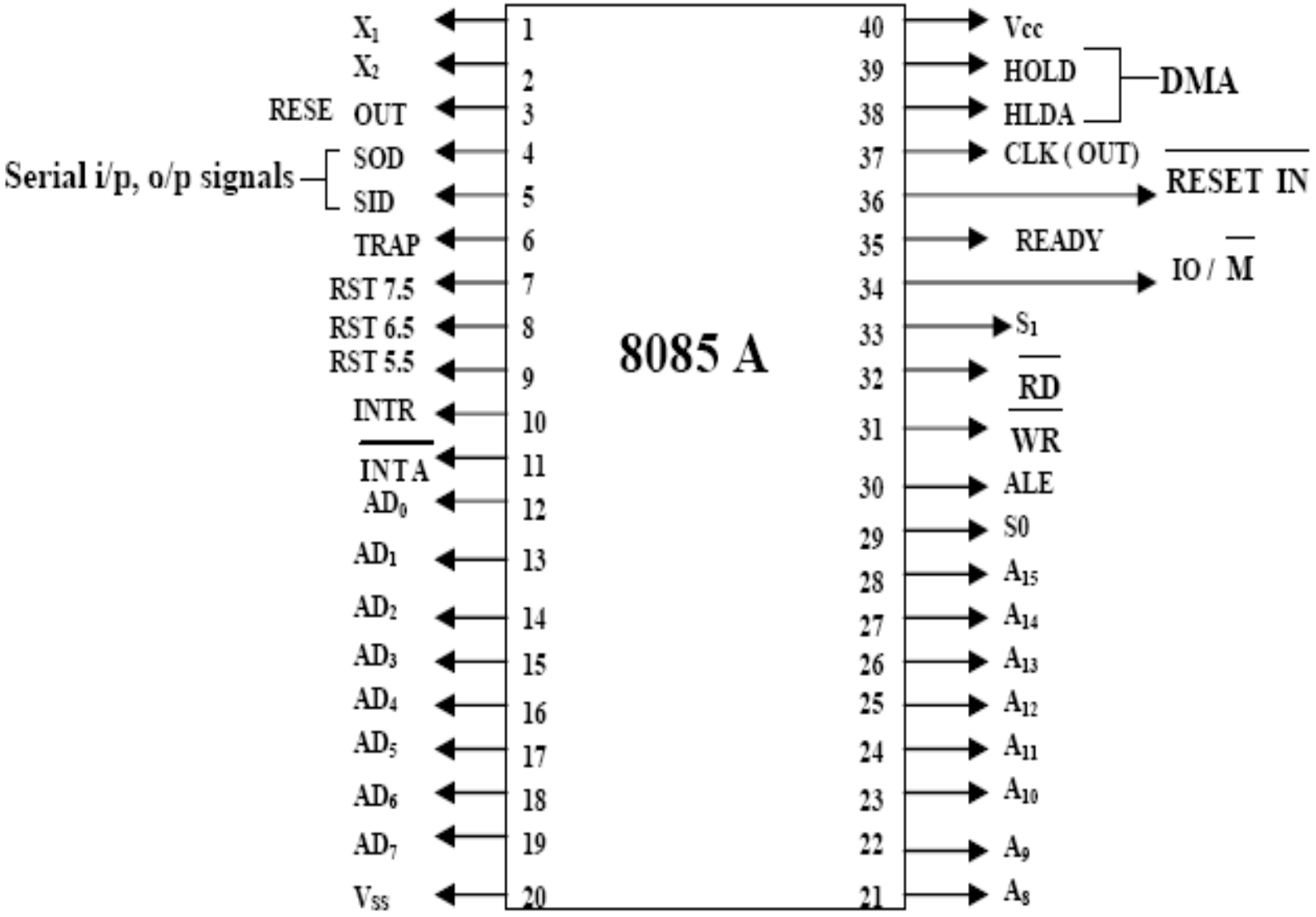
- The 8085 was manufactured from 1977 to 1990s by Intel using NMOS technology.
- It has a CPU Clock rate of 3MHz, 5MHz and 6MHz. It has minimum feature size 3um.

