Name : Muhammad Musa

Department : BS(CS)

Semester : 4yh

ID # : 15366

Sessional Assignment No : 3rd

Subject : Computer Architecture

Submitted To :Muhammad Amin Sir

Dated : 17th May 2020

M. Amin Sir, BS (CS)

Name : Muhammad Musa
ID : 15366
Subject : Computer Architecture
Deptt : BS (CS)
Assignment No : 03

Q1.

Ans:(A) 1. The first is to disable interrupts while an interrupt is being processed. A disabled interrupt simply means that the processor can and will ignore that interrupt request signal.

2. A second approach is to define priorities for interrupts and to allow an interrupt of higher priority to cause a lower-priority interrupt handler to be itself interrupted.

Ans:(B) The types of exchanges that are needed by indicating the major forms of input and output for processor, memory, and I/O modules are;

- Memory to processor:
  The processor reads an instruction or a unit of data from memory.
- Processor to memory:
  The processor writes a unit of data to memory.

- **I/O to processor:-** The processor reads data from an I/O device via an I/O module.
- **Processor to I/O:-** The processor sends data to the I/O device.
- **I/O to or from memory:-** For these two cases, an I/O module is allowed to exchange data directly with memory, without going through the processor, using direct memory access.

Ans(C) **GPI Protocol Layer:-** In this layer, the packet is defined as the unit of transfer. One key function performed at this level is a cache coherency protocol, which deals with making sure that main memory values held in multiple caches are consistent. A typical data packet payload is a block of data being sent to or form a cache.

Ans:(D) **Physical and Logical Architecture of PCIe:-**

→ A root complex device, also referred to as a chipset or a host bridge, connects the processor

and memory subsystem to the PCI Express switch fabric comprising one or more PCIe and PCIe switch devices.

→ PCIe links from the chipset may attach to the following kinds of devices that implement PCIe:

• Switch:- The switch manages multiple PCIe streams.

• PCIe endpoint:- An I/O device or controller that implements PCIe, such as a Gigabit ethernet switch, a graphics or video controller, disk interface, or a communication controller.

• Legacy endpoint:- Legacy endpoint category is intended for existing designs that have been migrated to PCI Express, and it allows legacy behaviors such as use of I/O space and locked transactions.

• PCIe/PCI bridge:- Allow older PCI devices to be connected to PCIe-based systems.

Q2.

Ans(A) Instruction Cycle:- The processing required for a single instruction is called an "instruction cycle."

Using the simplified two-step description given previously, the instruction cycle is depicted. The two steps are referred to as the fetch cycle and the execute cycle. Program execution halts only if the machine is turned off, some sort of unrecoverable error occurs, or a program instruction that halts the computer is encountered.

Ans(B) Instruction Cycle State Diagram:-

The states in instruction cycle can be described as follows:

→ Instruction address calculation (iac):-
Determine the address of the next instruction to be executed. Usually, this involves adding a fixed number to the address of the previous instruction.

→ Instruction fetch (if):-
Read instruction from its memory location into the processor.

→ Instruction operation decoding (iod):-
Analyze instruction to determine type of operation to be performed and operand(s) to be used.

→ Operand address calculation (oac):-
If the operation involves reference to an operand in memory or available via I/O, then determine

the address of the operand.

→ Operand fetch (of):- Fetch the operand from memory or read it in from I/O.

→ Data Operation (do):- Perform the operation indicated in the instruction.

→ Operand store (OS):- Write the result into memory or out to I/O.

Ans:(c) Classes of Interrupts:-

I. Program:- It is generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space.

II. Timer:- It is generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.

III. I/O:- It is generated by an I/O controller, to signal normal completion of an operation, request service

from the processor, or to signal a variety of error conditions.

IV. Hardware Failure:-
It is generated by a failure such as power failure or a memory parity error.

Ans:(D) Bus Interconnection Scheme:-
The most common computer interconnection structures are based on the use of one or more system buses.

A system bus consists, typically of from about fifty to hundreds of separate lines. The lines can be classified into three functional groups : data, address, and control lines.

a) Data Lines:-
The data lines provide a path for moving data among system modules. These lines, collectively are called the data bus.

b) Address Lines:-
The address lines are used to designate the source or destination of the data on the data bus.

c) Control Lines:-
The control lines are used to control the access to and the use of the data and

address lines. Because the data and the address lines are shared by all components, there must be a means of controlling their use.

Q3.

Ans:(A) Programming in Hardware:-

"The "program" is in the form of hardware and is termed as hardwired program."

Suppose we construct a general purpose configuration of arithmetic and logic functions. This set of hardware will perform various functions on data depending on control signals applied to the hardware. In the original case of customized hardware, the system accepts data and produces results.

Programming in Software:-

"The new method of programming which is a sequence of codes or instructions is called software programming."

