1. Draw and explain the block diagram of the programmable peripheral interface (8255A).

Ans.

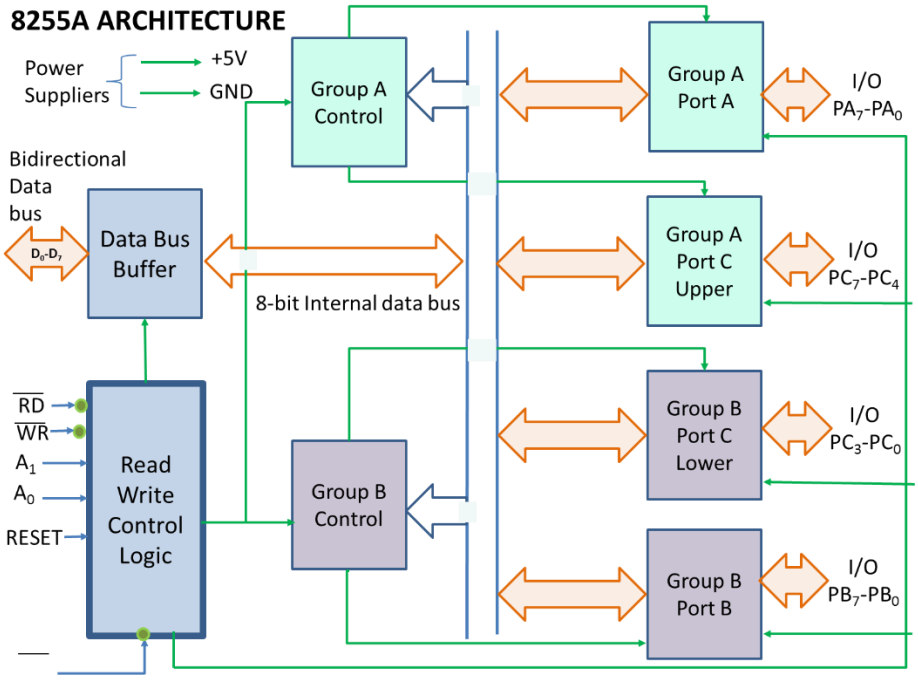


Figure: 8255A Architecture

Read Write Control Logic

RD (READ)	This is an active low signal that enables Read operation. When signal is low MPU reads data from selected I/O port of 8255A
WR (WRITE)	This is an active low signal that enables Write operation. When signal is low MPU writes data into selected I/O port or control register
RESET	This is an active high signal, used to reset the device. That means clear control registers
CS	This is Active Low signal. When it is low, then data is transfer from 8085 CS signal is the master Chip Select. A_0 and A_1 specify one of the I/O ports or control register

CS	A1	A0	Selected
0	0	0	PORT A
0	0	1	PORT B
0	1	0	PORT C
0	1	1	Control Register
1	X	X	8255A is not selected

Data Bus Buffer

- This three-state bi-directional 8-bit buffer is used to interface the 8255 to the system data bus.
- Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU.
- Control words and status information are also transferred through the data bus buffer.

Group A and Group B Controls

- The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255.
- The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255.
- Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Ports A, B, and C

- The 8255 contains three 8-bit ports (A, B, and C).
- All can be configured to a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255.
- Port A One 8-bit data output latch/buffer and one 8-bit data input latch.
- Both "pull-up" and "pull-down" bus-hold devices are present on Port A.
- Port B One 8-bit data input/output latch/buffer and one 8-bit data input buffer.
- Port C One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control.
- Each 4-bit port contains a 4-bit latch and it can be used for the control signal output and status signal inputs in conjunction with ports A and B.

2. Explain 8255A I/O Operating Modes

Ans. 8255A has three different I/O operating modes:

1. Mode 0
2. Mode 1
3. Mode 2

Mode 0

- Simple I/O for port A,B and C
- In this mode, Port A and B is used as two 8-bit ports and Port C as two 4-bit ports.
- Each port can be programmed in either input mode or output mode where outputs are latched and inputs are not latched.
- Ports do not have handshake or interrupt capability.