The salient features of 8085 µp are:

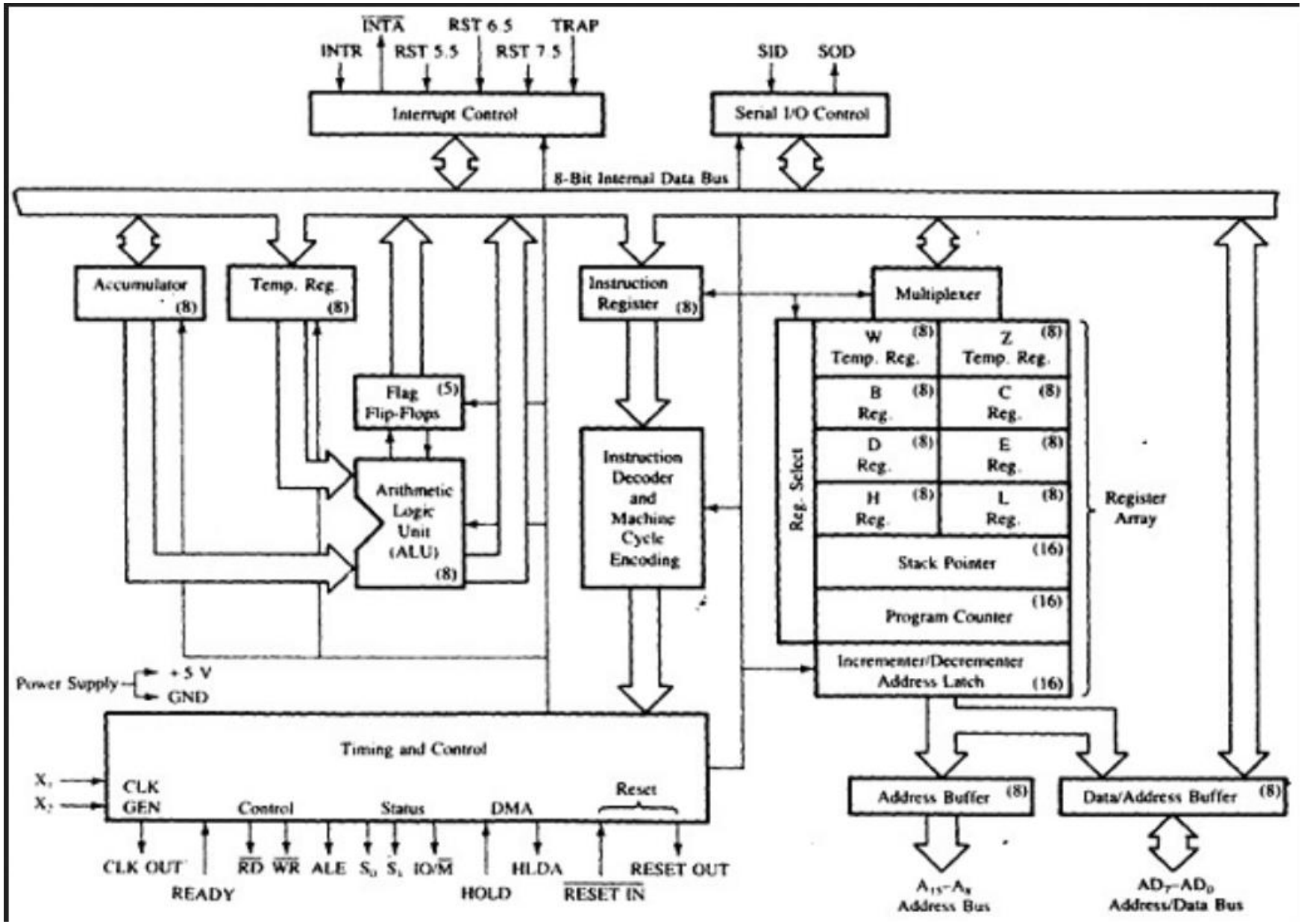
- It is a 8 bit microprocessor thus has 8-bit data bus
- It is manufactured with N-MOS technology.
- It has 16-bit address bus hence it can address upto 2^{16} = 65536 bytes (64KB) memory locations through A_0-A_{15}.
- The first 8 lines of address bus and 8 lines of data bus are multiplexed $AD_0 – AD_7$.
- Data bus is a group of 8 lines $D_0 – D_7$.
- It supports external interrupt request.

