Final Assessment│Fall 2022

Md. Shafayet Hossain

CSE- 21st Batch

Microprocessor & Interfacing

Code: CSE- 413│**ID:** 2121210071

Answer to the Question no. 1 (a)

❖ Basic Blocks of a Microcomputer

A microcomputer has three basic blocks: a central processing unit (CPU), a memory unit, and an input/output (I/O) unit. The CPU executes all the instructions and performs arithmetic and logic operations on data. The CPU of the microcomputer is called the *microprocessor* typically a single VLSI (very large scale integration) chip that contains all the registers and control unit, and arithmetic-logic circuits of the microcomputer.

A *memory unit* stores both data and instructions. The memory section typically contains ROM and RAM chips. The ROM can only be read and is nonvolatile; that is, it retains its contents when the power is turned off. A ROM is typically used to store instructions and data that do not change. For example, it might store a table of seven-segment codes for outputting data to a display external to the microcomputer for turning on a digit from 0 through 9. One can read from and write into a RAM. The RAM is volatile; that is, it does not retain its contents.

❖ Microprocessor Features

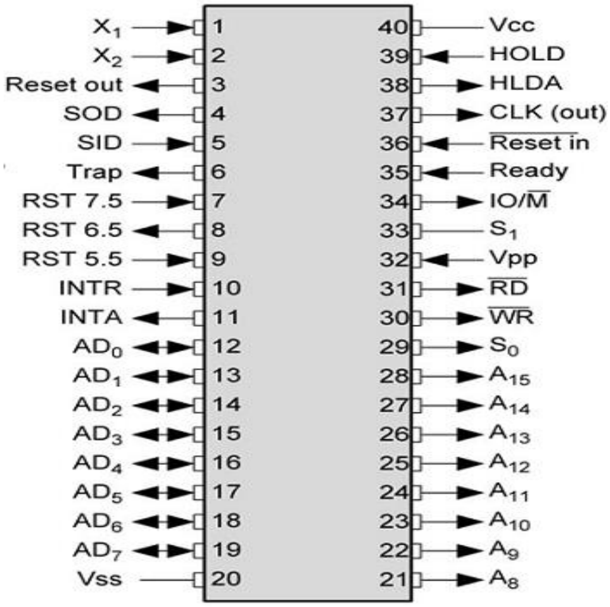
Microprocessor is the heart of computer systems and it is called as Central Processing Unit (CPU). CPU consists of Arithmetic and logical unit (ALU) to perform mathematical and comparison operations register to store inputs/ results / intermediate values and control unit to monitor/control the entire operations. Microprocessors with read-only memory to hold boot programs and random access memory to store processed results can act as full-fledged computers and such microcomputers are fitted into physical devices for monitoring its operations.

Microprocessor is used in a variety of applications due to their unique features such as size, weight, cost, high computing power, and low power consumption, etc., Microprocessor fitted systems are used.

1. to monitor and control operations of Industrial devices by measuring key parameters like temperature, pressure, speed.
2. in instruments to raise an alert or warning on extreme conditions
3. to automate office work / business processes and improve white collar productivity
4. in simplifying publishing activity
5. to speed up the information exchange through Telephone and Satellite network
6. in rolling out innovations in entertainment, games and Photography

Answer to the Question no. 1 (b)

The 8085 PIN Configuration-



The pins of a 8085 microprocessor can be classified into seven groups −

Address bus

A15-A8, it carries the most significant 8-bits of memory/IO address.

Data bus

AD7-AD0, it carries the least significant 8-bit address and data bus.

Control and status signals

These signals are used to identify the nature of operation. There are 3 control signal and 3 status signals.

Three control signals are RD, WR & ALE.

- **RD** − This signal indicates that the selected IO or memory device is to be read and is ready for accepting data available on the data bus.
- **WR** − This signal indicates that the data on the data bus is to be written into a selected memory or IO location.
- **ALE** − It is a positive going pulse generated when a new operation is started by the microprocessor. When the pulse goes high, it indicates address. When the pulse goes down it indicates data.

Three status signals are IO/M, S0 & S1.

IO/M

This signal is used to differentiate between IO and Memory operations, i.e. when it is high indicates IO operation and when it is low then it indicates memory operation.

S1 & S0

These signals are used to identify the type of current operation.

Power supply

There are 2 power supply signals − VCC & VSS. VCC indicates +5v power supply and VSS indicates ground signal.

Clock signals

There are 3 clock signals, i.e. X1, X2, CLK OUT.

