İTÜ Computer Engineering Department
24.04.2014
Time: 1 hour 45 minutes
Emre HARMANCI
Feza BUZLUCA

COMPUTER ARCHITECTURE 2ND MIDTERM

QUESTION 1: (35 Points)
A CPU with 8-bit data bus, has an Interrupt Request input (**IRQ**) and an Interrupt Acknowledgement output (**INTA**). Both signals are active at "1". Normally, the CPU works with vectored interrupts and reads the interrupt vector number after the acknowledgement of the interrupt when it's Data acknowledgement input (**DACK**) is "1". The CPU also supports autovectored interrupts. After the acknowledgement of an interrupt request (**INTA=1**) if the vectored/autovectored input (**VA**) of the CPU is made "1", then it does not read a vector number and works in autovectored mode.
(**VA=0**): vectored; (**VA=1**): autovectored interrupts.

In this system there four interrupt sources (*A1*, *A2*, *B1*, *B2*) of two different types (*A*, *B*).

- Type ***A***: These devices (*A1*, *A2*) work with **vectored** interrupts. They have an Interrupt Request output (**IRQ**), an Interrupt Acknowledgement input (**INTA**), and 8-bit vector number output (**VN**). 20 ns after the interrupt has been acknowledged (**INTA=1**) the device outputs its vector number at the **VN** and removes its request (**IRQ=0**).
- Type ***B***: These devices work (*B1*, *B2*) with **autovectored** interrupts. They have an Interrupt Request output (**IRQ**), and an Interrupt Acknowledgement input (**INTA**). If the interrupt is acknowledged (**INTA=1**) the device removes its request (**IRQ=0**).

Priority (precedence) order of the devices: *A1* > *B1* > *A2* > *B2* (**Read carefully!**)

a) Design and draw the system with the CPU, 4 devices (*A1*, *A2*, *B1*, *B2*) and the priority interrupt controller. First, show the priority interrupt controller only as a box. Then design and draw the internal structure of the priority interrupt controller using logical gates.
b) Assume that the devices *A1* and *B1* assert their interrupt requests at the same time. Show step by step all the signals that are sent in the system until the requests of both devices has been fulfilled.
c) How does the CPU determine the start address of the interrupt service routine to be run if the interrupt source is a device of type ***A*** or of type ***B*** ?

QUESTION 2: (35 Points)
Instruction cycle of a CPU has the following 4 states (cycles) with the given durations.
1. Instruction fetch: 60 ns, 2. Instruction Decode: 20 ns, 3. Operand fetch: 60 ns, 3. Execution: 30 ns,
4. Interrupt (if necessary): 200 ns.
Note that the CPU accesses the memory in instruction fetch and operand fetch cycles but not in the decode and execution cycles. The CPU enters the interrupt cycle only if there are interrupt requests and housekeeping operations (saving return address, taking the vector address, etc.) take 200 ns.
Memory access time and I/O interface access times are both 50 ns.

In this system there is a 2-wire DMAC (*Direct Memory Access Controller*) that is configured to transfer words from the I/O interface to the memory using the **cycle-stealing** technique. The type of the DMAC is **fly-by** (Implicit). Data don't pass through DMAC.

The processor will run a program with 10 instructions and the DMAC is configured to transfer 10 bytes from the I/O interface to the memory. Assume that we start a clock (Time=0) when the CPU begins to run the program and as the CPU is in the instruction fetch cycle for the first instruction (Time=5ns) the DMAC attempts to start the data transfer.

a) When (Time =?) will the DMAC complete the transfer of the first byte? Why?
b) When (Time =?) will the CPU finish the first instruction? Why?
c) When (Time =?) will the DMAC complete the transfer of all 10 bytes? When (what Time) will the CPU complete the run of the entire program?

d) Assume that, instead of the DMA technique, **interrupt-driven I/O** technique is used to transfer 10 bytes from the I/O interface to the memory.
The interrupt service program transfers one word each time and it takes 500 ns (Housekeeping operations are not included).
Assume that, as the CPU is in the instruction fetch cycle for the first instruction (Time=5ns) the first interrupt request arrives from the I/O interface.
When (Time =?) can the first word transferred from the I/O interface to the memory? Why?

e) When (Time =?) will be all the 10 bytes transferred and the main program of 10 instructions finished when the **interrupt-driven** technique is used?

QUESTION 3: (30 Points)

a) Note: in RAID 4, disks operate independently (not synchronized). Large strips (blocks) are used.
Draw a RAID 4 system with total 4 disks (data + parity) and distribute 6 blocks (block 0 – block 5) over the disks.
Assume that the access time of a disk is **ta**.

i) How long does it take to read words from two blocks (for example block 0 and block 4) in two different disks? (One word from block 0, one word from block 4)

ii) How long does it take to update (write) words of two blocks (for example block 0 and block 4) in two different disks? Explain. (One word in block 0 and one word in block 4)

b) Answer the question in **a)** (i and ii) for the RAID 5 system.