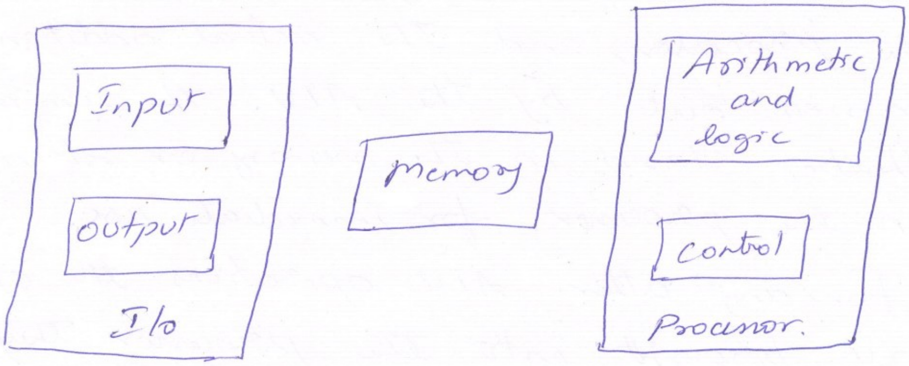
Q.2 a. Explain basic functional units of a computer with the help of neat block diagram. (8)

Answer:

A computer consists of five functionally independent main parts: (1) input (2) memory (3) Arithmetic and logic (4) output (5) control units. as shown in the diagram.



Input:- This unit accepts coded information from human operators, from electromechanical devices such as keyboards or from other computers over digital communication lines.

Memory Unit:- The function of the memory unit is to store programs and data. There are two classes of storage. called primary and Secondary.

Primary storage is a fast memory that operates at electronic speeds.

Secondary storage is used when large amounts of data and many programs have to be stored, particularly for information that is accessed infrequently.

Arithmetic and Logical unit :- most of the computer operations are executed in the ALU of the procesor.

Suppose two numbers located in the memory are to be added. They are brought into the processor, and the actual additon is carried out by the ALU. The sum may Thenbe stored in the memory or retained in the procesor for immediate use.

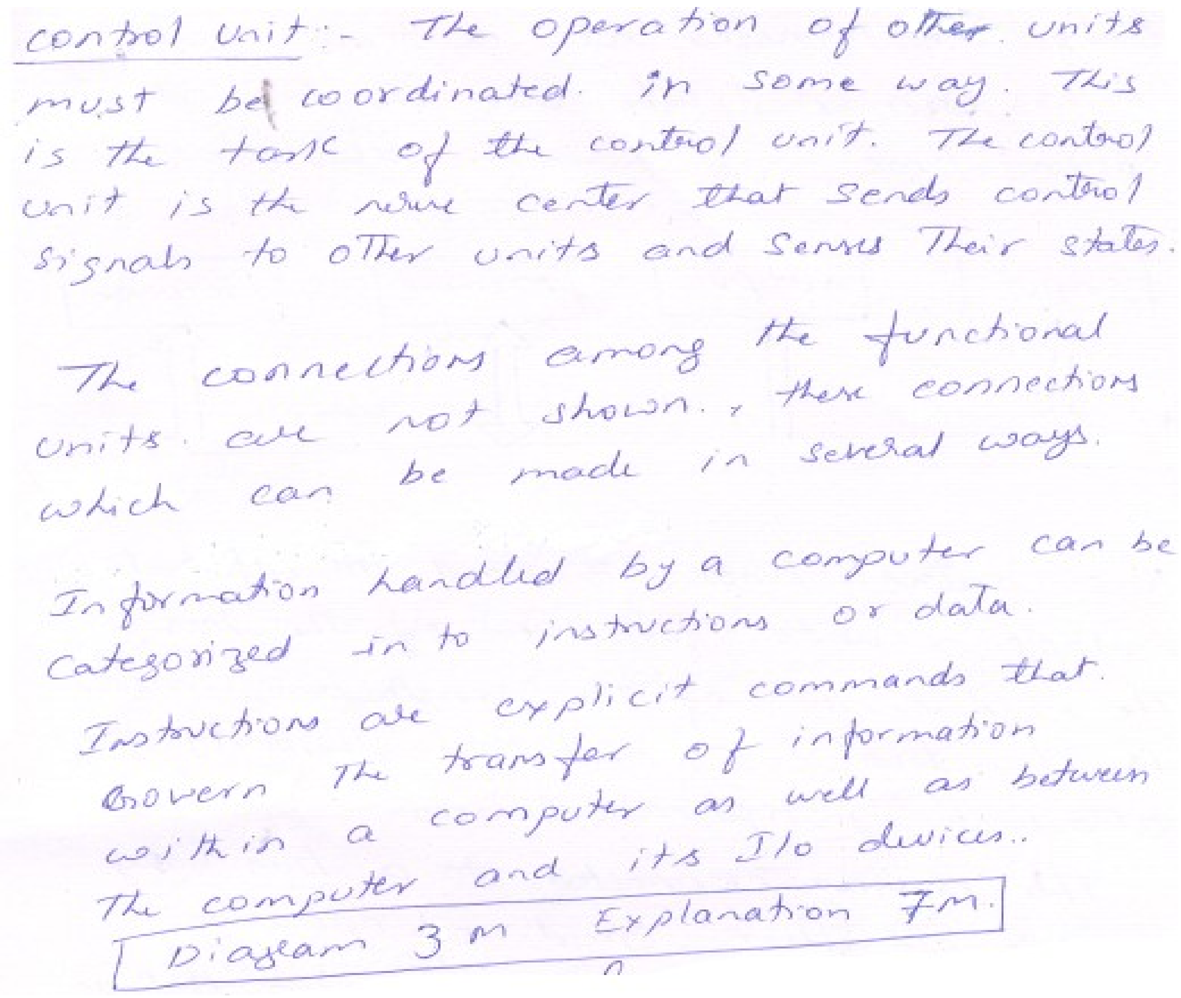
For any other ALU operations the operands are brought into the processor, they are stored in high-speed storage elements called registers.

The control and arithmetic and logic units are many times faster than other devices connected to a computer system. This enables a single procesor to control a number of external devices such as keyboards, displays, magnetic and optical disks.

Output unit :- This unit is the counter part of input unit. Its function is to send procesed results to the outside world.

Example :- printer, plotter.

Some units, such as graphics displays, provide both an output function and an input function.



control unit:- The operation of other units must be coordinated in some way. This is the task of the control unit. The control unit is the nerve center that Sends control signals to other units and Senses Their states.

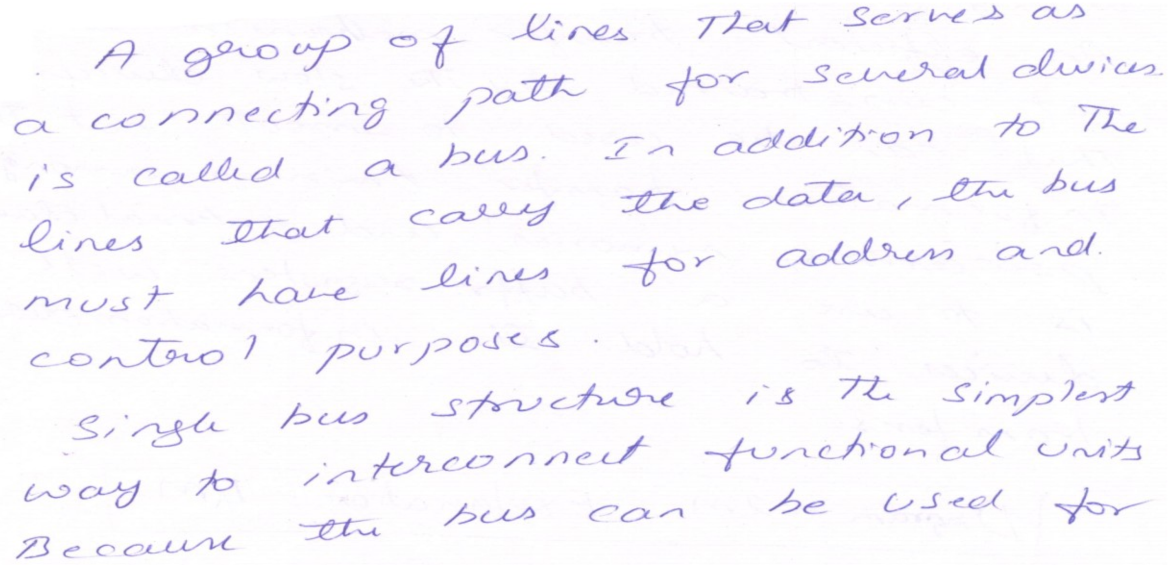
The connections among the functional units. are not shown., these connections which can be made in several ways.

Information handled by a computer can be categorized in to instructions or data. Instructions are explicit commands that Govern The transfer of information within a computer as well as between The computer and its I/o devices..

Diagram 3 M Explanation 7M.

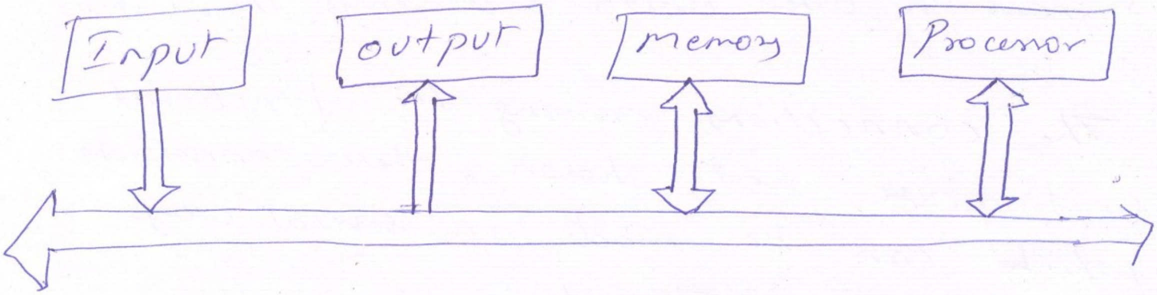
b. What is bus? Explain single bus structure with the help of neat diagram. (8)

Answer:



A group of lines That serves as a connecting path for several devices is called a bus. In addition to The lines That carry the data, the bus must have lines for address and control purposes.