Mode 1: Input or Output with Handshake

- Handshake signal are exchanged between MPU and peripheral prior to data transfer.
- In this mode, Port A and B is used as 8-bit I/O ports.
- Mode 1 is a handshake Mode whereby ports A and/or B use bits from port C as handshake signals.
- In the handshake mode, two types of I/O data transfer can be implemented: status check and interrupt.
- Port A uses upper 3 signals of Port C: PC3, PC4, PC5
- Port B uses lower 3 signals of Port C : PC0, PC1, PC2
- PC6 and PC7 are general purpose I/O pins

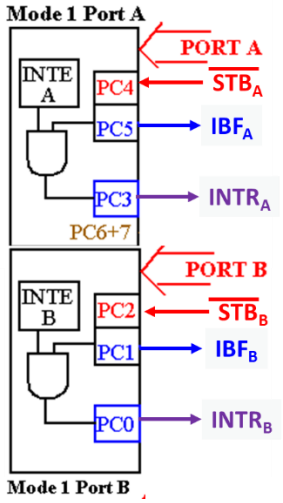


Figure: Mode1 Input Handshake

STB (Strobe Input):

- This active low signal is generated by a peripheral device to indicate that, it has transmitted a byte of data. The 8255A, in response to **STB**, generates **IBF** and **INTR**.

IBF (Input Buffer Full)

This signal is acknowledged by 8255A to indicate that the input latch has received the data byte. It will get reset when the MPU reads the data.

INTR(Interrupt Request)

This is an output signal that may be used to interrupt the MPU. This signal is generated if STB, IBF and INTE (internal flip-flop) are all at logic 1. It will get reset by the falling edge of RD

INTE(Interrupt Enable)

- This signal is an internal flip-flop, used to enable or disable the generation of INTR signal.
- The interrupt enable signal is neither an input nor an output; it is an internal bit programmed via the PC4 (port A) or PC2 (port B) bits.

Mode 2

- In this mode, Port A can be configured as the bidirectional port and Port B either in Mode 0 or Mode 1.
- Port A uses five signals from Port C as handshake signals for data transfer.
- The remaining three signals from Port C can be used either as simple I/O or as handshake for port B.

3. Explain BSR Mode of the programmable peripheral interface (8255A) with necessary diagrams.

Ans.

- These are two basic modes of operation of 8255.
  I/O mode and Bit Set-Reset mode (BSR).
- In I/O mode, the 8255 ports work as programmable I/O ports, while in BSR mode only port C (PC0-PC7) can be used to set or reset its individual port bits.
- Under the I/O mode of operation, further there are three modes of operation of 8255, so as to support different types of applications, mode 0, mode 1 and mode 2.

8255A: BSR(Bit Set/Reset) Mode

- In this mode any of the 8-bits of port C can be set or reset depending on D_0 of the control word.
- The bit to be set or reset is selected by bit select flags D_3, D_2 and D_1 of the CWR (Control Word Register).
- BSR Control Word affects one bit at a time
- It does not affect the I/O mode

