Name: Shahab Zeb

Department: BS(CS)

Semester: 4th

ID # 14590

Assignment NO: 3rd (Chapter # 03)

Subject: Computer Architecture

Submitted To Muhammad Amin Sir

Dated:

Q1 Give answer to each of the following

A. Discuss the two approaches for dealing with multiple interrupts?

Ans:

1. First is to disable interrupts while an interrupts is being processed. A disabled interrupts simply means that the processor can & will ignore that interrupt request signal.

The Drawback to the preceding approach is that it does not take into account relative priority or time-critical needs.

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

B. Discuss the types of exchanges that are needed by indicating the major form of input & output for processor memory & I/O modules?

Ans:

The type of exchanges that are needed by indicating the major forms

inputer output for processor, memory & I/O modules are:

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

c. Discuss the Quick path interconnect (QPI) protocol layer.

Ans: QPI 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 from a device

9. Discuss the physical & logical architecture of PCIe in detail?

Ans:

Physical & logical Architecture of PCIe:

- Root Complex devices also referred as a chipset or a host bridge connect processor & memory subsystem to PCI Express ~~syst~~ switch fabric comprising one or more PCIe & PCIe switch devices.
- PCIe links from the chipset may attached to the following kinds of devices that implement PCIe:
  - Switch: The switch manages multiple PCIe streams.
  - PCIe endpoint: An I/O devices 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, & it allows legacy behaviors such as use of I/O space & locked

transactions

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

Q 2: Write short notes on the following

A): Instruction cycle:

The processing required for a single instruction is instruction cycle.

Using the simplified two-step description given previously, the instruction cycle is depicted. The two steps are the Fetch Fetch cycle & the execute cycle

Program execution halts only if the machine is turned off, some sort of unrecoverable error occurs, or a program instruction that halt the computer is encountered.

B. Instruction cycle State diagram:

The states in instruction cycle can be described as:

- Instruction address calculation (iac):

Determine the address of the next instruction to be executed. This involved a fixed number to the address of the previous instruction.

- Instruction Fetch (IF):
  Read instruction from its memory location into processor.
- Instruction operation decoding (IOD):
  Analyze instruction to determine type of operation to be performed & operand(s) to be used.
- Operand address calculation (OAC):
  If the operation includes 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.

C): Classes of Interrupts:

1 PROGRAM:
It is generated by some condition that occurs as a result of an instruction execution such as arithmetic overflow, division by zero, attempts to execute an illegal machine

instruction, or reference outside a users allowed memory.

II) TIMER:

Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.

III) I/O:

Generated by I/O controller to signal normal completion of an operation, request services from the processor or to signal a variety of error conditions.

IV) HARDWARE FAILURE

Generated by a failure such as power failure or memory parity error.

B) Bus Interconnection Scheme:

The most common computer interconnection structures are based on the use of one or more system buses. A system bus consist typically of from about fifty to hundreds of seperate lines. The lines can be classified into three functional Groups, Data, Address & Control lines

(a) DATA Lines:

Provide a path for moving data among system modules

these lines, collectively called the data bus.

b) Address lines:

These are used to designate the source or destination of the data in the data bus.

The width of the address bus determines the maximum possible memory capacity of the system.

c) Control lines:

are used to control the access to & the use of the data and address lines. Because the data & address lines are shared by all components, there must be means of controlling their use.

Typical control lines include:

Memory write, Memory read, I/O write, I/O read, Transfer ACK, Bus request, Bus grant, Interrupt request, Interrupt ACK, Clock & Reset

Q3: Differentiate the following:

Programming in the Hardware & Software

Ans:

Programming in Hardware:

the "program" is in the form

hardware is termed a hardware program.

Suppose we construct a general-purpose configuration of arithmetic & logic function. 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 & 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 have to provide a new sequence of codes. Each code is, in effect, an instruction & part of the hardware interprets each instruction & generates control signals.

B. Program flow of control without interrupt & with interrupt:

- In 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~~ fetch cycle & fetches the next instruction of the current program.

C) Disabled interrupt & nested interrupt processing.

- 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 & will be checked by the processor after the processor has enabled interrupts.

- Nested Interrupts:

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

a user program begain at
t=0. At t=10, a printer interrupt
occurs; user information is placed
in the system stack & execution
continues at the printer interrupt
service routine (ISR). While this
routine is still executing at t=15,
a communications interrupt occurs.