Single bus structure is the simplest way to interconnect functional units Because the bus can be used for only one transfer at a time, only two units can actively use the bus at any given time. as shown in the diagram.



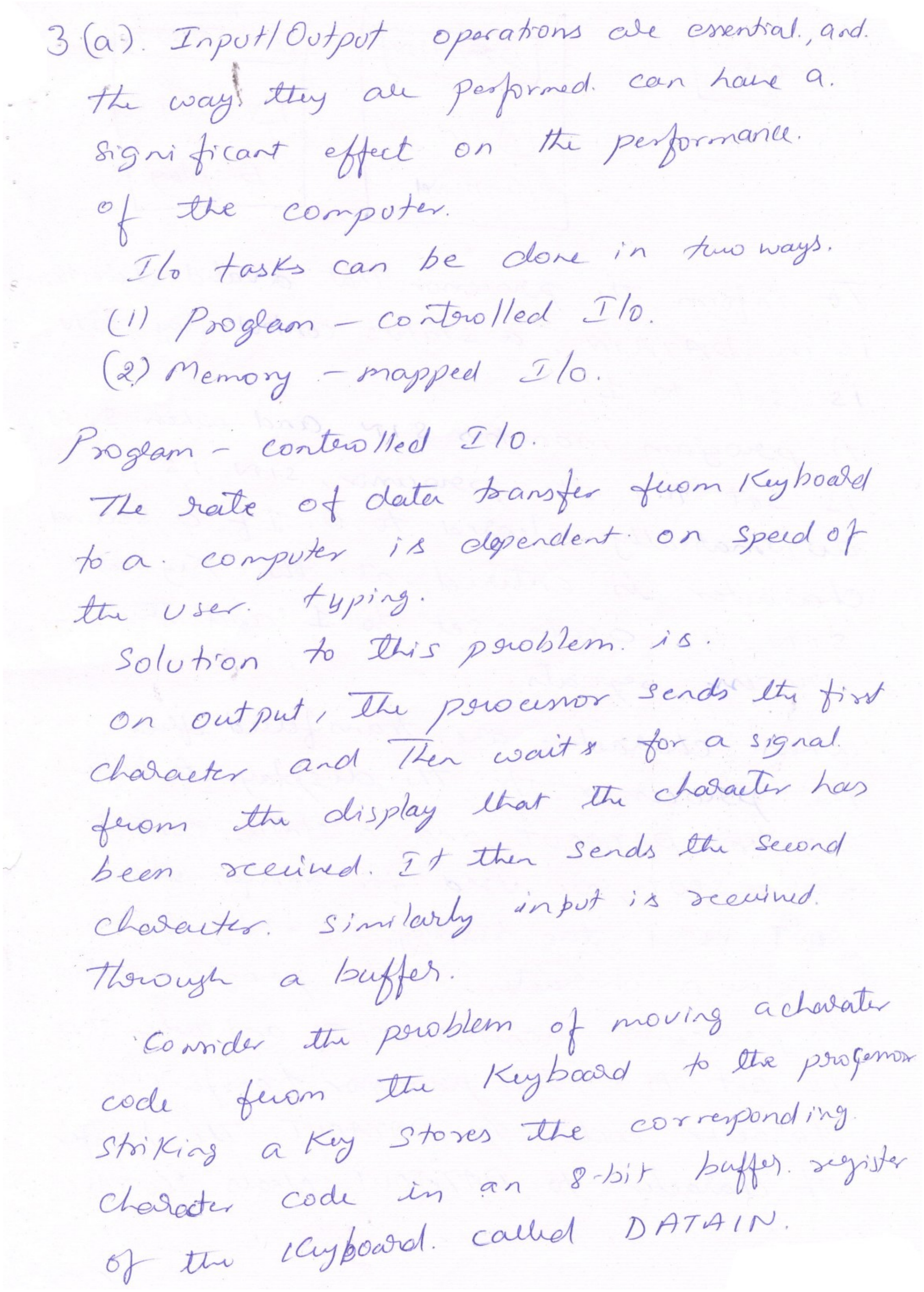
Bus control lines are used to albitrate multiple requests for use of the bus. The main virtue of the single bus structure is its low cost and its flexibility for attaching peripheral devices.

The devices connected to a bus vary widely in Their speed of operation. An efficient transfer mechanism that is not constrained by the slow devices. and That can be used to smooth out the information transfer timings among processors, memories, and external devices. is to use a buffer registers with the devices to hold. the information during transfers.

Diagram – 2M Explanation. – 4M.

Q.3 a. Explain how data transfer is achieved between memory of a computer and the outside world. (10)

Answer:



3 (a). Input/Output operations are essential, and the way they are performed can have a significant effect on the performance of the computer.

I/o tasks can be done in two ways.

(1) Program - controlled I/o.

(2) Memory - mapped I/o.

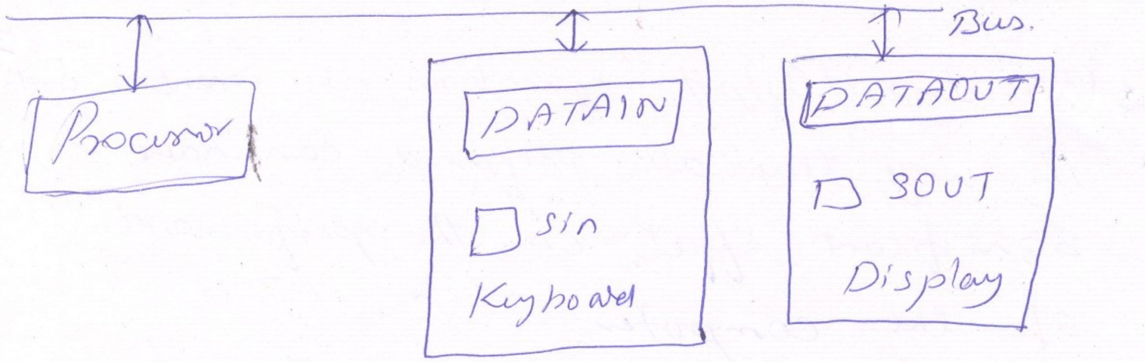
Program - controlled I/o.

The rate of data transfer from keyboard to a computer is dependent on speed of the user typing.

Solution to this problem is.

On output, The processor sends the first character and Then waits for a signal from the display that the character has been received. It then sends the second character. similarly input is received through a buffer.

Consider the problem of moving a character code from the keyboard to the processor. Striking a key stores the corresponding character code in an 8-bit buffer register of the keyboard called DATAIN.



To inform the processor that a valid character is in DATAIN, a status control flag, SIN is set to 1.

A program monitors SIN and when SIN is set to 1 the processor, SIN is automatically cleared to 0. If a second character is entered at the Keyboard. character is again set to 1 and the SIN is again set to 1 and the process repeats.

when characters are transfered from. The processor to the display. A buffer register. DATAOUT, and a status control flag, SOUT, are used for transfer. when SOUT is 1, the display is ready to receive a character. under program control the processor monitors SOUT, and when SOUT is set to 1, the processor transfers a character code to DATAOUT. The transfer of character to DATAOUT clears SOUT to 0.

memory-mapped I/o

many computers use this allangement in which some memory addless values are used to sefer to peripheral device buffer segisters, such as DATAIN and DATAOUT. No special instructions are needed. to access the contents of these registers. data can be transferred between these registers and the processor using. Move, store, & Load instructions

Example: contents of the Keyboard character buffer DATAIN can be transferred to register R1 in the processor by the instruction

MoveByte DATAIN, R1

contents of register R1 can be transferred to DATAOUT by the instructions.

