**SET A**

University Roll No: -------------------------

**RAJASTHAN INSTITUTE OF ENGINEERING & TECHNOLOGY JAIPUR, RAJASTHAN**

**B. Tech. II Midterm EXAMINATION 2017-18**

**Semester and Branch**: III Year, VI Semester, EE

**Subject:** Advanced Microprocessor

***Duration*: Hours             *Maximum Marks*: 20**

**Student Instructions:**

* **All questions are compulsory.**
* **All questions carry equal marks.**
* **Use pencils for the diagrams.**

**Solution**

**Q.1 Explain the PIN Diagram of 8255A programmable peripheral Interface Chip.**

Ans:



**Data Bus Buffer**

It is a tri-state 8-bit buffer, which is used to interface the microprocessor to the system data bus. Data is transmitted or received by the buffer as per the instructions by the CPU. Control words and status information is also transferred using this bus.

**Read/Write Control Logic**

This block is responsible for controlling the internal/external transfer of data/control/status word. It accepts the input from the CPU address and control buses, and in turn issues command to both the control groups.

**CS**

It stands for Chip Select. A LOW on this input selects the chip and enables the communication between the 8255A and the CPU. It is connected to the decoded address, and A0 & A1 are connected to the microprocessor address lines.

Their result depends on the following conditions −

|  |  |  |  |
| --- | --- | --- | --- |
| CS | A1 | A0 | Result |
| 0 | 0 | 0 | PORT A |
| 0 | 0 | 1 | PORT B |
| 0 | 1 | 0 | PORT C |
| 0 | 1 | 1 | Control Register |
| 1 | X | X | No Selection |

**WR**

It stands for write. This control signal enables the write operation. When this signal goes low, the microprocessor writes into a selected I/O port or control register.

**RESET**

This is an active high signal. It clears the control register and sets all ports in the input mode.

**RD**

It stands for Read. This control signal enables the Read operation. When the signal is low, the microprocessor reads the data from the selected I/O port of the 8255.

**A0 and A1**

These input signals work with RD, WR, and one of the control signal. Following is the table showing their various signals with their result.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A1 | A0 | RD | WR | CS | Result |
| 0 | 0 | 0 | 1 | 0 | Input Operation  PORT A → Data Bus |
| 0 | 1 | 0 | 1 | 0 | PORT B → Data Bus |
| 1 | 0 | 0 | 1 | 0 | PORT C → Data Bus |
| 0 | 0 | 1 | 0 | 0 | Output Operation  Data Bus → PORT A |
| 0 | 1 | 1 | 0 | 0 | Data Bus → PORT A |
| 1 | 0 | 1 | 0 | 0 | Data Bus → PORT B |
| 1 | 1 | 1 | 0 | 0 | Data Bus → PORT D |

OR

**Q.1 Explain the Architecture of 8253A Timer Chip.**

Ans:

The Intel 8253 and 8254 are Programmable Interval Timers (PTIs) designed for microprocessors to perform timing and counting functions using three 16-bit registers. Each counter has 2 input pins, i.e. Clock & Gate, and 1 pin for “OUT” output. To operate a counter, a 16-bit count is loaded in its register. On command, it begins to decrement the count until it reaches 0, then it generates a pulse that can be used to interrupt the CPU.

Difference between 8253 and 8254

The following table differentiates the features of 8253 and 8254 −

8253/8254

* Its operating frequency is 0 - 2.6 MHz Its operating frequency is 0 - 10 MHz
* It uses N-MOS technology It uses H-MOS technology
* Read-Back command is not available
* Read-Back command is available
* Reads and writes of the same counter cannot be interleaved. Reads and writes of the same counter can be interleaved.