The salient features of 8085 µp are:

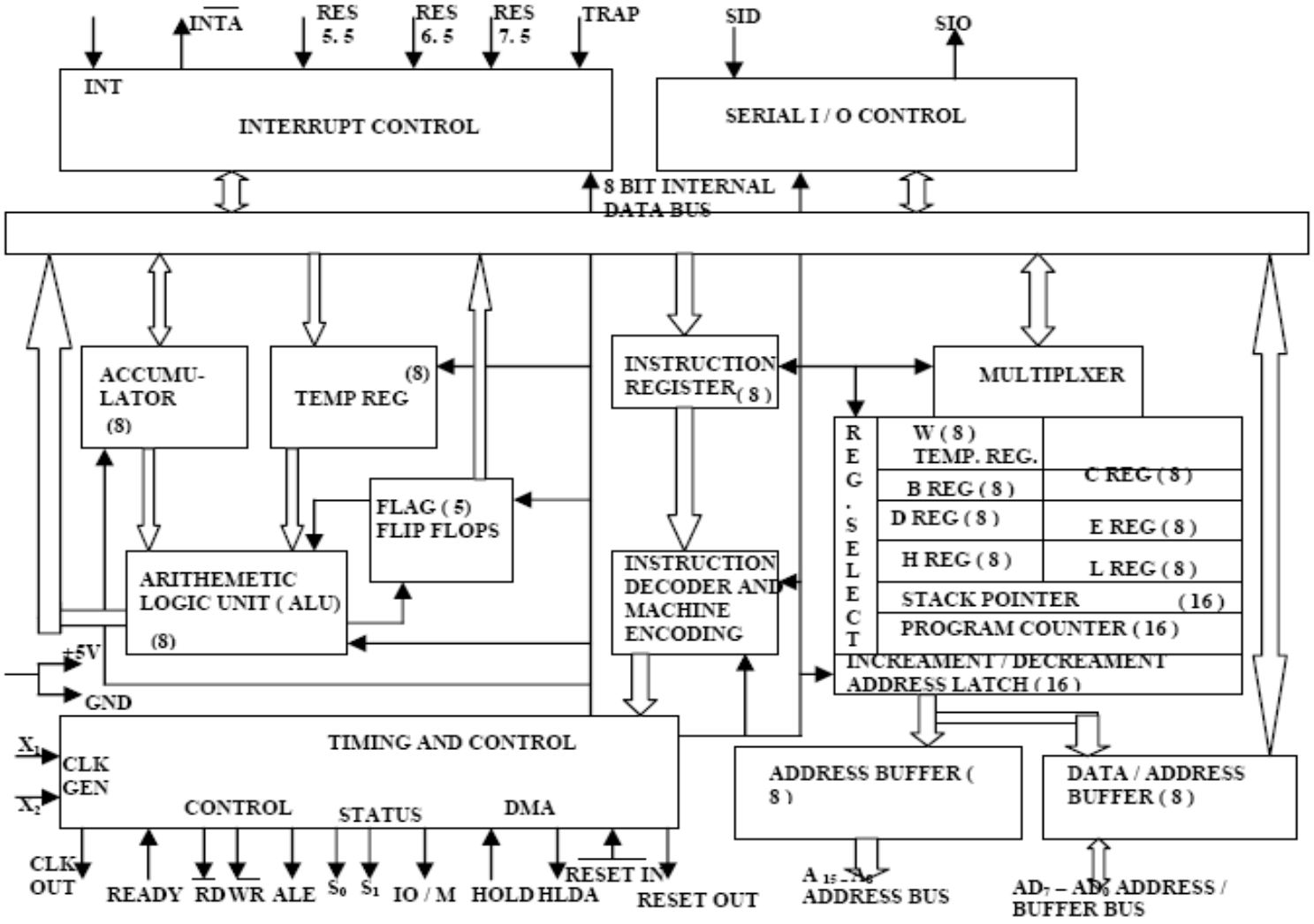
- A 16 bit stack pointer (SP)
- Six independent 8-bit general purpose registers B, C, D, E, H and L or paired as BC, DE, HL.
- It requires a signal +5V power supply by using depletion mode transistors.
- It is operates at 3.2 MHZ single phase clock.
- It is enclosed with 40 pins DIP (Dual in line package).
- It requires a signal +5V power supply and operates at 3.2 MHZ single phase clock.
- It is enclosed with 40 pins DIP (Dual in line package).
- It is used in washing machines, microwave ovens, mobile phones, and many microcontroller circuits.