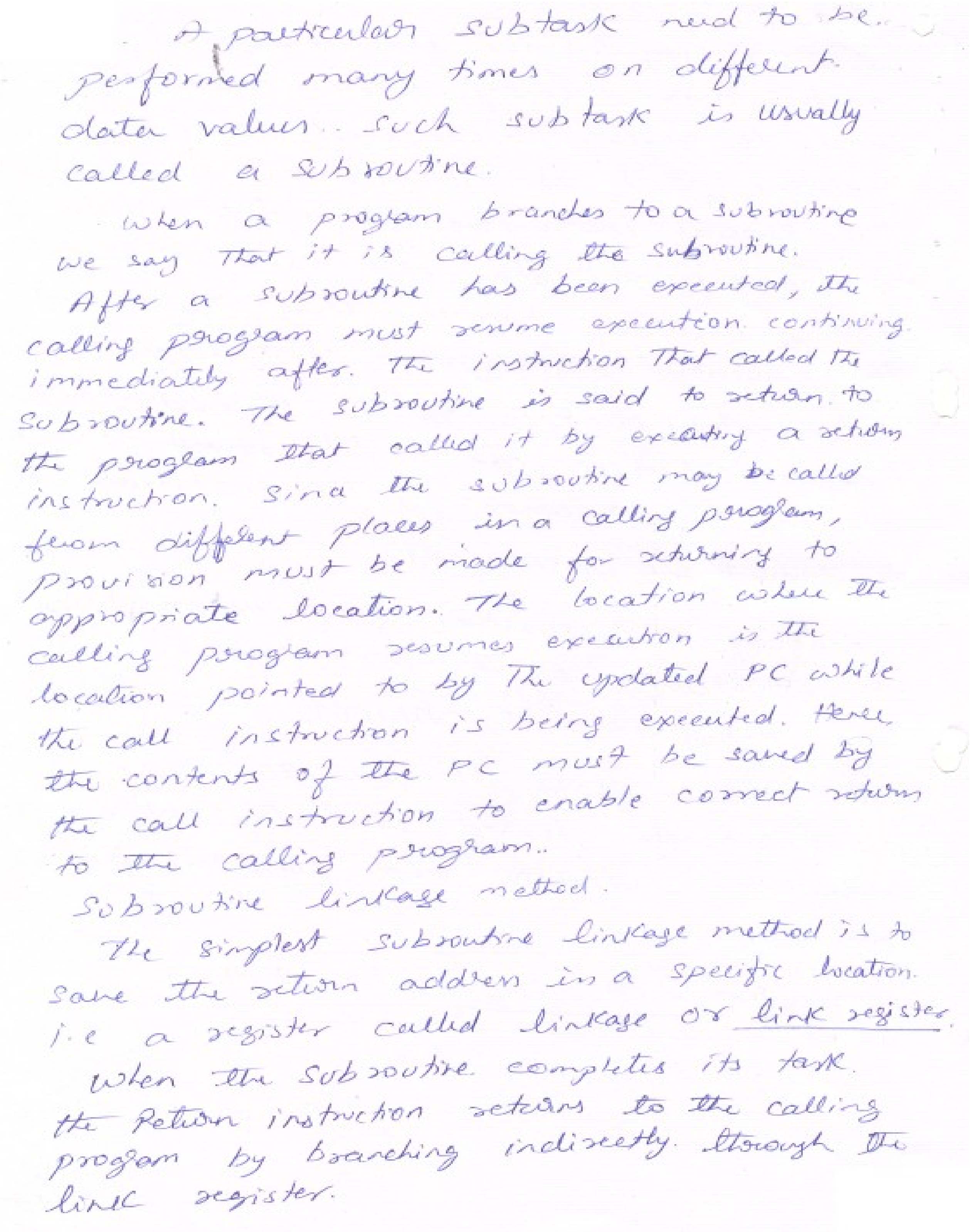
MoveByte R1, DATAOUT.

The status flags SIN and SOUT are automatically cleared when the buffer registers DATAIN and DATAOUT are referenced.

diagram 2M Explanation 8M.

b. Define subroutine. Explain linking of subroutines using link register. (6)

Answer:



A particular subtask need to be performed many times on different data values.. such subtask is usually called a subroutine.

When a program branches to a subroutine we say that it is calling the subroutine. After a subroutine has been executed, the calling program must resume execution continuing immediatly after. The instruction That called the subroutine. The subroutine is said to return to the program that called it by executing a return instruction. Since the subroutine may be called from different places in a calling program, provision must be made for returning to the appropriate location. The location where the calling program resumes execution is the location pointed to by The updated PC while the call instruction is being executed. Hence the contents of the PC must be saved by the call instruction to enable correct return to the calling program..

Subroutine linkage method.

The simplest subroutine linkage method is to save the return address in a specific location. i.e a register called linkage or link register.

When the subroutine completes its task. the Return instruction returns to the calling program by branching indirectly through the link register.

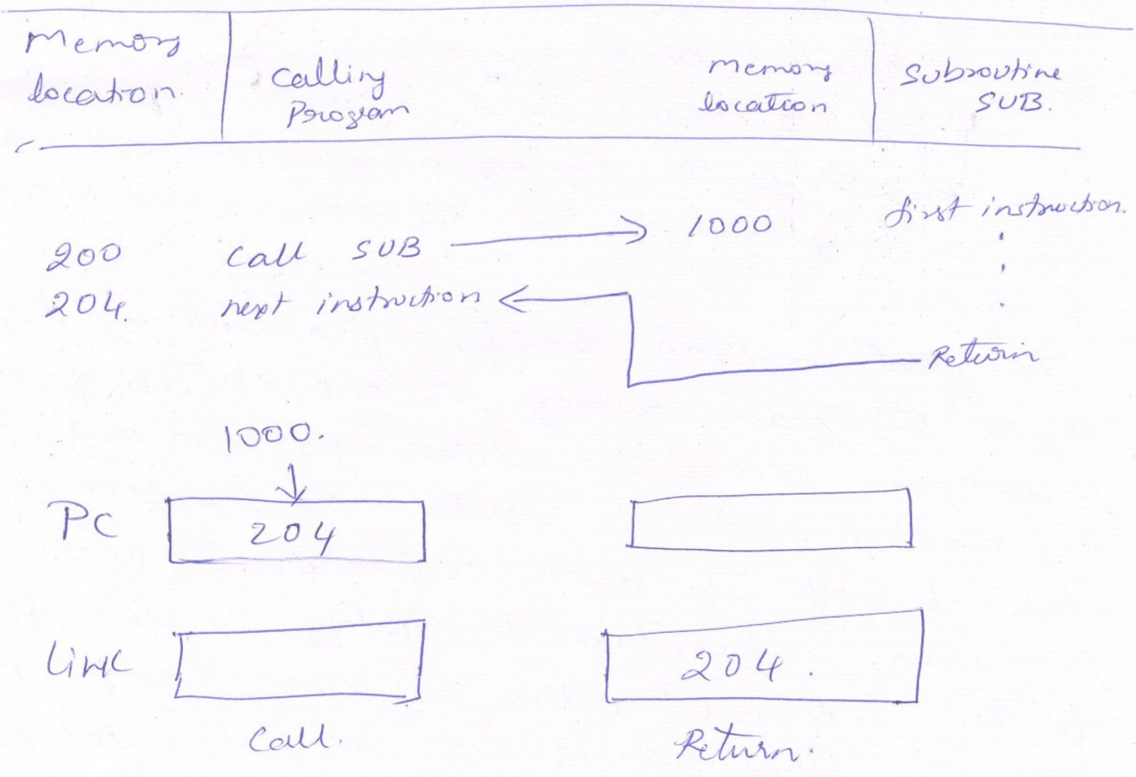
The call instruction is just a special branch instruction that performs the following opeations.

* store the contents of the PC in the link register.

* Branch to the target address specified by the instruction.

The return instruction is a special instruction that performs the operation:

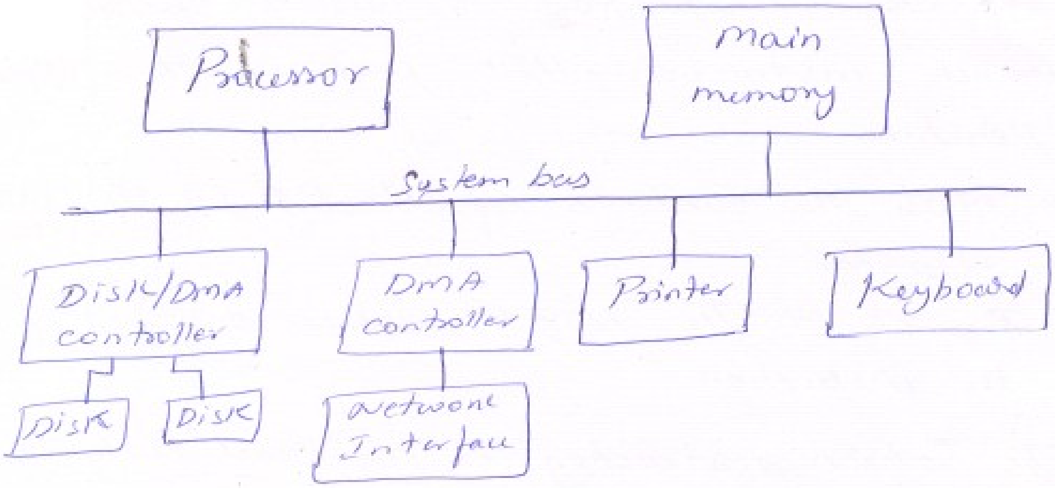
* Branch to the address contained in the link register.



Example 2M Explanation 4M.

Q.4 a. Explain the use of DMA controllers in a computer system. (8)

Answer:



A DMA controller connects a high-speed network to the computer bus. The disk controller which controls 2 disks, also has DMA capability and provides two DMA channels. It can perform two independent DMA operations, as if each disk had its own DMA controller. The registers need to store the memory address, the word count, and are duplicated, so that one can be used with each device.

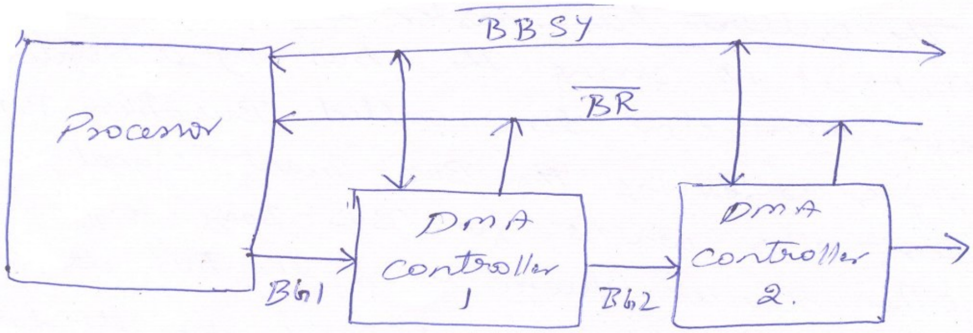
To start a DMA transfer of a block of data from the main memory to one of the disks a program writes the address and word count information into the registers of the corresponding channel of the disk controller. It also provides information to identify the data for further retrieval. when the DMA transfer is completed. done bit is set. and if IE bit is set the controller sends an interrupt signal to the processor and sets the IRQ bit.

status bit is used to record information regarding whether the transfer took place correctly or errors occured.