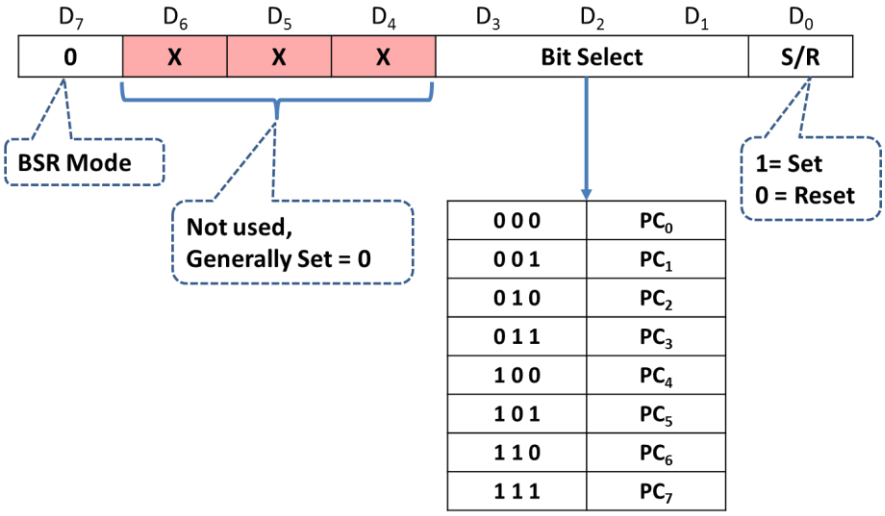


Figure: BSR Mode Control Word

4. Explain 8255A Control Word and Control Register with necessary diagram.

Ans. Control Register

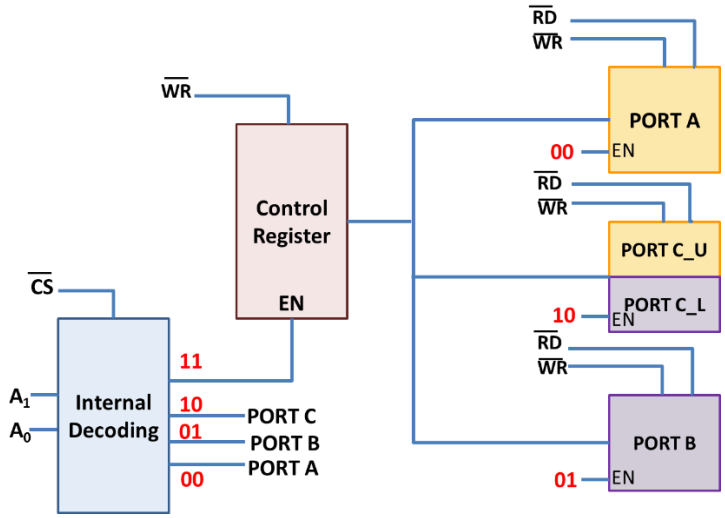


Figure: Control Register 8255A

Control Word: Content of Control register is known as Control Word.

- Control word specify an I/O function for each port this register can be.

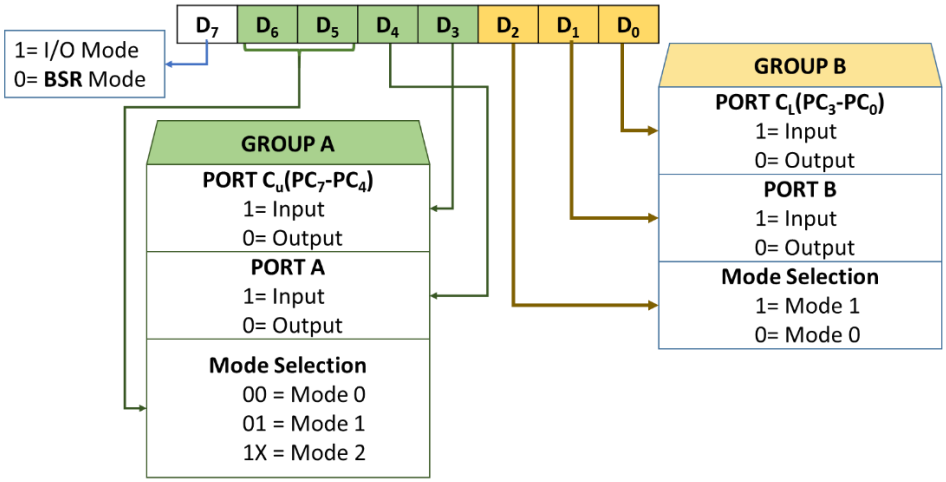


Figure:8255A Control Word

- Accessed to write a control word when A0 and A1 are at logic1, the register is not accessible for a read operation.
- Bit D7 of the control register either specifies the I/O function or the bit Set/Reset function, as classified in figure 1.
- If bit D7=0, bits D6-D0 determine I/O function in various mode, as shown in figure 4.
- If bit D7=0 port C operates in the bit Set/Reset (BSR) mode.
- The BSR control word does not affect the function of port A and B.

