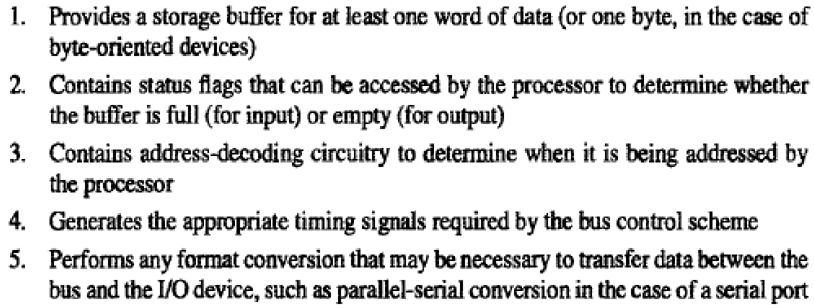
Internal Assessment Test 2–December2022

(Scheme & solution)

Sub:	Computer Organization and Architecture				SubCode:	21CS34	Branch:	ISE
Date:	27/12/2022	Duration:	90min's	MaxMarks:	50	Sem/Sec:	IIIA, B &C	OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1a	List out functions of the I/O interface.	4	CO2	L1



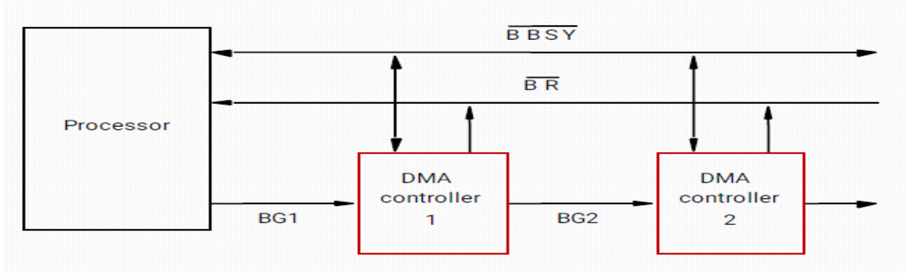
1. Provides a storage buffer for at least one word of data (or one byte, in the case of byte-oriented devices)
2. Contains status flags that can be accessed by the processor to determine whether the buffer is full (for input) or empty (for output)
3. Contains address-decoding circuitry to determine when it is being addressed by the processor
4. Generates the appropriate timing signals required by the bus control scheme
5. Performs any format conversion that may be necessary to transfer data between the bus and the I/O device, such as parallel-serial conversion in the case of a serial port

1b	Define Bus arbitration. Explain in detail any one approach of bus arbitration.	1+2+3 (Definition +Diagram+ explanation)	CO2	L1

Processor and DMA controllers both need to initiate data transfers on the bus and access main memory. The device that is allowed to initiate transfers on the bus at any given time is called the bus master. When the current bus master relinquishes its status as the bus master, another device can acquire this status. The process by which the next device to become the bus master is selected and bus mastership is transferred to it is called bus arbitration.

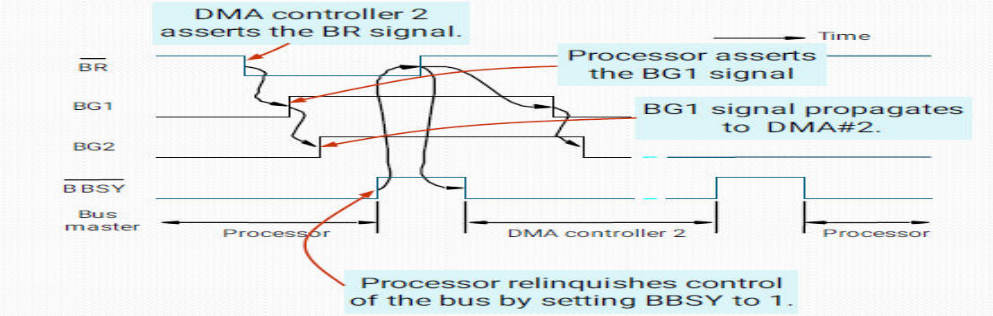
- 2 Approaches of Bus arbitration
- ➢ Centralized arbitration:
  - A single bus arbiter performs the arbitration.
- ➢ Distributed arbitration:
  - All devices participate in the selection of the next bus master.