Explanation 6m Figure 2m.

b. Define Bus Arbitration. Explain centralised Arbitration. (2+6)

Answer:



The bus arbiter may be the processor or a separate unit connected to The bus. Above diagram processor contains The bus arbitration circuitry.

Processor is The bus master unless it grants bus mastership to one of the DMA controllers. A DMA controller indicates that it needs to become the bus master by activating the Bus-Request line, $\overline{BR}$. The signal on The Bus-Request line is the logical OR of the bus requests from all the devices connected to it. when bus-request is activated, The processor activates the Bus-Grant signal, BG1, indicates to the DMA controller that it can become the bus master. or use the bus. This signal is connected when it becomes free.

to all DMA controllers using a daisy-chain fashion. Thus, if DMA controller 1 is requesting the bus, it blocks the propagation of the grant signal to other devices.

The current bus master indicates to all devices that it is using the bus by activating another open-collector line called Bus-Busy $\overline{BBSY}$.

After receiving the Bus-Grant signal, a DMA controller waits for Bus-Busy to prevent other devices from using the bus at the same time.

Following sequence of events for the devices when DMA 2 requests and acquires bus mastership and later releases the bus is depicted below.

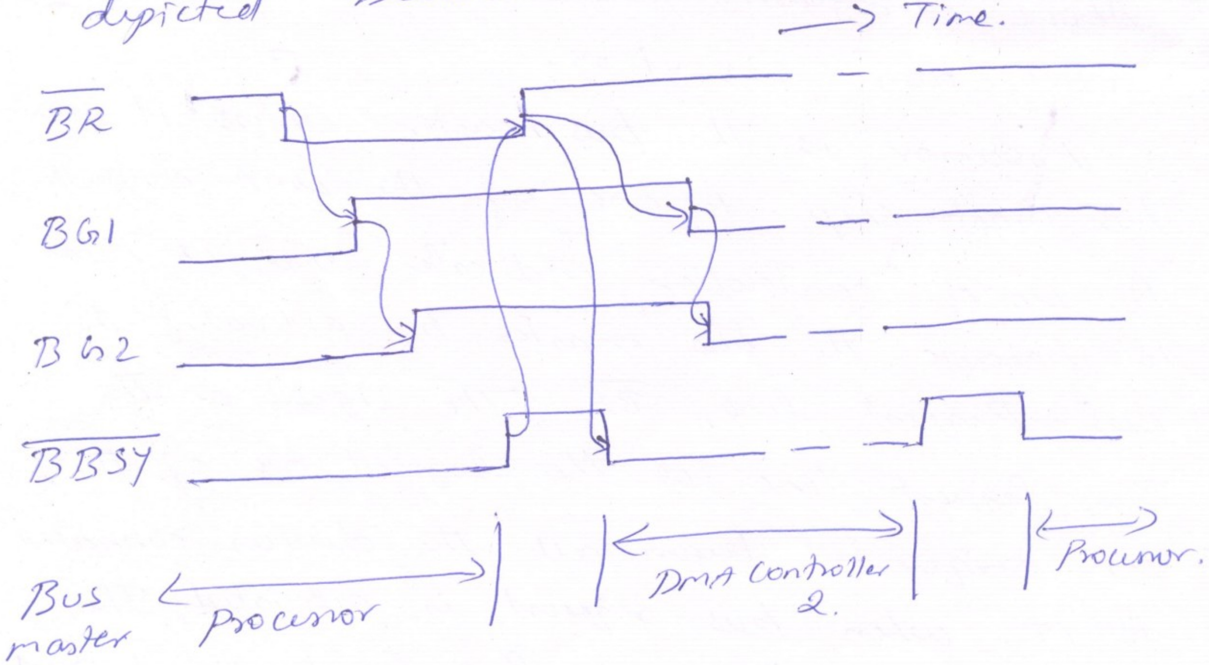
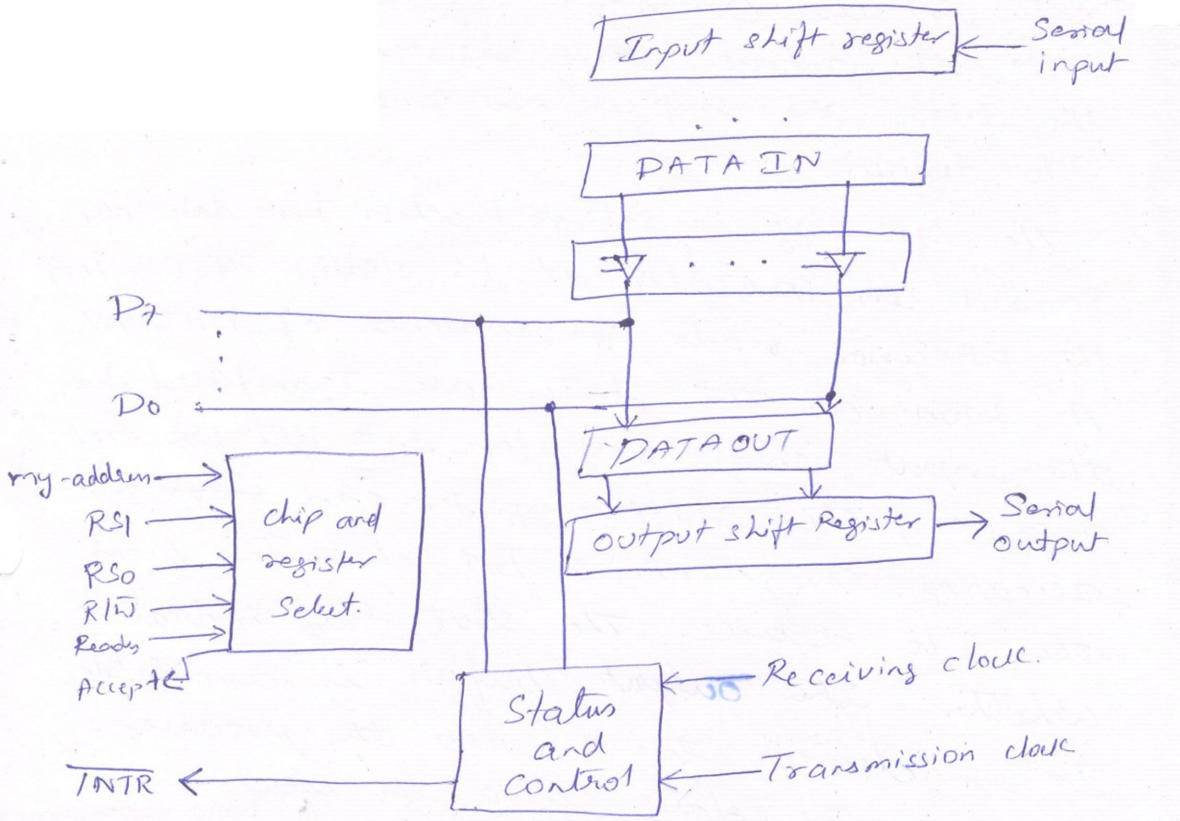


Figure & wave forms 4M Explanation 4M.

Q.5 a. Explain serial port with the help of serial interface block diagram. (8)

Answer:



A Serial port is used to connect the processor to I/o devices that requires transmission of data one bit at a time. The Key feature of an interface circuit for a Serial port is that it is capable of communicating in a bit-Serial fashion. on the device side and in a bit-parallel fashion on the bus side. The transformation between the parallel and serial formats is achieved with shift registers that have parallel access capability.

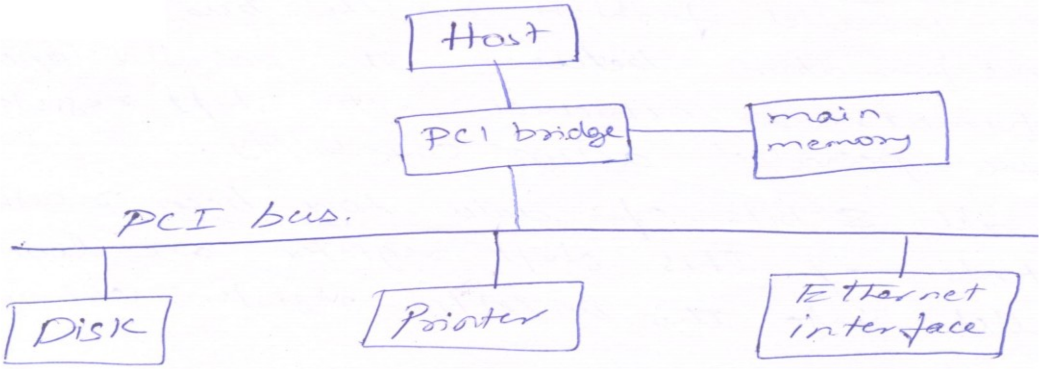
when all 8 bits of data have been received the contents of this shift register are loaded in parallel into the DATAIN register. Output data in the DATAOUT register are loaded into the output shift register, from which the bits are shifted out and sent to the I/o device.