5. What is the need of the programmable interrupt controller (8259A)? Draw and explain the block diagram of 8259A.

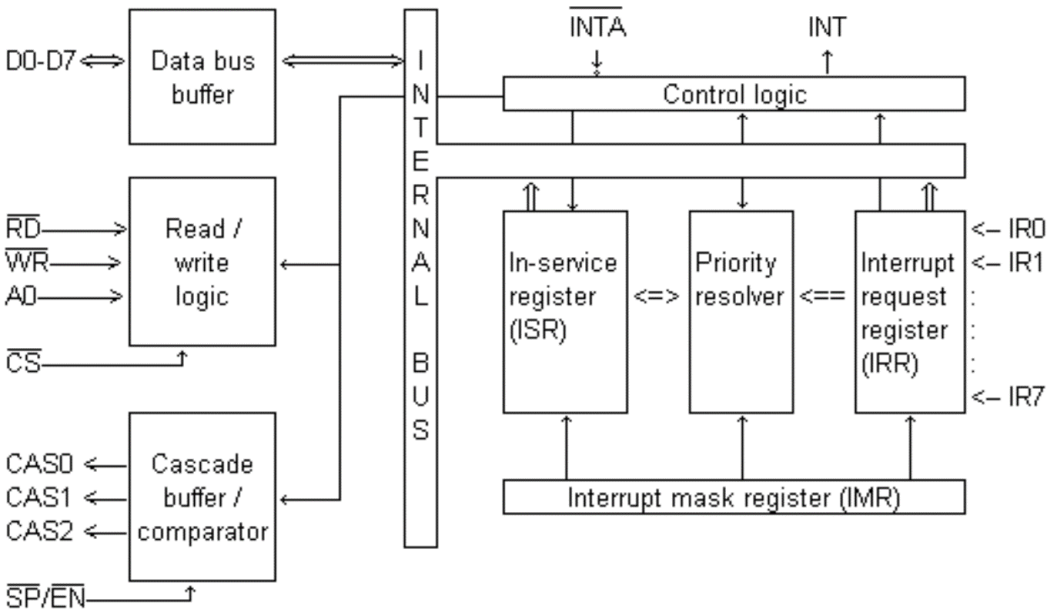
Ans.

- The Intel 8259 is a Programmable Interrupt Controller (PIC) designed for use with the 8085 and 8086 microprocessors.
- The 8259 can be used for applications that use more than five numbers of interrupts from multiple sources.

The main features of 8259 are listed below

- Manage eight levels of interrupts.
- Eight interrupts are spaced at the interval of four or eight locations.
- Resolve eight levels of priority in fully nested mode, automatic rotation mode or specific rotation mode.
- Mask each interrupt individually.
- Read the status of pending interrupt, in-service interrupt, and masked interrupt.
- Accept either the level triggered or edge triggered interrupt

8259 Internal Block Diagram



Read/Write Logic

- It is typical R/W logic.
- When address line A0 is at logic 0, the controller is selected to write a command word or read status.
- The Chip Select logic and A0 determine the port address of controller.

Control Logic

- It has two pins: INT as output and INTA as input.
- The INT is connected to INTR pin of MPU

Interrupt Registers and Priority Resolver

1. Interrupt Request Register (IRR)
2. Interrupt In-Service Register (ISR)
3. Priority Resolver
4. Interrupt Mask Register (IMR)

Interrupt Request Register (IRR) and Interrupt In-Service Register (ISR)

- Interrupt input lines are handled by two registers in cascade – IRR and ISR
- IRR is used to store all interrupt which are requesting service.
- ISR is used to store all interrupts which are being serviced.

Priority Resolver

- This logic block determines the priorities of the bit set in IRR.
- IR_0 is having highest priority, IR_7 is having lowest priority

Interrupt Mask Register

- It stores bits which mask the interrupt lines to be masked
- IMR operates on the IRR.
- Masking of high priority input will not affect the interrupt request lines.

Cascade Buffer / Comparator

This block is used to expand the number of interrupt levels by cascading two or more 8259As.

6. State the difference between the vectored and non-vectored interrupts. Explain vectored interrupts of the 8085 microprocessor.

Ans. Difference between the vectored and non-vectored interrupts

VECTORED INTERRUPT

- In vectored interrupts, the processor automatically branches to the specific address in response to an interrupt.
- In vectored interrupts, the manufacturer fixes the address of the ISR to which the program control is to be transferred.
- The TRAP, RST 7.5, RST 6.5 and RST 5.5 are vectored interrupts.
- TRAP is the only non-maskable interrupt in the 8085.