(Q 4) Solve Each of the following:

A) Ting hypothetical machine has two I/O instructions.

0011 = Load AC from I/O
0111 = Store AC to I/O

In these cases, the 12-bit address
identifies a particular I/O device.
show the program execution for the
following program:

Load AC from device 5.
Add content of memory location
940
Store AC to device 6.

Assume that the next value retrieved
from device 5 is 3 & that location
940 contains a value of 2.

Ans.

Memory (initial contents in hex): 300: 305; 301: 5940; 302: 7006; Step 1: 305 → IR; Step 2: 3 → AC Step 3: 5940 → IR; step 4: 3+2=5 → AC Step 5: 7006 → IR; Step 6: AC → Device 6.

B) The program execution of figure 02 is described in text using 6 steps. Expand this description to show MAR & MBR?

Ans

1.a) The PC contain 300, the address of the first instruction. This value is loaded in to the MAR.

b) The value in location 300 (which is the instruction with the value 1940 in hexadecimal) is loaded in MBR, and the PC is incremented. These two steps can be done in parallel.

c) The value in MBD is loaded in to the IR.

2) a) The address portion of the IR (940) is loaded into MAR

b) The value in location 940 is loaded into the MBR

c) the value in the MBR is loaded into AC.

3) a) the value in PC (301) into the MAR.
b) the value in the location 301 (which is the instruction with the value 5941) is loaded into the MBR and the PC is incremented.
c) the value in the MBR is loaded in the IR.
4) a) the address portion of IR (941) is loaded into MAR.
b) the value in location 941 is loaded into MBR.
c) the old value of AC & the value of location MBR are added & the result is stored in the AC.
5) a) the value in PC (302) is loaded into the MAR.
b) the value of location 302 (which is the instruction with the value 2941) is loaded into the MBR & the PC is incremented.
c) the value in MBR is loaded into IR.
6) a) the address portion of IR (941) is loaded into MAR.

(b) The value in the ac is loaded into the MBR
The value in MBR is stored in location 941

(c) Answer,

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

(b) 1: If the local address bus is 32 bits, the whole address can be transferred at once & decoded in memory. However 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 & 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 & a 32-bit program counter, unless on-chip segment register are used that may

work with a smaller program counter. If the instruction register to contain the whole instruction, it will have to be 32-bits long; if it will contain only the OP code (called the OP code register) then it will have to be 8 bits long.

(D) Answer:

Clock cycle =
1 = 125 ns. 8 MHz Bus cycle =
4 x 125 ns = 500 ns
2 bytes transferred every 500 ns
thus transfer rate 4 MBytes/sec

Doubling the frequency may mean adopting a new chip manufacturing technology (assuming equal instruction will have the same number of clock cycle); doubling the external data bus means wider (may be narrower on-chip data bus drivers/latches and modifications to the bus control logic. In the first case the speed of memory chips will also need to double (roughly) not to slow down - the +word length

the memory will have to double to be able to send/receive 32-bit quantities.

(E) Answer,

(a) During a single bus cycle, the 8 bit microprocessor transfers ~~two~~ one bytes while 16-bit microprocessor transfers 2 bytes. The 16-bit microprocessor has twice the data transfer rate.

(b) Suppose we do 100 transfers of operands & instructions, of which 50 are one byte long & 50 are two bytes long. The 8-bit microprocessor takes $50+(2\times16-50)$ $=150$ bus cycle for the transfer. The 16-bit microprocessor requires $50+50=100$ bus cycles. Thus the data transfer rate differs by a factor of 1.5.

(F) Answer:

A bus cycle take 0.25 μs so a memory cycle takes 1 μs. If both operand are even aligned it take 2 μs to fetch the two operands. If one is odd aligned the time required is 3 μs. If both are odd-aligned ~~rate~~ the time required is 4 μs.

(G) Answer)

Consider a maximum of 100 instruction & operands. On average, they consist of 20% 32-bit items, 40% 16-bit items & 40% 8-bit items the number of bus cycle required for the 16-bit microprocessor is (20×2)+40+40=120 for the 32-bit microprocessor. the number required is 100. This amounts to an improvement of 20/120 or about 17%.