The SIN flag is set to 1 when new data are loaded in DATAIN. it is cleared to 0 when the processor reads the contents of DATAIN. As soon as the data are transfelled from the input shift register into the DATAIN. register, the shift register can ~~shift~~ start accepting the next 8-bit character from the I/o device. The SOUT flag indicates whether the output buffer is available. It is cleared to 0 when the processor writes new data into the DATAOUT. register and set to 1 when data are transfered. from DATAOUT into the output shift register.

Diaglam 3m Explanation. 5m.

b. **Explain the use of PCI bus in a computer system.** **(8)**

Answer:



PCI is a low cost bus that is truly processor independent. An important feature that the PCI pioneered is a plug-and-play capability for connecting I/O devices.

The PCI is designed to support a write operation when the processor sends a memory address followed by a Sequence of data words, to be written in successive memory locations starting at that address.

The bus supports three address space. memory, I/O, & configuration. The PCI bridge provides a separate physical connection for the main memory.

At any given time, one device is the bus master. It has the right to initiate data transfers by issuing read and write commands. A master is called an initiator in PCI terminology. The addressed device that responds to read and write commands is called a target.

The main bus signals used for transfelling data are.

CLK	A 33-MHz or 66 MHz clock
FRAME#	Sent by the initiator to indicate the duration of a transaction.

AD 32 address/data. lines, which may be optionally increased to 64.

C/BE# 4 command/byte-enable lines (8 for 64-bit bus)

IRDY#, TRDY# Initiator-ready and Target-ready signals.

DEVSEL# A response from the device indicating that it has recognized its address and is ready for a data transfer transaction.

IDSEL# Initialization Device Select.

Signals whose name ends with symbol # are asserted when in The low voltage state.

A complete transfer operation on the bus, involving an address and a burst of data is called a transaction. Individual word transfers within a transaction are called phases.

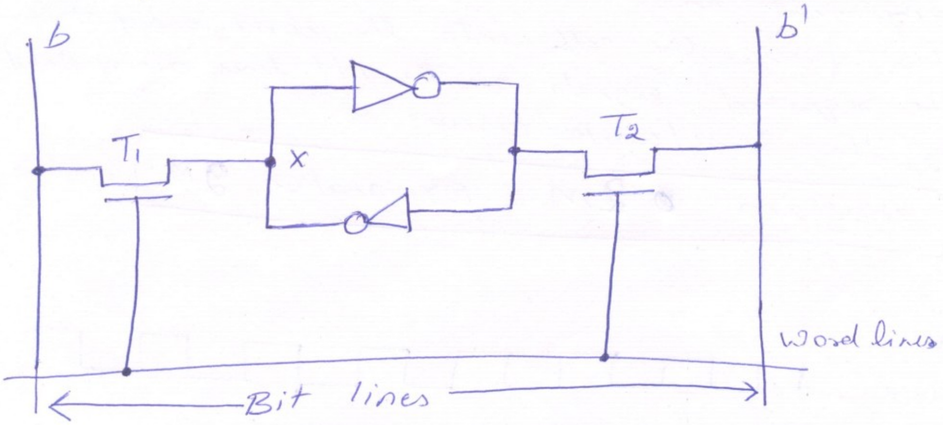
Diagram 2M Explanation 6M.

Q.6a. Describe the working of static RAM cell. (4)

Answer:

6 (a). memories that consists of circuits capable of retaining Their state as long as power is applied. are Known as Static memories.

The following figure shows static RAM cell.



Two inverters are cross-connected to form a latch. The latch is connected to two bit lines by transistors T₁ & T₂. These transistors act as switches that can be opened or closed under control of the word line. when the word line is at ground level, the transistors are turned off and the latch retains its state.

Read operation

In order to read the state of the SRAM cell the word line is activated to close switches T₁ and T₂. If the cell is in state 1 the signal on bit line b is high and the signal on bit line b' is low.

Sense/Write circuits at the end of the bit lines monitor the state of b and b' and set the output accordingly.

Write Operation

The state of the cell is set by placing the appropriate value on bit line b and its complement on b' and then activating the word line. This forces the cell into the corresponding state. The required signals on the bit lines are generated by the Sense/Write circuit.

Diagram 2M Explanation 2M

b. Write the timing diagram for burst read of length 4. (6)

Answer:

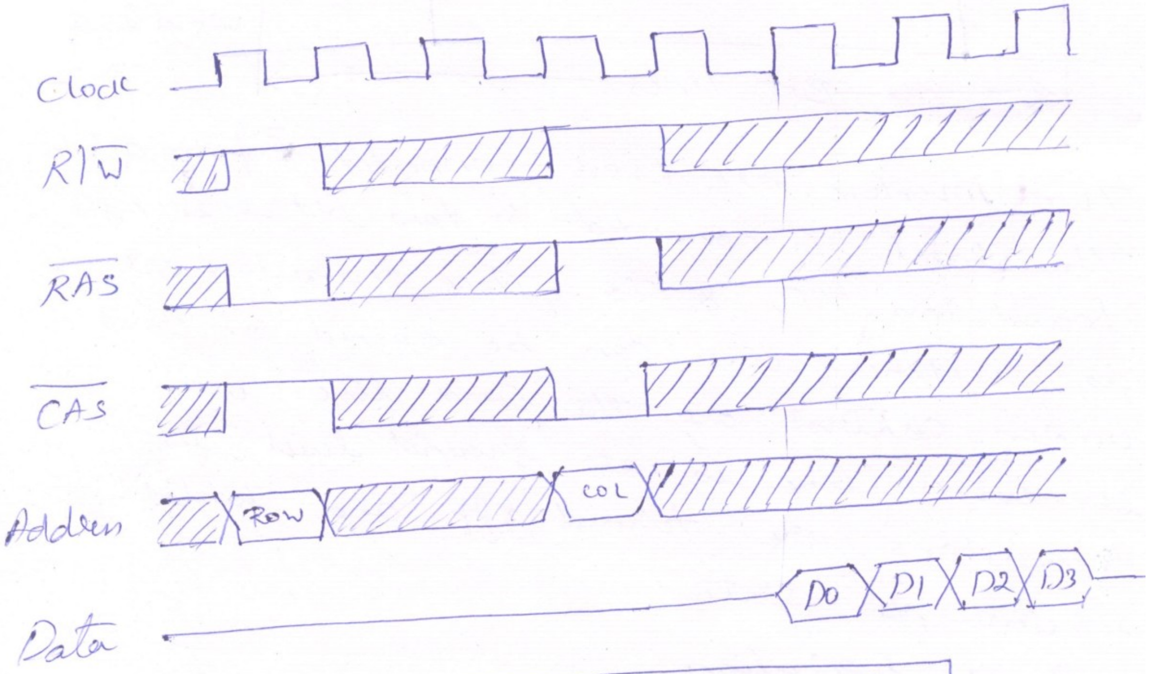
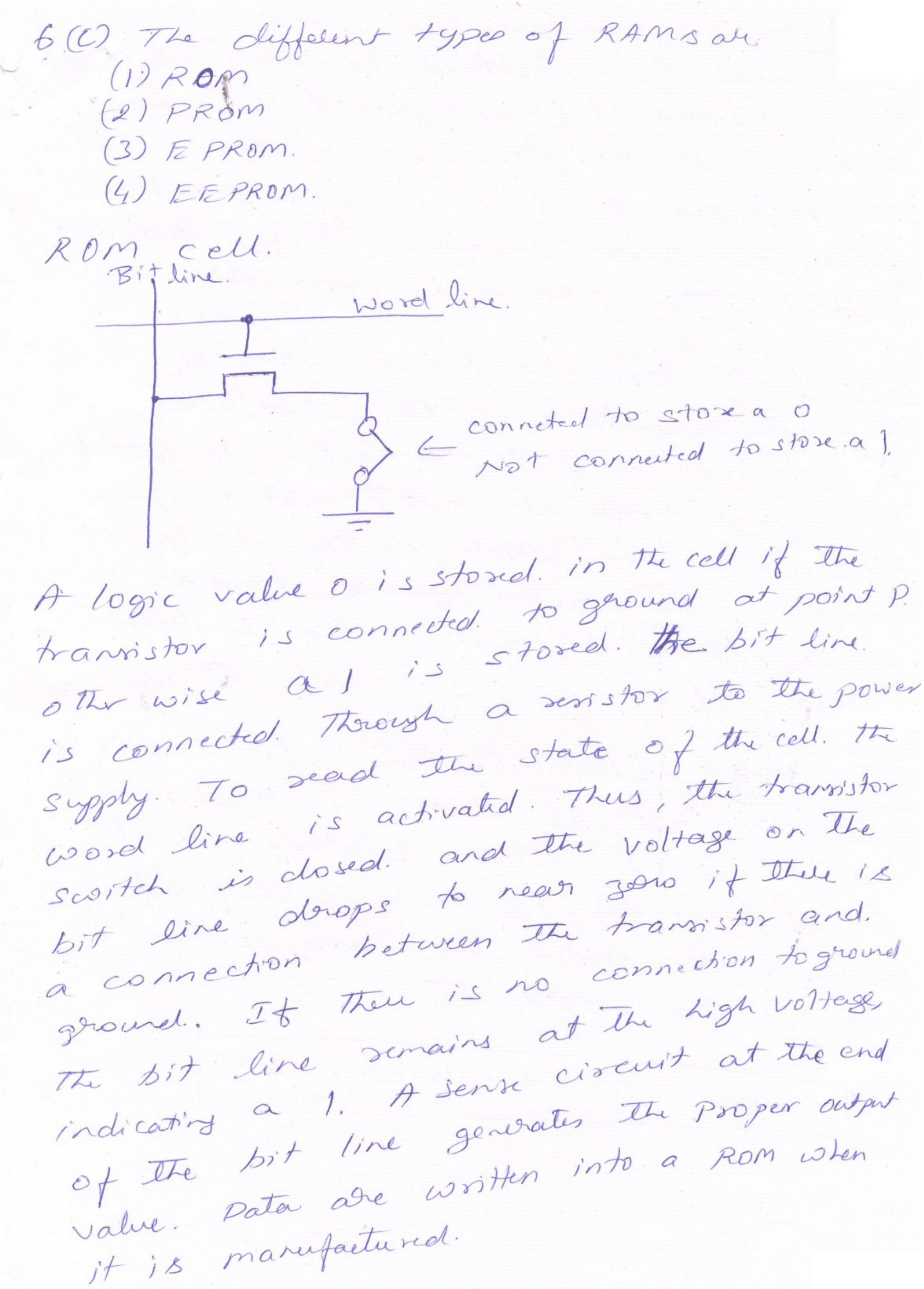


Diagram of waveforms 6M

c.Explain different types of read only memories. (6)

Answer:



6 (c) The different types of RAMs are

(1) ROM
(2) PROM
(3) E PROM.
(4) EEPROM.