Centralized Bus Arbitration



Bus arbiter may be the processor or a separate unit connected to the bus.Normally, the processor is the bus master, unless it grants bus membership to one of the DMA controllers. DMA controller requests the control of the bus by asserting the Bus Request (BR) line. In response, the processor activates the Bus-Grant1 (BG1) line, indicating that the controller may use the bus when it is free. BG1 signal is connected to all DMA controllers in a daisy chain fashion. BBSY signal is 0, it indicates that the bus is busy.

When BBSY becomes 1, the DMA controller which asserted BR can acquire control of the bus.

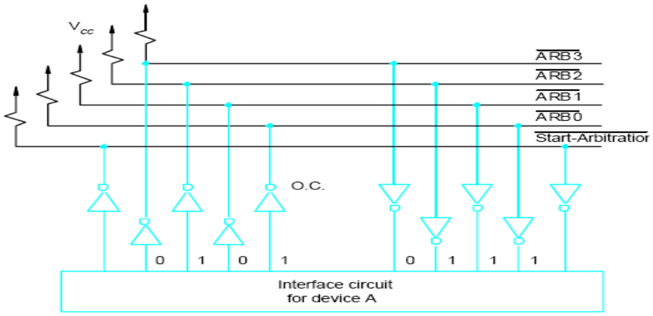

DMA controller 2 asserts the BR signal.
Time
$\overline{BR}$
BG1
BG2
$\overline{BBSY}$
Bus master
Processor
DMA controller 2
Processor
Processor asserts the BG1 signal
BG1 signal propagates to DMA#2.
Processor relinquishes control of the bus by setting BBSY to 1.

OR

Distributed arbitration

All devices waiting to use the bus share the responsibility of carrying out the arbitration process. Arbitration process does not depend on a central arbiter and hence distributed arbitration has higher reliability. Each device is assigned a 4-bit ID number.

All the devices are connected using 5 lines, 4 arbitration lines to transmit the ID, and one line for the Start- Arbitration signal. To request the bus a device: Asserts the Start-Arbitration signal. Places its 4-bit ID number on the arbitration lines. The pattern that appears on the arbitration lines is the logical-OR of all the 4-bit device IDs placed on the arbitration lines.



Arbitration process:

Each device compares the pattern that appears on the arbitration lines to its own ID, starting with MSB. If it detects a difference, it transmits 0s on the arbitration lines for that and all lower bit positions. The pattern that appears on the arbitration lines is the logical-OR of all the 4-bit device IDs placed on the arbitration lines. Device A has the ID 5 and wants to request the bus. - Transmits the pattern 0101 on the arbitration lines.

Device B has the ID 6 and wants to request the bus: - Transmits the pattern 0110 on the arbitration lines. Pattern that appears on the arbitration lines is the logical OR of the patterns - Pattern 0111 appears on the arbitration lines.

Arbitration process:

Each device compares the pattern that appears on the arbitration lines to its own ID, starting with MSB. If it detects a difference, it transmits 0s on the arbitration lines for that and all lower bit positions. Device A compares its ID 5 with a pattern 0101 to pattern 0111. It detects a difference at bit position 0, as a result, it transmits a pattern 0100 on the arbitration lines. The pattern that appears on the arbitration lines is the logical-OR of 0100 and 0110, which is 0110. This pattern is the same as the device ID of B, and hence B has won the arbitration

2	Explain DMA in detail.	4+6 (Theory+ Register with explanation)	CO4	L3

Direct Memory Access (DMA):

A special control unit may be provided to transfer a block of data directly between an I/O device and the main memory, without continuous intervention by the processor.