NON-VECTORED INTERRUPT

- In non-vectored interrupts the interrupted device should give the address of the interrupt service routine (ISR).
- The INTR is a non-vectored interrupt.
- Hence when a device interrupts through INTR, it has to supply the address of ISR after receiving interrupt acknowledge signal.

Interrupt	Maskable	Vectored
INTR	Yes	No
RST 5.5	Yes	Yes
RST 6.5	Yes	Yes
RST 7.5	Yes	Yes
TRAP	No	Yes

Explain vectored interrupts of the 8085 microprocessor

The vector addresses of 8085 interrupts are given below:

Software Interrupt

RST 0	0000H
RST 1	0008H
RST 2	0010H
RST 3	0018H
RST 4	0020H
RST 5	0028H
RST 6	0030H
RST 7	0038H

Hardware Interrupt

RST 7.5	003CH
RST 6.5	0034H
RST 5.5	002CH
TRAP	0024H

Software Interrupt

- The software interrupts of 8085 are RST 0, RST 1, RST 2, RST 3, RST 4, RST 5, RST 6 and RST 7.
- The software interrupts cannot be masked and they cannot be disabled.

Hardware Interrupt

- The vectored hardware interrupts of 8085 are TRAP, RST 7.5, RST 6.5, RST 5.5.
- An external device, initiates the hardware interrupts of 8O85 by placing an appropriate signal at the interrupt pin of the processor.
- The processor keeps on checking the interrupt pins at the second T -state of last machine cycle of every instruction.
- If the processor finds a valid interrupt signal and if the interrupt is unmasked and enabled, then the processor accepts the interrupt.
- The acceptance of the interrupt is acknowledged by sending an INTA signal to the interrupted device.
- The processor saves the content of PC (program Counter) in stack and then loads the vector address of the interrupt in PC. (If the interrupt is non-vectored, then the interrupting device has to supply the address of ISR when it receives INTA signal).
- It starts executing ISR in this address.
- At the end of ISR, a return instruction, RET will be placed.
- When the processor executes the RET instruction, it POP the content of top of stack to PC.
- Thus the processor control returns to main program after servicing interrupt.

7. Explain Interfacing Seven-Segment LEDs as an Output

Ans.

- Interface the 8085 Microprocessor System with seven segment display through its programmable I/O port 8255.
- Seven segment displays is often used in the digital electronic equipment to display information regarding certain process.
- I/O devices (or peripherals) such as LEDs and keyboards are essential components of the microprocessor-based or microcontroller-based systems.
- Seven-segment LEDs Often used to display BCD numbers (1 through 9) and a few alphabets.
- A group of eight LEDs physically mounted in the shape of the number eight plus a decimal point.
- Each LED is called a segment and labeled as 'a' through 'g'.

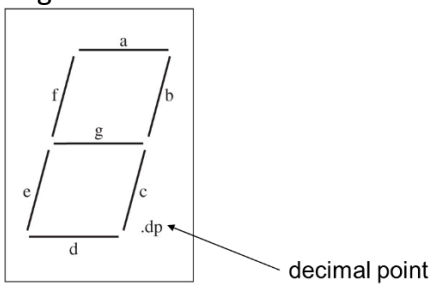
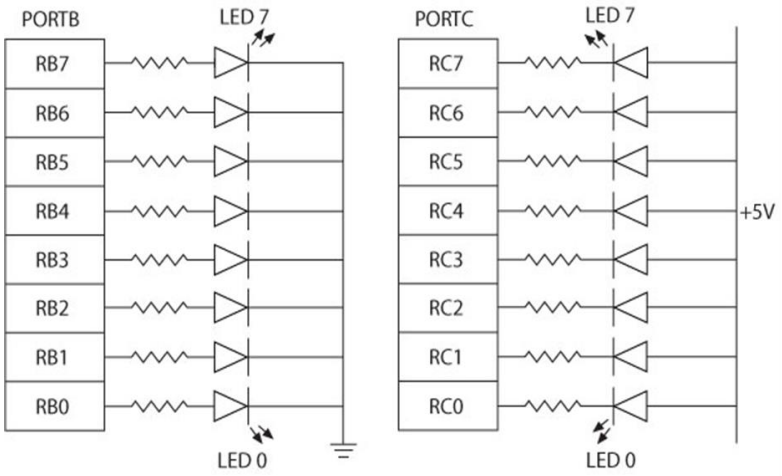


Figure: Seven Segment LED

- Commonly used output peripherals in embedded systems are LEDs, seven-segment LEDs, and LCDs; the simplest is LED

Two ways of connecting LEDs to I/O ports:

1. LED cathodes are grounded and logic 1 from the I/O port turns on the LEDs - The current is supplied by the I/O port called current sourcing.
2. LED anodes are connected to the power supply and logic 0 from the I/O port turns on the LEDs - The current is received by the chip called current sinking.