ROM cell.

Bit line.
Word line.
connected to store a 0
Not connected to store a 1.

A logic value 0 is stored. in the cell if the transistor is connected. to ground at point P. otherwise a 1 is stored. the bit line is connected Through a resistor to the power supply. To read the state of the cell. the word line is activated. Thus, the transistor switch is closed. and the voltage on the bit line drops to near zero if there is a connection between the transistor and. ground. If there is no connection to ground the bit line remains at the high voltage indicating a 1. A sense circuit at the end of the bit line generates the proper output value. Data are written into a ROM when it is manufactured.

PROM. -> Programmable ROM. (PROM)

Programmability is achieved by inserting a fuse at point P. Before it is programmed, the memory contains all 0's. The user can insert 1s at the required locations. by burning out the fuses at these locations using high-current pulses. This process is irreversible.

EPROE -> Erasable Programmable ROM.

Reprogrammable ROM ~~it~~ has structure similar to the ROM cell, the connection to ground is always made at point P and a special transistor is used, which has the ability to function either as a normal transistor or as a disabled transistor That is always turned off.

EEPROM — Electrically Erasable Programmable ROM

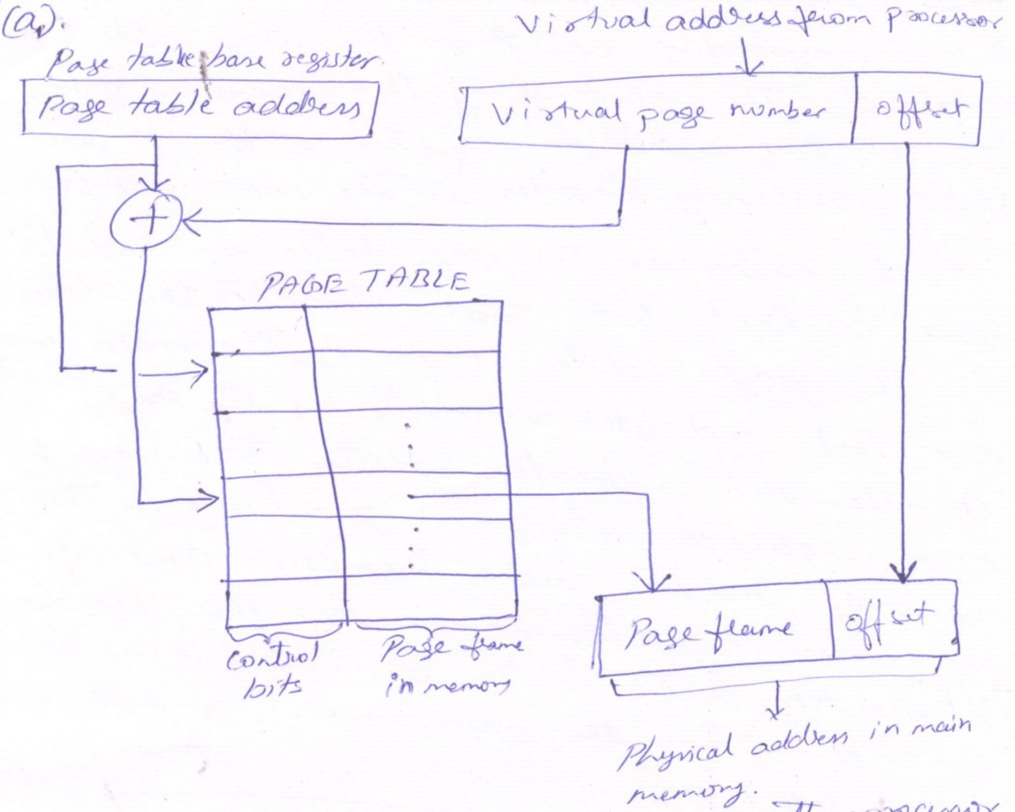
In this type of ROM's it is possible to erase the cell contents selectively. The only disadvantage of EEPROM is that different voltages are needed for erasing, writing, & reading the stored data.

Diagram 2M. Explanation 4M.

Q.7 a. Explain virtual-memory address translation without using TLB. (8)

Answer:

7(a).



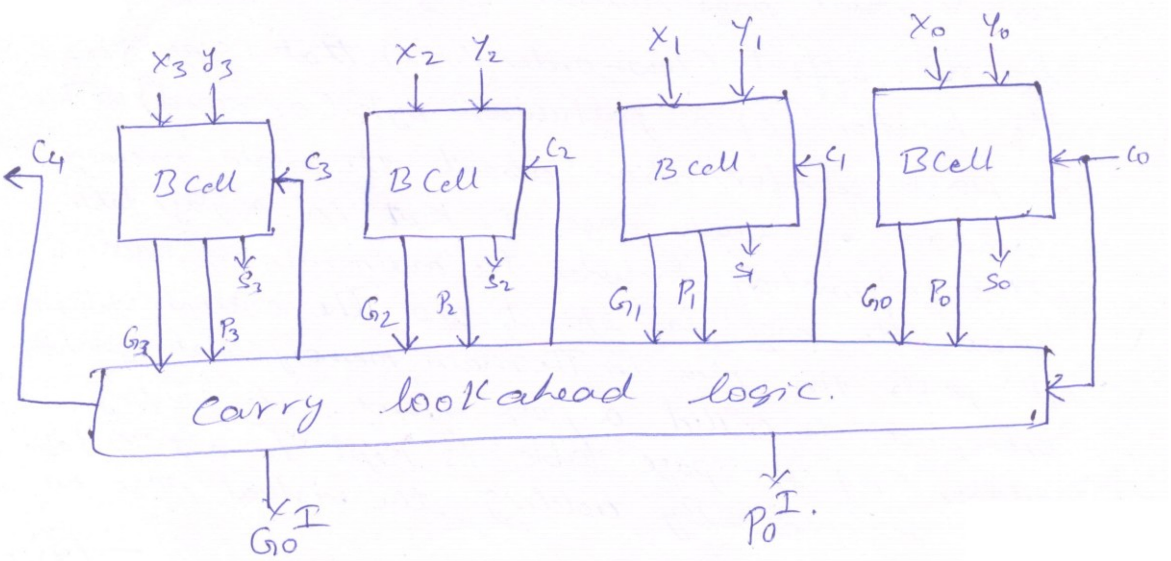
Each virtual address generated by the processor whether it is for an instruction fetch or an operand fetch/store operation, is interpreted as virtual page number (high order bits) followed by an offset (low-order bits) that specifies the location of a particular byte (or word) within a page. Information about the main memory location of each page is kept in a page table. This information includes the main memory address where the page is stored. and the current status of the page. An area in the main memory that can hold one page is called a page frame. The starting address of the page table is kept in a page table base register. By adding the virtual page no to the contents of This register, the address of The corresponding entry in the page table is obtained. The contents of this location give the starting address of the page if That page currently resides in The main memory.

Each entry in the page table also includes some control bits that describe the status of The page while it is in The main memory. One bit indicates validity of the page. Another bit indicates whether the page has been modified during its residency in the memory. As in cache memories this information is needed to determine whether the page should be written back to the disk before it is removed from the main memory to make room for another page.

Diagram 3M Explanation 5M.

b. Explain 4-bit adder with carry look ahead logic. (8)

Answer:



A fast adder circuit must speed up the generation of the carry signals. The logic expressions for s_i (sum) and c_{i+1} (carry-out) of stage i are.

$$s_i = x_i \oplus y_i \oplus c_i$$

&

$$c_{i+1} = x_i y_i + x_i c_i + y_i c_i$$

Factoring the second equation into

$$c_{i+1} = x_i y_i + (x_i + y_i) c_i$$

We can write.

$$c_{i+1} = G_i + P_i c_i$$

where.

$$G_i = x_i y_i \quad \text{and} \quad P_i = x_i + y_i$$

The expressions G_i & P_i are called the generate and propagate functions for stage i.

The carriers for a 4 bit adder can be implemented. as.