Control unit which performs these transfers is a part of the I/O device’s interface circuit. This control unit is called as a DMA controller.DMA controller performs functions that would be normally carried out by the processor: For each word, it provides the memory address and all the control signals. To transfer a block of data, it increments the memory addresses and keeps track of the number of transfers. DMA controller can transfer a block of data from an external device to the processor, without any intervention from the processor. However, the operation of the DMA controller must be under the control of a program executed by the processor. That is, the processor must initiate the DMA transfer. To initiate the DMA transfer, the processor informs the DMA controller of: Starting address, Number of words in the block. Direction of transfer (I/O device to the memory, or memory to the I/O device).Once the DMA controller completes the DMA transfer, it informs the processor by raising an interrupt signal.

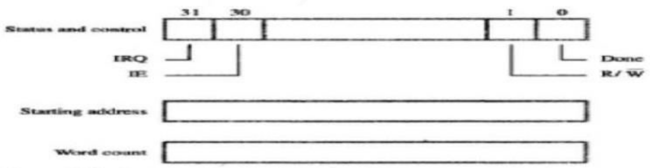
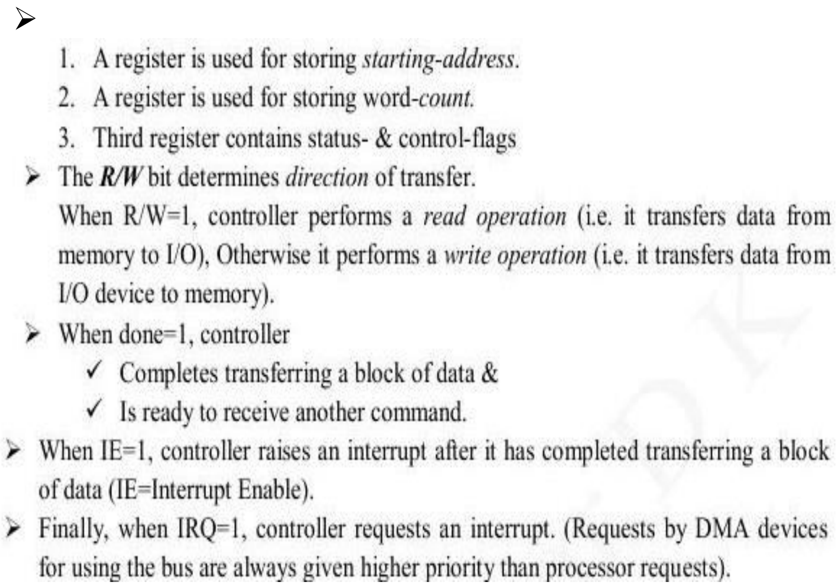
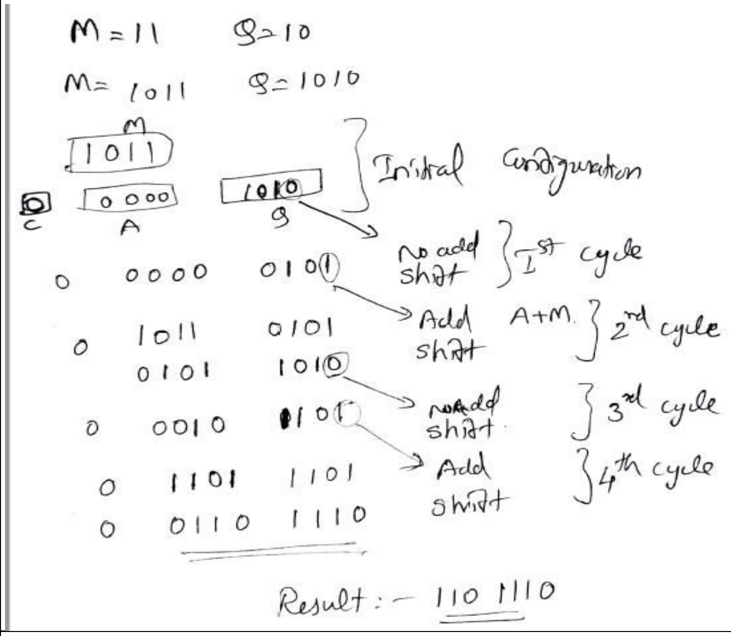


Figure 4.18 Registers in a DMA interface.