Features of 8253 / 54

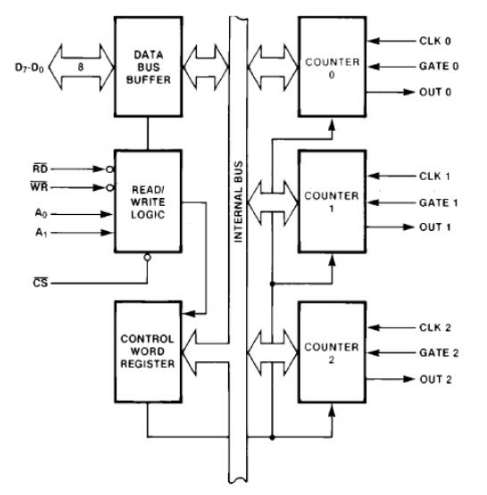
The most prominent features of 8253/54 are as follows −

* It has three independent 16-bit down counters.
* It can handle inputs from DC to 10 MHz.
* These three counters can be programmed for either binary or BCD count.
* It is compatible with almost all microprocessors.

8254 has a powerful command called READ BACK command, which allows the user to check the count value, the programmed mode, the current mode, and the current status of the counter.

8254 Architecture

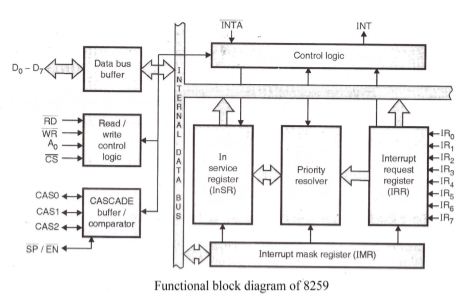
The architecture of 8254 looks as follows –



**Q.2 Explain the Architecture of 8259A Programmable Interrupt Controller Chip.**

**Ans:**  
Features:

1. It is a LSI chip which manages 8 levels of interrupts i.e. it is used to implement 8 level interrupt systems.
2. It can be cascaded in a master slave configuration to handle up to 64 levels of interrupts.
3. It can identify the interrupting device.
4. It can resolve the priority of interrupt requests i.e. it does not require any external priority resolver.
5. It can be operated in various priority modes such as fixed priority and rotating priority.
6. The interrupt requests are individually mask-able.
7. The operating modes and masks may be dynamically changed by the software at any time during execution of programs.
8. It accepts requests from the peripherals, determines priority of incoming request, checks whether the incoming request has a higher priority value than the level currently being serviced and issues an interrupt signal to the microprocessor.
9. It provides 8 bit vector number as an interrupt information.
10. It does not require clock signal.
11. It can be used in polled as well as interrupt modes.
12. The starting address of vector number is programmable.
13. It can be used in buffered mode.



It contains following blocks-

Data bus buffer-

* It is used to transfer data between microprocessor and internal bus.

Read/write logic-

* It sets the direction of data bus buffer.
* It controls all internal read/write operations.
* It contains initialization and operation command registers.

Cascaded buffer and comparator-

* In master mode, it functions as a cascaded buffer. The cascaded buffers outputs slave identification number on cascade lines.
* In slave mode, it functions as a comparator. The comparator reads slave identification number from cascade lines and compares this number with its internal identification number.
* In buffered mode, it generates an (EN) ̅ signal.

Control logic-

* It generates an INT signal. In response to an (INTA) ̅ signal, it releases three byte CALL address or one byte vector number.
* It controls read/write control logic, cascade buffer/comparator, in service register, priority resolver and IRR.

Interrupt request register-

* It is used to store all pending interrupt requests.
* Each bit of this register is set at the rising edge or at the high level of the corresponding interrupt request line.
* The microprocessor can read contents of this register by issuing appropriate command word.

In service register (InSR)-

* It is used to store all interrupt levels currently being serviced.
* Each bit of this register is set by priority resolver and reset by end of interrupt command word.
* The microprocessor can read contents of this register by issuing appropriate command word.