Pin Diagram of 8085


INTA
RST 6.5
TRAP
INTR
RST 5.5
RST 7.5
SID
SOD
Interrupt Control
Serial I/O Control
8-Bit Internal Data Bus
Accumulator (8)
Temp. Reg. (8)
Instruction Register (8)
Multiplexer
Flag Flip-Flops (5)
Arithmetic Logic Unit (ALU) (8)
Instruction Decoder and Machine Cycle Encoding
W Temp. Reg. (8)
Z Temp. Reg. (8)
B Reg. (8)
C Reg. (8)
D Reg. (8)
E Reg. (8)
H Reg. (8)
L Reg. (8)
Reg. Select
Register Array
Stack Pointer (16)
Program Counter (16)
Incrementer/Decrementer Address Latch (16)
Power Supply
+5 V
GND
Timing and Control
X1
X2
CLK GEN
Control
Status
DMA
Reset
CLK OUT
READY
RD WR ALE S0 S1 IO/M
HOLD
HLDA
RESET IN
RESET OUT
Address Buffer (8)
Data/Address Buffer (8)
A15-A8
Address Bus
AD7-AD0
Address/Data Bus

The internal architecture of 8085 uP



The general-purpose registers

- Registers B, C, D, E, H and L are general purpose.
- Each of these registers is 8-bit wide
- These registers increase the uPs versatility.
- Using data copy instructions, the programmer can use these registers to store or copy data.
- Some instructions may combine these registers into register pairs to perform 16 bit operations and also as data pointers to reference memory addresses.

General Purpose Registers

- **Reg B** and **Reg C** can be used as one **16-bit BC register** pair. When used as a pair, the C register contains low-order byte. the B register contains high-order byte.
- **Reg D and Reg E** can be used as one 16-bit DE register pair. When used as a pair the E register contains low-order byte.
- **Reg H and Reg L** can be used as one 16-bit HL register pair. When used as a pair the L register contains low-order byte.

Special purpose registers

- **Program Counter (PC);** is a 16-bit register. This register deals with sequencing the execution of instructions. This register is a memory pointer.

  The uP uses this register to sequence the execution of the instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched.

  When a byte (machine code) is being fetched, the program counter is incremented by one to point to the next memory location
- **Stack Pointer (SP); I**s also a register used as a memory pointer. It points to a memory location in R/W memory, called the stack. The beginning of the stack is defined by loading address in the stack pointer.

Special purpose registers

- **Instruction Register/Decoder;** Temporary store for the current instruction of a program. Latest instruction are sent here from memory prior to execution. Decoder then takes instruction and 'decodes' or interprets the instruction. Decoded instruction then passed to next stage.
- **Memory Address Register;** Holds address, received from PC, of next program instruction. Feeds the address bus with addresses of location of the program under execution.