- **X1, X2** − A crystal (RC, LC N/W) is connected at these two pins and is used to set frequency of the internal clock generator. This frequency is internally divided by 2.
- **CLK OUT** − This signal is used as the system clock for devices connected with the microprocessor.

Interrupts & externally initiated signals

Interrupts are the signals generated by external devices to request the microprocessor to perform a task. There are 5 interrupt signals, i.e. TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR. We will discuss interrupts in detail in interrupts section.

- **INTA** − It is an interrupt acknowledgment signal.
- **RESET IN** − This signal is used to reset the microprocessor by setting the program counter to zero.
- **RESET OUT** − This signal is used to reset all the connected devices when the microprocessor is reset.
- **READY** − This signal indicates that the device is ready to send or receive data. If READY is low, then the CPU has to wait for READY to go high.
- **HOLD** − This signal indicates that another master is requesting the use of the address and data buses.
- **HLDA (HOLD Acknowledge)** − It indicates that the CPU has received the HOLD request and it will relinquish the bus in the next clock cycle. HLDA is set to low after the HOLD signal is removed.

Serial I/O signals

There are 2 serial signals, i.e. SID and SOD and these signals are used for serial communication.

- **SOD** (Serial output data line) − The output SOD is set/reset as specified by the SIM instruction.
- **SID** (Serial input data line) − The data on this line is loaded into accumulator whenever a RIM instruction is executed.

Answer to the Question no. 1 (c)

Classification of Micro-processor:

Microprocessors are specifically three types, and their names are CISC, RISC and EPIC..

CISC processor: is used to limit the quantity of instructions in step with application.
This processor needs a small size of RAM (Random get entry to memory). We use a few registers if a couple of operations require simplest a single preparation. The compiler works too little to assemble the facts from excessive-level to low-degree language or system language because the length of the CISC processor code is concise.
Instance of CISC processor is Intel 386, Intel 486, Pentium, Pentium II, Pentium seasoned, IBM 370, IBM 268, VAX eleven/780 and many others.

RISC Processor: The whole name of the RISC processor is the decreased instruction Set pc. by way of the name, it's miles recognized that the instruction is pretty easy and completed fast. The RISC processor's instructions get finished by simplest one clock cycle. This preparation also used some addressing modes. The RISC processor is used multiple registers, so the interaction with reminiscence is drastically much less.
Example of CISC processor is IBM RS6000, DEC Alpha 21064, DEC Alpha 21164, DEC Alpha 210642, DEC Alpha 211066, DEC Alpha 211066, DEC Alpha 21068, DEC Alpha 21164, electricity pc 601, strength pc 604, power laptop 615, power computer 620, HP 7100LC and so on.

EPIC Processor: The overall name of the EPIC processor is Explicitly Parallel education Computing. by the name, it is recognized that the guidance works parallelly by way of the use of a compiler. This guidance works very complexly. a few clock frequencies procedure the complex instruction of the EPIC processor. EPIC processor can encode the training.
Furthermore, this processor can encode 128-bit bundles of guidance. every bundle consists of 3 complete commands in the set of 128-bit bundles. each of the three commands encoded in 41 bits is likewise a template 5-bit training. The five-bit template practise of the EPIC processor consists of the kind of the facts and also knows which preparation can be processed parallel.

Answer to the Question no. 2 (a)

How DMA Operations are Performed:

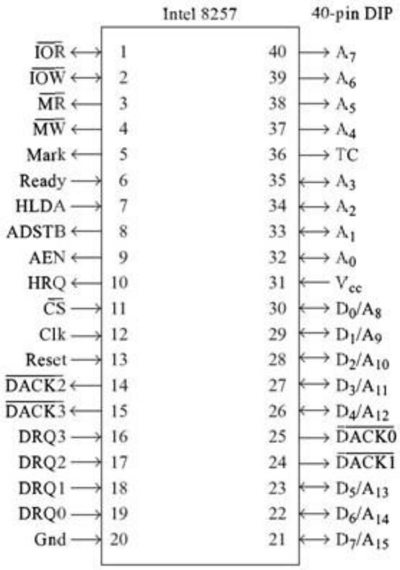
Following is the sequence of operations performed by a DMA −