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_0.$$

$$C_3 = G_2 + P_2 G_1 + P_2 G_1 G_0 + P_2 P_1 P_0 C_0.$$

$$C_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0.$$

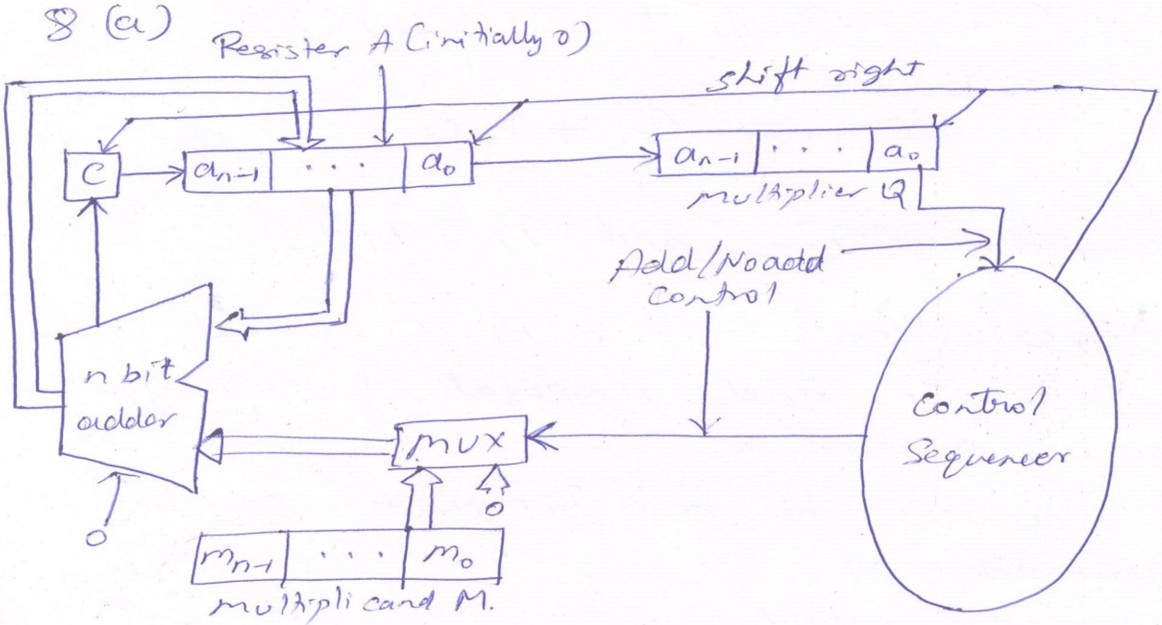
The carriers are implemented in the block labeled carry-lookahead logic. An adder implemented in this form is called a carry lookahead adder. Delay through the adder is 3 gate delays for all carry bits and 4 gate delays. for all sum bits.

Diagram 3M. Explanation & equations 5M.

Q.8 a. Explain sequential circuit binary multiplier with an example. (8)

Answer:

8 (a)



The simplest way to perform multiplication is to use the adder circuitry in the ALU for a number of sequencial steps. This circuit performs multiplication by using a single n-bit adder n times to implement The spatial addition performed by the n rows of ripple-carry adders.

Register A and Q combined hold PP_i while multipler bit q_i generates the signal. Add/Noadd. This signal controls the addition of the multiplicand, M, to PP_i to generate PP(i+1). The product is computed in n cycles. The partial product grows in length by one bit per cycle. from the initial vector, PP0, of n 0s in register A. The carry out from the adder is stored in flip-flop C.

In the begining.

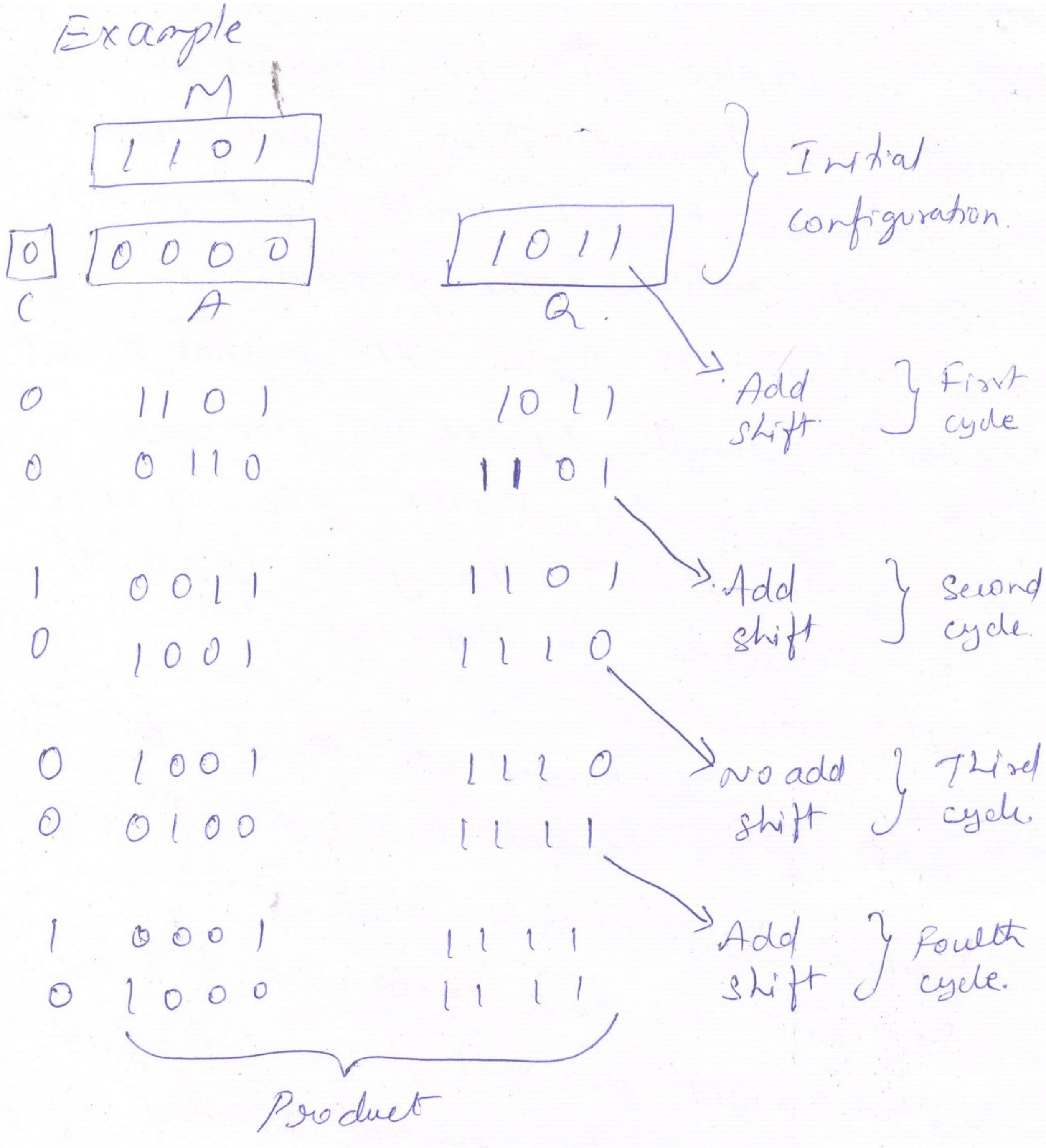
multiplier is loaded to Register Q.

multiplicand is loaded to Register M.

C and A are cleared to 0.

At the end of each cycle C, A, & Q are shifted right one bit position to allow for growth of the partial product. as the multiplier is shifted out of register Q. Because of this shifting, multiplier bit q_i appears at the LSB position of Q to generate the Add/Noadd signal at the correct time. starting with q_0 during the first cycle, q_1 during the second cycle and so on. After they are used, the multiplier bits are discarded by the right-shift operation.

Example



	M			
	1101			Initial configuration
C	A	Q		
0	0000	1011		
0	1101	1011	Add	First cycle
0	0110	1101	shift	
1	0011	1101	Add	Second cycle
0	1001	1110	shift	
0	1001	1110	no add	Third cycle
0	0100	1111	shift	
1	0001	1111	Add	Fourth cycle
0	1000	1111	shift	

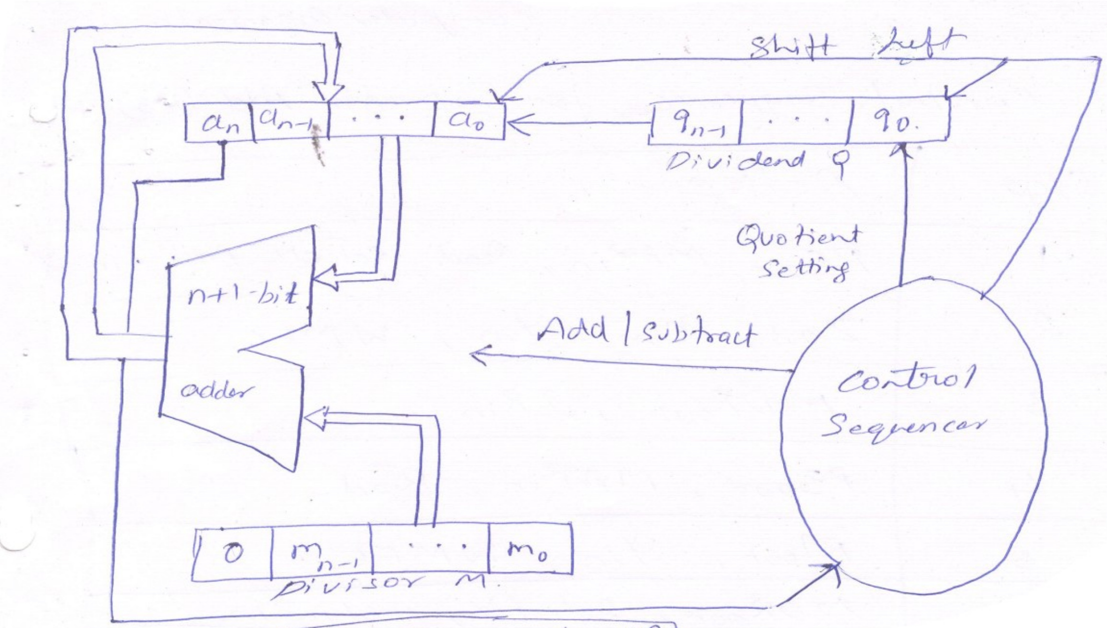
Product

Diagram 3M. Explanation 4M.