Priority resolver-

* It determines the priorities of the bit set in the IRR. To make decision, the priority resolver looks at the ISR.
* If the higher priority bit in the InSR is set then it ignores the new request.
* If the priority resolvers find that the new interrupt has a higher priority than the highest priority interrupt currently being serviced and the new interrupt is not in service, then it will set appropriate bit in the InSR and send the INT signal to the microprocessor for new interrupt request.

Interrupt mask register (IMR)-

* It is a programmable register.
* It is used to mask unwanted interrupt request by writing appropriate command word.
* The microprocessor can read contents of this register without issuing any command word.

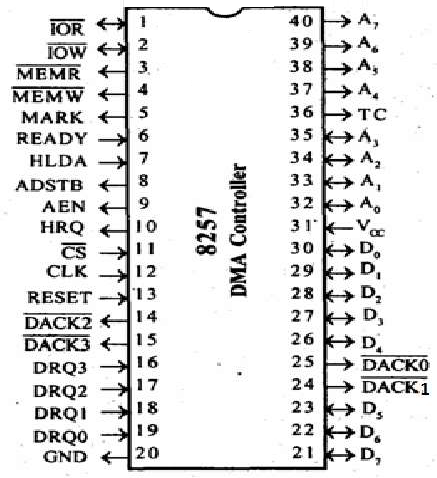
OR

**Q.2 What is the DMA? Explain the PIN Diagram of DMA Controller.**

### Ans:

### 8257 Pin Description

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



### DRQ0−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 DRQ0 has the highest priority and DRQ3 has the lowest priority among them.

### DACKo − DACK3

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.

### Do − D7

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.

### Ao - A3

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.

### A4 - A7

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.

### Vcc

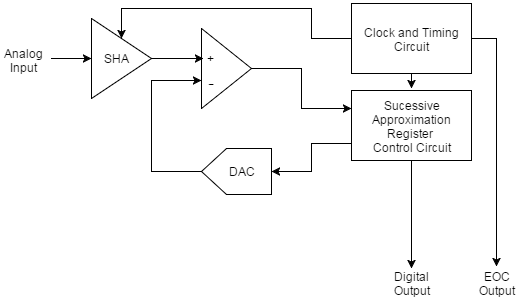
It is the power signal which is required for the operation of the circuit.

**Q.3 What is the means of Analog to Digital Converter? Explain the Successive Approximation method to convert the Analog data in the digital form.**

One of the most common analog-to-digital converters used in applications requiring a sampling rate under 10 MSPS is the Successive Approximation Register ADC. This ADC is ideal for applications requiring a resolution between 8-16 bits. For more information on resolution and sampling rates, please refer to the first in this series of articles: [Deciphering Resolution and Sampling Rate.](https://www.allaboutcircuits.com/technical-articles/understanding-analog-to-digital-converters-deciphering-resolution-and-sampl/) The SAR ADC is one of the most intuitive analog-to-digital converters to understand and once we know how this type of ADC works, it becomes apparent where its strengths and weaknesses lie.

Basic Operation of the SAR ADC

The basic successive approximation register analog-to-digital converter is shown in the schematic below:



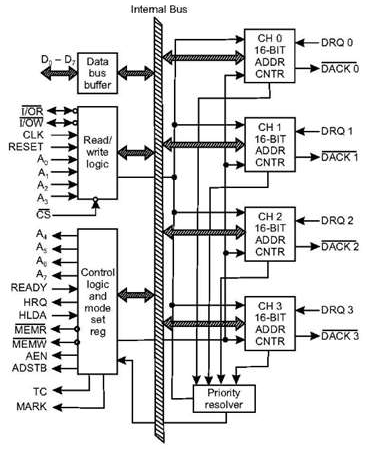
The SAR ADC does the following things for each sample:

1. The analog signal is sampled and held.
2. For each bit, the SAR logic outputs a binary code to the DAC that is dependent on the current bit under scrutiny and the previous bits already approximated. The comparator is used to determine the state of the current bit.
3. Once all bits have been approximated, the digital approximation is output at the end of the conversion (EOC).