Special purpose registers

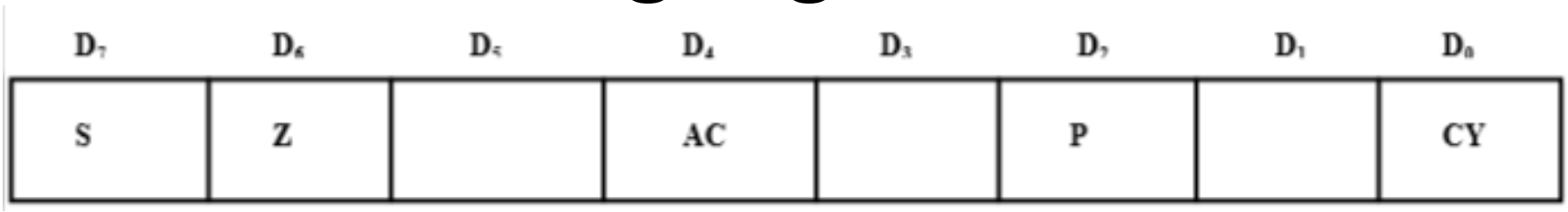
- **Stack pointer** is a 16 bit register. This register is always decremented/incremented by 2 during push and pop.
- **Accumulator or A register** is an 8-bit register used for arithmetic, logic, I/O and load/store operations. The accumulator register is a part of arithmetic/logic unit (ALU).

  Is used to perform arithmetic and logical operations and result of these operations are stored in the accumulator.
- **Flag Register** has five 1-bit flags.

Flag register

- **Flags** are flip-flops in the ALU, which are set or reset after an operation according to data conditions of the result in the accumulator and other registers.
- The uP uses these flags to test data conditions and in the decision-making process.
- Examples are the Zero(Z) flag, Carry (CY), Sign (S), Parity (P), and Auxiliary Carry (AC) flags. For example, when an arithmetic operation results in zero, the flip-flop called the Zero(Z) flag is set to one.

Flag register



Flag Register has five 1-bit flags

i. **S**ign - set if the MSB of the result is set. That is when accumulator bit D7 is 1
ii. **Z**ero - set if the result in accumulator is zero.
iii. **A**uxiliary **C**arry - set if there was a carry out from bit 3 to bit 4 of the result.
iv. **P**arity - set if the parity (the number of set bits in the result) is even.
v. **C**arry - set if there was a carry during addition, or borrow during subtraction/comparison/rotation.

- **A6 - A1s** (Output 3 State)

  Address Bus; The most significant 8 bits of the memory address or the 8 bits of the I/0 address,3 stated during Hold and Halt modes.
- **AD0 - AD7** (Input/Output 3-state): Multiplexed Address/Data Bus; Lower 8 bits of the memory address (or I/0 address) appear on the bus during the first clock cycle of a machine state. It then becomes the data bus during the second and third clock cycles. 3 stated during Hold and Halt modes.
- **ALE** (Output): Address Latch Enable: It occurs during the first clock cycle of a machine state and enables the address to get latched into the on chip latch of peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. ALE can also be used to strobe the status information. ALE is never 3stated.

- **S0, S1** (Outputs) indicate Data Bus Status. Thus encoded status of the bus cycle:

  S1 S0 : 00-HALT 01-WRITE 10-READ 11-FETCH

  S1 can be used as an advanced R/W status.
- **RD** (Output 3state) READ; indicates the selected memory or 1/0 device is to be read and that the Data Bus is available for the data transfer.
- **WR** (Output 3state) WRITE; indicates the data on the Data Bus is to be written into the selected memory or 1/0 location. Data is set up at the trailing edge of WR. 3stated during Hold and Halt modes.