Example 3M.

b. Write the circuit arrangement for restoring division. (4)

Answer:



[Diagram 4 M.]

c. Write the algorithm for non restoring division. (4)

Answer:

Algorithm for non restoring division.

Step1: Do the following n times.

1. If the sign of A is 0, shift A and Q left one bit position and subtract M from. A; otherwise, shift A and Q left and add m to A.

2. Now, if the sign of A is 0, set q_0 to 1; otherwise, set q_0 to 0.

step 2: If the sign of A is 1, add m to A.

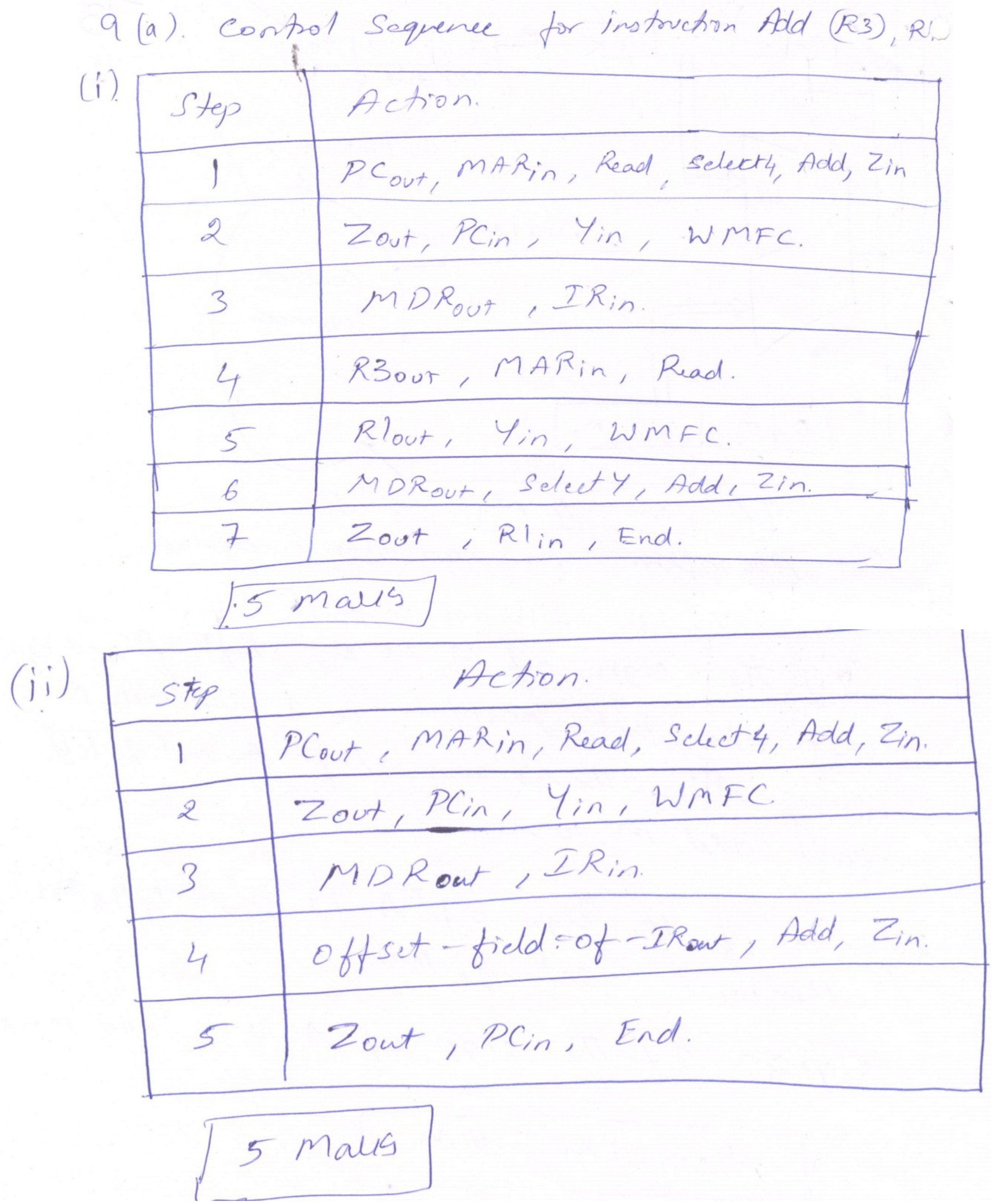
[Algorithm 2M]

Q.9 a. Write the control sequence for the following: (5×2)

(i) Instruction add (R3), R4

(ii) Unconditional Branch Instruction

Answer:



9 (a). Control Sequence for instruction Add (R3), R

(i)

Step	Action.
1	PCout, MARin, Read, select4, Add, Zin
2	Zout, PCin, Yin, WMFC.
3	MDRout, IRin.
4	R3out, MARin, Read.
5	R1out, Yin, WMFC.
6	MDRout, SelectY, Add, Zin.
7	Zout, R1in, End.

5 Marks

(ii)

Step	Action.
1	PCout, MARin, Read, Select4, Add, Zin.
2	Zout, PCin, Yin, WMFC
3	MDRout, IRin.
4	offset-field-of-IRout, Add, Zin.
5	Zout, PCin, End.

5 Marks

b. Write the three bus organization of the data path. (6)

Answer:

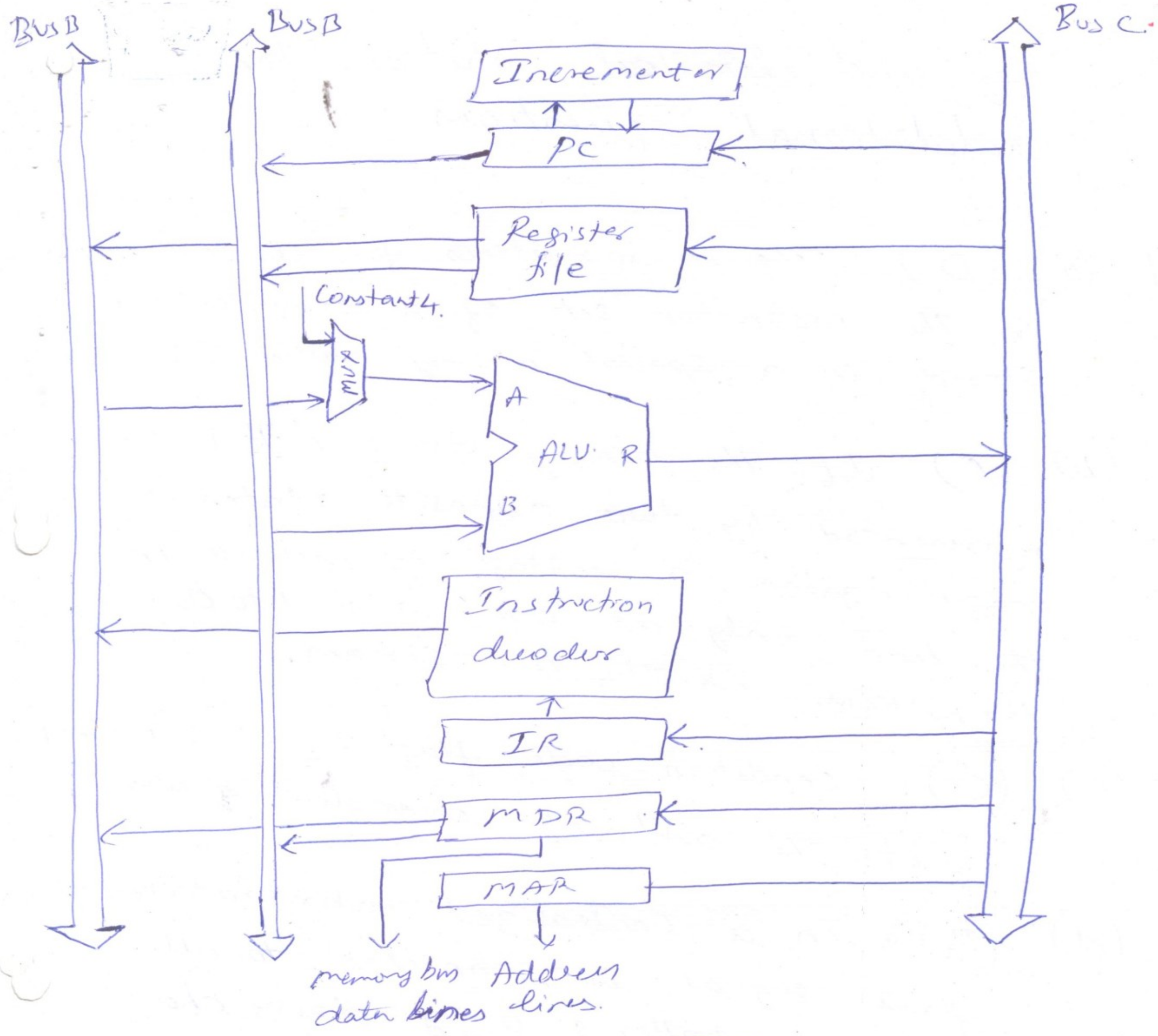


diagram 6 marks

TEXT BOOK

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, TMH, 2002