The SAR operation is best explained as a binary search algorithm. Consider the code shown below. In this code, the current bit under scrutiny is set to 1. The resultant binary code from this is output to the DAC. This is compared to the analog input. If the result of the DAC output subtracted from the analog input is less than 0 the bit under scrutiny is set to 0.

OR

**Q.3 Explain the Architecture of DMA Controller.**



The functional block diagram of 8257 is shown in fig.

* The functional blocks of 8257 are data bus buffer, read/write logic, control logic, priority resolver and four numbers of DMA channels.
* Each channel has two programmable 16-bit registers named as address register and count register
* Address register is used to store the starting address of memory location for DMA data transfer.
* The address in the address register is automatically incremented after every read/write/verify transfer.
* The count register is used to count the number of byte or word transferred by DMA

Q**.4 Explain the Architecture of Keyboard/Display Controller 8279A Chip.**

Ans:

8279 programmable keyboard/display controller is designed by Intel that interfaces a keyboard with the CPU. The keyboard first scans the keyboard and identifies if any key has been pressed. It then sends their relative response of the pressed key to the CPU and vice-a-versa.

### How Many Ways the Keyboard is Interfaced with the CPU?

The Keyboard can be interfaced either in the interrupt or the polled mode. In the Interrupt mode, the processor is requested service only if any key is pressed, otherwise the CPU will continue with its main task.

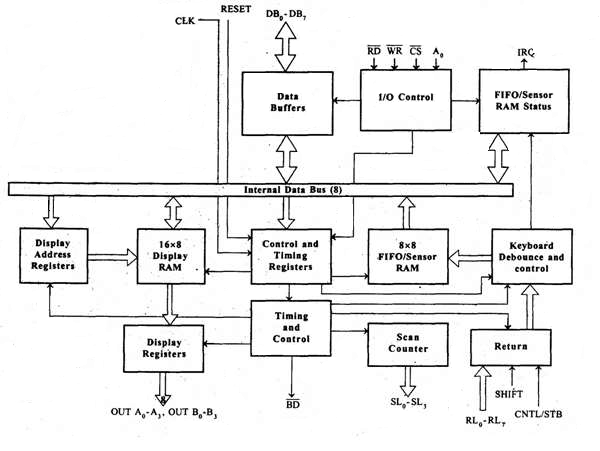
In the Polled mode, the CPU periodically reads an internal flag of 8279 to check whether any key is pressed or not with key pressure.

### How Does 8279 Keyboard Work?

The keyboard consists of maximum 64 keys, which are interfaced with the CPU by using the key-codes. These key-codes are de-bounced and stored in an 8-byte FIFORAM, which can be accessed by the CPU. If more than 8 characters are entered in the FIFO, then it means more than eight keys are pressed at a time. This is when the overrun status is set.

If a FIFO contains a valid key entry, then the CPU is interrupted in an interrupt mode else the CPU checks the status in polling to read the entry. Once the CPU reads a key entry, then FIFO is updated, and the key entry is pushed out of the FIFO to generate space for new entries.

### Architecture and Description



### I/O Control and Data Buffer

This unit controls the flow of data through the microprocessor. It is enabled only when D is low. Its data buffer interfaces the external bus of the system with the internal bus of the microprocessor. The pins A0, RD, and WR are used for command, status or data read/write operations.

### Control and Timing Register and Timing Control

This unit contains registers to store the keyboard, display modes, and other operations as programmed by the CPU. The timing and control unit handles the timings for the operation of the circuit.

### Scan Counter

It has two modes i.e. Encoded mode and Decoded mode. In the encoded mode, the counter provides the binary count that is to be externally decoded to provide the scan lines for the keyboard and display.

In the decoded scan mode, the counter internally decodes the least significant 2 bits and provides a decoded 1 out of 4 scan on SL0-SL3.