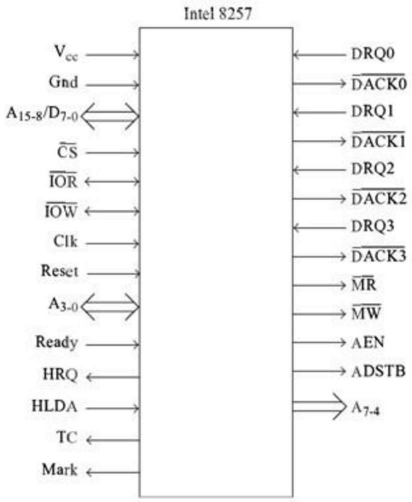
- Initially, when any device has to send data between the device and the memory, the device has to send DMA request (DRQ) to DMA controller.
- The DMA controller sends Hold request (HRQ) to the CPU and waits for the CPU to assert the HLDA.
- Then the microprocessor tri-states all the data bus, address bus, and control bus. The CPU leaves the control over bus and acknowledges the HOLD request through HLDA signal.
- Now the CPU is in HOLD state and the DMA controller has to manage the operations over buses between the CPU, memory, and I/O devices.

Answer to the Question no- 2 (b)

8257 Pin Description:

The following image shows the pin diagram of a 8257 DMA controller –





DRQ$_0$–DRQ3

These are the four individual channel DMA request inputs, which are used by the peripheral devices for using DMA services. When the fixed priority mode is selected, then DRQ$_0$ has the highest priority and DRQ$_3$ has the lowest priority among them.

DACK$_o$ – DACK$_3$

These are the active-low DMA acknowledge lines, which updates the requesting peripheral about the status of their request by the CPU. These lines can also act as strobe lines for the requesting devices.

D_0 – D_7

These are bidirectional, data lines which are used to interface the system bus with the internal data bus of DMA controller. In the Slave mode, it carries command words to 8257 and status word from 8257. In the master mode, these lines are used to send higher byte of the generated address to the latch. This address is further latched using ADSTB signal.

IOR

It is an active-low bidirectional tri-state input line, which is used by the CPU to read internal registers of 8257 in the Slave mode. In the master mode, it is used to read data from the peripheral devices during a memory write cycle.

IOW

It is an active low bi-direction tri-state line, which is used to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is used to load the data to the peripheral devices during DMA memory read cycle.

CLK

It is a clock frequency signal which is required for the internal operation of 8257.

RESET

This signal is used to RESET the DMA controller by disabling all the DMA channels.

A_0 - A_3

These are the four least significant address lines. In the slave mode, they act as an input, which selects one of the registers to be read or written. In the master mode, they are the four least significant memory address output lines generated by 8257.

CS

It is an active-low chip select line. In the Slave mode, it enables the read/write operations to/from 8257. In the master mode, it disables the read/write operations to/from 8257.

A_4 - A_7

These are the higher nibble of the lower byte address generated by DMA in the master mode.

READY

It is an active-high asynchronous input signal, which makes DMA ready by inserting wait states.

HRQ

This signal is used to receive the hold request signal from the output device. In the slave mode, it is connected with a DRQ input line 8257. In Master mode, it is connected with HOLD input of the CPU.

HLDA

It is the hold acknowledgement signal which indicates the DMA controller that the bus has been granted to the requesting peripheral by the CPU when it is set to 1.

MEMR

It is the low memory read signal, which is used to read the data from the addressed memory locations during DMA read cycles.

MEMW

It is the active-low three state signal which is used to write the data to the addressed memory location during DMA write operation.

ADST

This signal is used to convert the higher byte of the memory address generated by the DMA controller into the latches.

AEN

This signal is used to disable the address bus/data bus.

TC

It stands for 'Terminal Count', which indicates the present DMA cycle to the present peripheral devices.

MARK

The mark will be activated after each 128 cycles or integral multiples of it from the beginning. It indicates the current DMA cycle is the 128th cycle since the previous MARK output to the selected peripheral device.

V_{cc}

It is the power signal which is required for the operation of the circuit.

Architecture OF 8257 DMA Controller Chip:

The 8257 works in two modes- slave mode and the master mode. Similarly, the processor also works in two modes- active mode and the HOLD mode. The processor is normally in the active mode. In the active mode, the processor is the master of the computer system, including the 8257. Only when DMA transfer is required to be performed, the processor goes to the HOLD state and gives up control of the system bus. In such a state the processor is logically disconnected from the rest of the computer system, and the 8257 becomes the master for the rest of the computer system.

The 8257 is in the slave mode when the processor is programming the 8257, or when the processor is reading the contents of the internal registers of 8257. At this point of time the processor is in active mode, and is the master of the computer system, including the 8257.

The 8257 is in the master mode when the 8257 is actually controlling DMA data transfer. At this point of time the processor is in the HOLD state. Hence 8257 is the master of the computer system, excluding the processor, which is logically disconnected in the computer system.