Reset signals

- **RESET IN**: When this signal goes low, the program counter (PC) is set to Zero, µp is reset and resets the interrupt enable and HLDA flip-flops. The data and address buses and the control lines are 3-stated during RESET and because of asynchronous nature of RESET, the processor internal registers and flags may be altered by RESET with unpredictable results. RESET IN is a Schmitt-triggered input, allowing connection to an R-C network for power-on RESET delay
- Upon power-up, RESET IN must remain low for at least 10 ms after minimum Vcc has been reached. • For proper reset operation after the power – up duration, RESET IN should be kept low a minimum of three clock periods. • The CPU is held in the reset condition as long as RESET IN is applied. Typical Power-on RESET RC values R1 = 75KΩ, C1 = 1µF.
- **RESET OUT**: This signal indicates that µp is being reset. This signal can be used to reset other devices. The signal is synchronized to the processor clock and lasts an integral number of clock periods

interrupts

- 8085uP has 5 interrupts.
- They are presented below in the order of their priority (from lowest to highest):
  - **I.** **INTR** is maskable and non-vectored When the interrupt occurs the processor fetches from the bus of the 8 RST instructions (RST0 - RST7). The processor saves current program counter into stack and branches to memory location N * 8 (where N 3-bit number from 0 to 7 supplied with the RST instruction). CALL instruction (3 byte instruction). The processor calls the subroutine, address which is specified in the second and third bytes of the instruction.
  - **II.** **RST5.5** is a maskable and vectored interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 2CH (hexadecimal) address.
  - **III.** **RST6.5** is a maskable and vectored interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 34H (hexadecimal) address.
  - **IV.** **RST7.5** is a maskable and vectored interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 3CH (hexadecimal) address.
  - **V.** **TRAP** is a non-maskable and vectored interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 24H (hexadecimal) address.
- All maskable interrupts can be enabled or disabled using EI and DI instructions. RST 5.5, RST6.5 and RST7.5 interrupts can be enabled or disabled individually using SIM instruction.

- **SID** - Serial Input Data Line: is used for Serial communication, the data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.
- **SOD** – Serial Output Data Line: is used for Serial communication , the SIM instruction loads the value of bit 7 of the accumulator into SOD latch if bit 6 (SOE) of the accumulator is 1. The output SOD is set or reset as specified by the SIM instruction
- **HLD** (Input): Is HOLD signal. It indicates that another master is requesting the use of the address and data buses. The CPU, upon receiving the Hold request. will relinquish the use of buses as soon as the completion of the current machine cycle. Internal processing can continue.
- **HLDA** (Output): Is HOLD ACKNOWLEDGE; indicates that the CPU has received the Hold request and that it will relinquish the buses in the next clock cycle. HLDA goes low after the Hold request is removed. The CPU takes the buses one half clock cycle after HLDA goes low.
- **READY (Input) .** This signal Synchronizes the fast CPU and the slow memory, peripherals. If Ready is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If Ready is low, the CPU will wait for Ready to go high before completing the read or write cycle.

- **X1, X2 (Input)**

  Crystal or R/C network connections to set the internal clock generator X1 can also be an external clock input instead of a crystal. The input frequency is divided by 2 to give the internal operating frequency**.**
- **CLK (Output)**

  Clock Output for use as a system clock when a crystal or R/ C network is used as an input to the CPU. The period of CLK is twice the X1, X2 input period.
- **IO/M (Output)**

  IO/M indicates whether the Read/Write is to memory or l/O Tristated during Hold and Halt modes.
- **Vcc** +5 volt supply.
- **Vss** Ground Reference.

TASK

1. Give a general block diagram of a microprocessor based system. Explain briefly the various blocks of the system.
2. Give some examples of the types of devices used for each block you have stated in question 1 above.
3. What is a microprocessor? Sketch and explain the various pins of 8085.
4. Explain what each of these registers are generally used for:
   - SP
   - PC
   - IR
   - MAR
   - HL
5. Explain the architecture of 8085 with the help of its internal block schematic.
6. Write the flag register and explain each of the flags with an example.
7. Mention the list of registers of 8085 that are accessible to the programmer.
8. State and explain some control signals of the 8085 uP.
9. Discuss the relevance of the TRAP interrupt in the 8085 uP.

- Perform operations below and write:
  i. the contents of the accumulator immediately after the operation (3 marks)
  ii. state the status of each flag immediately after the operation. (2 marks)

1. 2CH + D4H
2. 1010 1010 + 1011 1100