	 ➢ 1. A register is used for storing *starting-address.* 2. A register is used for storing word-*count.* 3. Third register contains status- & control-flags ➢ The ***R/W*** bit determines *direction* of transfer. When R/W=1, controller performs a *read operation* (i.e. it transfers data from memory to I/O), Otherwise it performs a *write operation* (i.e. it transfers data from I/O device to memory). ➢ When done=1, controller ✓ Completes transferring a block of data & ✓ Is ready to receive another command. ➢ When IE=1, controller raises an interrupt after it has completed transferring a block of data (IE=Interrupt Enable). ➢ Finally, when IRQ=1, controller requests an interrupt. (Requests by DMA devices for using the bus are always given higher priority than processor requests).			
3	Multiply the given numbers using a sequential positive binary multiplier. i)11, 10  M = 11 Q = 10 M = 1011 Q = 1010 M 1011 0 C 0000 A 1010 Q Initial configuration 0 0000 0101 No add shift 1st cycle 0 1011 0101 0101 1010 Add A+M. shift 2nd cycle 0 0010 0101 no add shift 3rd cycle 0 1101 1101 0 0110 1110 Add shift 4th cycle Result :- 110 1110	10	CO4	L3
4	List out the actions needed to execute the instruction ADD (R3), R1. Write and explain the sequence of control steps for the execution of the same. Add (R3), R1.Fetch the instruction. Fetch the first operand (the contents of the memory location pointed to by R3). Perform the addition. Load the result into R1	7+3 (Steps +Expla nation)	CO4	L2

5	Draw and explain the single-bus organization of the data path inside a processor ● ALU ● Registers for temporary storage ● Various digital circuits for executing different micro operations.(gates, MUX,decoders,counters). ● Internal path for movement of data between ALU and registers. ● Driver circuits for transmitting signals to external units. ● Receiver circuits for incoming signals from external units. ● PC: ❖ Keeps track of execution of a program ❖ Contains the memory address of the next instruction to be fetched and executed. MAR: ❖ Holds the address of the location to be accessed. ❖ I/P of MAR is connected to Internal bus and an O/p to external bus. MDR: ❖ Contains data to be written into or read out of the addressed location. ❖ IT has 2 inputs and 2 Outputs. ❖ Data can be loaded into MDR either from memory bus or from internal processor bus. The data and address lines are connected to the internal bus via MDR and MAR Registers: ❖ The processor registers R0 to Rn-1 vary considerably from one processor to another. ❖ Registers are provided for general purpose used by programmer. ❖ Special purpose registers-index & stack registers. ❖ Registers Y,Z &TEMP are temporary registers used by processor during the execution of some instruction. Multiplexer: ❖ Select either the output of the register Y or a constant value 4 to be provided as input A of the ALU. ❖ Constant 4 is used by the processor to increment the contents of PC. ALU: Used to perform arithmetic and logical operation. Data Path: The registers, ALU and interconnecting bus are collectively referred to as the data path.	4+6 (Diagram+Explnanation)	CO4	L2
6	Evaluate the following operations on the 5-bit signed numbers using 2's complement representation system. Also indicate whether an overflow has occurred. (i)(-10)+ (-13) (ii) (-10) - (+7)	5+5	CO4	L4

	(−10) + (−13) 10 − 01010 −10 = 10101+1 = 10110 (−10) 10110 + 10011 1)01001 − overflow as it is enceeding the (−23) range −16 to +15 −10 − (7) = −10 + −7 . 7 − 00111 11000+ 1 11001 (−7) 10110 (−10) + 11001 1)01111 (−17) − overflow − as it enceeds the range −16 to +15.			

FacultySignature CCISignature HODSignature