In this method, Programming is much easier. Instead of rewiring the hardware for each new program, all we need to do is provide a new sequence of codes. Each code is, in effect, an instruction, and part of the

hardware interprets each instruction and generates control signals.

Ans: (B)
- In the interrupt cycle, the processor checks to see if any interrupts have occured, indicated by the presence of an interrupt signal.
- If no interrupts are pending, the processor proceeds to the fetch cycle and fetches the next instruction of the current program.

Ans: (C) Disabled Interrupt:-

A disabled interrupt simply means that the processor can and will ignore that interrupt request signal. If an interrupt occurs during this time, it generally remains pending and will be checked by the processor after the processor has enabled interrupts.

Nested Interrupt:-

A nested interrupt is to allow an interrupt of higher priority to cause a lower-priority interrupt handler to be itself interrupted.

Q4:-

Ans:(A) Memory (contents in hex):
300: 3005; 301: 5940;
302: 7006

Step 1: 3005 → IR
Step 2: 3 → AC
Step 3: 5940 → IR
Step 4: 3+2 = 5 → AC
Step 5: 7006 → IR
Step 6: AC → Device 6

Ans:(B)

1. a. The PC contains 300, the address of the first instruction. This value is loaded into the MAR.
   b. The value in location 300 is loaded into the MBR, and the PC is incremented. These two steps can be done in parallel.
   c. The value in the MBR is loaded into the IR.
2. a. The address portion of the IR (940) is loaded into the MAR.
   b. The value in location 940 is loaded into the MBR.
   c. The value in the MBR is loaded into the AC.
3. a. The value in the PC (301) is loaded into the MAR.
   b. The value in location 301 is loaded into the MBR, and the

PC is incremented.
c. The value in the MBR is loaded into the IR.

4. a. The address portion of the IR (941) is loaded into the MAR.
b. The value in location 941 is loaded into the MBR.
c. The old value of the AC and the value of location MBR are added and the result is stored in the AC.

5. a. The value in the PC (302) is loaded into the MAR.
b. The value in location 302 is loaded into the MBR, and the PC is incremented.
c. The value in the MBR is loaded into the IR.

6. a. The address portion of the IR (941) is loaded into the MAR.
b. The value in the AC is loaded into the MBR.
c. The value in the MBR is stored in location 941.

Ans:(C) a. 2^{24} = 16MBytes

b. (1) If the local address bus is 32 bits, the whole address can be transferred at once and decoded in memory. However, because the data bus is only 16 bits, it will

require 2 cycles to fetch a 32-bit instruction or operand.

(2) The 16 bits of the address placed on the address bus can't access the whole memory. Thus a more complex memory interface control is needed to latch the first part of the address and then the second part.

c. The program counter must be at least 24 bits. Typically, a 32-bit microprocessor will have a 32-bit external address bus and a 32-bit program counter, unless onchip segment registers are used that may work with a smaller program counter. If the instruction register is to contain the whole instruction, it will have to be 32-bit long; if it will contain only the op code then it will have to be 8 bits long.

Ans: (D) Clock Cycle =
1 = 125ns 8MHz Bus cycle = 4x125ns
= 500ns 2 bytes transferred every 500ns; thus transfer rate is 4 MBytes/sec.

Doubling the frequency may mean adopting a new chip manufacturing technology; doubling

the external data bus means wider on-chip data bus drivers/latches and modifications to the bus control logic. In the first case, the speed of the memory chips will also need to double not to slow down the microprocessor; in the second case, the "wordlength" of the memory will have to double to be able to send/receive 32-bit quantities.

Ans:(E) a. During a single bus cycle, the 8-bit microprocessor transfers one byte while the 16-bit microprocessor transfer two bytes. The 16-bit microprocessor has twice the data transfer rate.

b. Suppose we do 100 transfers of operands and instructions, of which 50 are one byte and 50 are two bytes long. The 8-bit microprocessor takes $50 + (2 \times 16 - 50) = 150$ bus cycles for the transfer. The 16-bit microprocessor requires $50 + 50 = 100$ bus cycles. Thus, the data transfer rates differ by a factor of 1.5.

Ans:(F) A bus cycles takes 0.25μs, so a memory cycles takes 1μs. If both

operands are even, aligned, it takes 2μs to fetch the two operands. If one is odd-aligned, the time required is 3μs. If both are odd-aligned, the time required is 4μs.

Ans: (G) Consider a mix of 100 instructions and operands. On average, they consist of 20% 32 bit items, 40% 16-bit items, and 40% 8-bit items. The number of bus cycles required for the 16-bit microprocessor is (2×20) + 40 + 40 = 120. For the 32-bit microprocessor, the number required is 100. This amounts to an improvement of 20/120 or about 17%.

[The END]