### Return Buffers, Keyboard Debounce, and Control

This unit first scans the key closure row-wise, if found then the keyboard debounce unit debounces the key entry. In case, the same key is detected, then the code of that key is directly transferred to the sensor RAM along with SHIFT & CONTROL key status.

### FIFO/Sensor RAM and Status Logic

This unit acts as 8-byte first-in-first-out (FIFO) RAM where the key code of every pressed key is entered into the RAM as per their sequence. The status logic generates an interrupt request after each FIFO read operation till the FIFO gets empty.

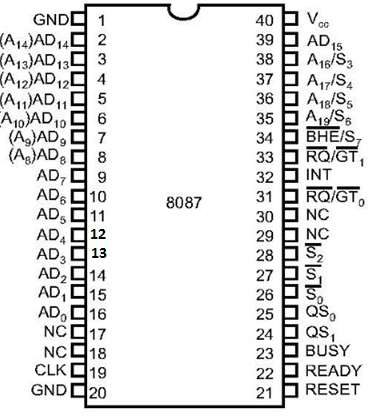
In the scanned sensor matrix mode, this unit acts as sensor RAM where its each row is loaded with the status of their corresponding row of sensors into the matrix. When the sensor changes its state, the IRQ line changes to high and interrupts the CPU.

### Display Address Registers and Display RAM

This unit consists of display address registers which holds the addresses of the word currently read/written by the CPU to/from the display RAM.

OR

**Q.4 Explain the PIN Diagram of 8087 Numeric Data Processor**



The following list provides the Pin Description of 8087 −

* **AD0 – AD15** − These are the time multiplexed address/data lines, which carry addresses during the first clock cycle and data from the second clock cycle onwards.
* **A19 / S6 – A16/S** − These lines are the time multiplexed address/status lines. It functions in a similar way to the corresponding pins of 8086. The S6, S4 and S3 are permanently high, while the S5 is permanently low.
* BHE¯¯¯¯¯¯¯¯¯¯¯¯BHE¯**/S7** − During the first clock cycle, the BHE¯¯¯¯¯¯¯¯¯¯¯¯BHE¯/S7 is used to enable data on to the higher byte of the 8086 data bus and after that works as status line S7.
* **QS1, QS0** − These are queue status input signals which provides the status of instruction queue, their conditions as shown in the following table −

|  |  |  |
| --- | --- | --- |
| **QS0** | **QS1** | **Status** |
| 0 | 0 | No operation |
| 0 | 1 | First byte of opcode from the queue |
| 1 | 0 | Empty the queue |
| 1 | 1 | Subsequent byte from the queue |

* **INT** − It is an interrupt signal, which changes to high when an unmasked exception has been received during the execution.
* **BUSY** − It is an output signal, when it is high it indicates a busy state to the CPU.
* **READY** − It is an input signal used to inform the coprocessor whether the bus is ready to receive data or not.
* **RESET** − It is an input signal used to reject the internal activities of the coprocessor and prepare it for further execution whenever required by the CPU.
* **CLK** − The CLK input provides the basic timings for the processor operation.
* **VCC** − It is a power supply signal, which requires +5V supply for the operation of the circuit.
* **S0, S1, S2** − These are the status signals that provide the status of the operation which is used by the Bus Controller 8087 to generate memory and I/O control signals. These signals are active during the fourth clock cycle.

|  |  |  |  |
| --- | --- | --- | --- |
| **S2** | **S1** | **S0** | **Queue Status** |
| 0 | X | X | Unused |
| 1 | 0 | 0 | Unused |
| 1 | 0 | 1 | Memory read |
| 1 | 1 | 0 | Memory write |
| 1 | 1 | 1 | Passive |

* **RQ/GT1 & RQ/GT0** − These are the **Request/Grant** signals used by the 8087 processors to gain control of the bus from the host processor 8086/8088 for operand transfers.