Answer to the Question no- 3 (a)

There are various **Shift and Rotate instructions** present in the 8086 microprocessor. Let us discuss them one by one and understand their working:

1. **SHR : Shift Right**
2. **SAR : Shift Arithmetic Right**
3. **SHL : Shift Left**
4. **SAL : Shift Arithmetic Left**
5. **ROL : Rotate Left**
6. **ROR : Rotate Right**
7. **RCL : Rotate Carry Left**
8. **RCR : Rotate Carry Right**

SHR : Shift Right- The SHR instruction is an abbreviation for ‘Shift Right’. This instruction simply shifts the mentioned bits in the register to the right side one by one by inserting the same number (bits that are being shifted) of zeroes from the left end. The rightmost bit that is being shifted is stored in the Carry Flag (CF).

SAR : Shift Arithmetic Right- The SAR instruction stands for ‘Shift Arithmetic Right’. This instruction shifts the mentioned bits in the register to the right side one by one, but instead of inserting the zeroes from the left end, the MSB is restored. The rightmost bit that is being shifted is stored in the Carry Flag (CF).

SHL : Shift Left- The SHL instruction is an abbreviation for ‘Shift Left’. This instruction simply shifts the mentioned bits in the register to the left side one by one by inserting the same number (bits that are being shifted) of zeroes from the right end. The leftmost bit that is being shifted is stored in the Carry Flag (CF).

SAL : Shift Arithmetic Left- The SAL instruction is an abbreviation for ‘Shift Arithmetic Left’. This instruction is the same as SHL.

ROL : Rotate Left- The ROL instruction is an abbreviation for ‘Rotate Left’. This instruction rotates the mentioned bits in the register to the left side one by one such that leftmost bit that is being rotated is again stored as the rightmost bit in the register, and it is also stored in the Carry Flag.

ROR : Rotate Right- The ROR instruction stands for ‘Rotate Right’. This instruction rotates the mentioned bits in the register to the right side one by one such that rightmost bit that is being rotated is again stored as the MSB in the register, and it is also stored in the Carry Flag (CF).

RCL : Rotate Carry Left- This instruction rotates the mentioned bits in the register to the left side one by one such that leftmost bit that is being rotated it is stored in the Carry Flag (CF), and the bit in the CF moved as the LSB in the register.

RCR : Rotate Carry Right- This instruction rotates the mentioned bits in the register to the right side such that rightmost bit that is being rotated it is stored in the Carry Flag (CF), and the bit in the CF moved as the MSB in the register.

Program Execution Transfer Instructions (Branch and Loop Instructions)

These instructions are used to transfer/branch the instructions during an execution. It includes the following instructions −

Instructions to transfer the instruction during an execution without any condition −

- **CALL** − Used to call a procedure and save their return address to the stack.
- **RET** − Used to return from the procedure to the main program.
- **JMP** − Used to jump to the provided address to proceed to the next instruction.

Instructions to transfer the instruction during an execution with some conditions −

- **JA/JNBE** − Used to jump if above/not below/equal instruction satisfies.
- **JAE/JNB** − Used to jump if above/not below instruction satisfies.
- **JBE/JNA** − Used to jump if below/equal/ not above instruction satisfies.
- **JC** − Used to jump if carry flag CF = 1
- **JE/JZ** − Used to jump if equal/zero flag ZF = 1
- **JG/JNLE** − Used to jump if greater/not less than/equal instruction satisfies.
- **JGE/JNL** − Used to jump if greater than/equal/not less than instruction satisfies.
- **JL/JNGE** − Used to jump if less than/not greater than/equal instruction satisfies.
- **JLE/JNG** − Used to jump if less than/equal/if not greater than instruction satisfies.
- **JNC** − Used to jump if no carry flag (CF = 0)
- **JNE/JNZ** − Used to jump if not equal/zero flag ZF = 0
- **JNO** − Used to jump if no overflow flag OF = 0
- **JNP/JPO** − Used to jump if not parity/parity odd PF = 0
- **JNS** − Used to jump if not sign SF = 0
- **JO** − Used to jump if overflow flag OF = 1
- **JP/JPE** − Used to jump if parity/parity even PF = 1
- **JS** − Used to jump if sign flag SF = 1

Answer to the Question no- 3 (b)

Memory Interfacing

When we are executing any instruction, we need the microprocessor to access the memory for reading instruction codes and the data stored in the memory. For this, both the memory and the microprocessor require some signals to read from and write to registers.

I/O Interfacing

There are various communication devices like the keyboard, mouse, printer, etc. So, we need to interface the keyboard and other devices with the microprocessor by using latches and buffers. This type of interfacing is known as I/O interfacing.