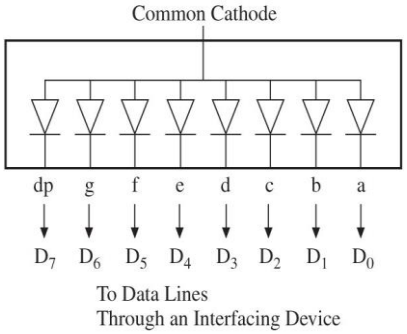
Common Cathode — Active high

Common Anode — Active low

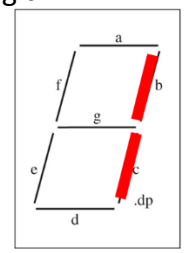
- In a common anode seven-segment LED All anodes are connected together to a power supply and cathodes are connected to data lines
- Logic 0 turns on a segment.

Example:

To display digit 1, so all segments except b and c should be off.



Byte 11111001 = F9H will display digit 1.



8. Explain I/O interfacing Methods

Ans. There are two method of interfacing memory or I/O devices with the microprocessor are as follows:

1) I/O mapped I/O

2) Memory mapped I/O

1) I/O MAPPED I/O

- In this technique, I/O device is treated as an I/O device and memory as memory. Each I/O device uses eight address lines.
- If eight address lines are used to interface to generate the address of the I/O port, then 256 Input/output devices can be interfaced with the microprocessor.
- The 8085 microprocessor has 16 bit address bus, so we can either use lower order address lines (A0 – A7) or higher order address lines(A8 – A15) to address I/O devices. We used lower order address bus & address available on A0 – A7 will be copied on the address lines A8 – A15.
- In I/O mapped I/O, the complete 64 Kbytes of memory can be used to address memory locations separately as the address space is not shared with I/O devices.
- In this interface type, the data transfer is possible between accumulator (A) and I/O devices only. Arithmetic and logical operation are not possible directly.
- As 8 bit device address used, Address decoding is simple so less hardware is required.
- The separate control signals are used to access I/O devices and memory such as IOR, IOW for I/O port and MEMR, MEMW for memory hence memory location are protected from the I/O access.

2) MEMORY MAPPED I/O

- In this technique, I/O devices are treated as memory and memory as memory, hence the address of the I/O devices are as same as that of memory i.e. 16 bit for 8085 microprocessor.
- So, the address space of the memory i.e. 64 Kbytes will be shared by the I/O devices as well as by memory. All 16 address lines i.e. A0-A15 is used to address memory locations as well as I/O devices.
- The control signals MEMR and MEMW are used to access memory devices as well as I/O devices.

Comparison of Memory-Mapped I/O and Peripheral Mapped I/O

No	Characteristics	Memory mapped I/O	I/O mapped I/O
1	Device Address	16 bit	8 Bit
2	Control signals for inputs	MEMR & MEMW	IOR & IOW
3	Instruction Available	All memory related instruction : LDA; STA; LDAX; STAX; MOV M,R; ADD M; SUB M	IN and OUT instructions only
4	Data Transfer	Between any register and I/O devices.	Between I/O device and Accumulator only.
5	Maximum Numbers of I/Os Possible	Memory Map (64K) is shared between I/Os and System memory.	I/O Mapped is independent of memory map; 256 Input and 256 output devices can be connected.
6	Execution Speed	13 T-State (LDA, STA, ..) 7 T-State (MOV M,R)	10 T-State
7	Hardware Requirement	More hardware is needed to decode 16 bit address	Less hardware is needed to decode 8 bit address
8	Other Feature	Arithmetic and logical operations are directly performed with I/O devices.	Not available