**Lesson Summary and Revision Notes: Week 8**

Briefly Explain the **three stages** of the Fetch-Execute Cycle .

* **Fetch**: The CPU retrieve instructions from a system hard drive or solid-state drive (SSD) and store them in a register
* **Decode**: The CPU determines which system components are required in the execution of the instruction, outlining parameters for a successful execution.
* **Execute**: The CPU assigns the specific actions to the relevant system components in order to carry out the initial program instructions, processing the actual data.

Define what is an Interrupt ?

An interrupt is a signal sent from a device or from software to the processor. This will cause the processor to temporarily stop what it is doing and service the interrupt.

Give examples of what can cause an interrupt

Interrupts can be caused by, for example:

* a timing signal
* input/output processes (a disk drive is ready to receive more data)
* a hardware fault (an error has occurred such as a paper jam in a printer)
* user interaction (the user pressed a key to interrupt the current process, such as <CTRL><ALT><BREAK>)
* a software error that cannot be ignored (if an .exe file could not be found to initiate the execution of a program OR an attempt to divide by zero)

What registers are used in Fetch?

* PC (Program Counter)
* CIR (Current Instruction Register)
* MAR (Memory Address Register)
* MDR (Memory Data Register)

Explain the Fetch Execute cycle with the use of Interrupts



A special register called the interrupt register is used in the fetch-execute cycle. The Interrupt

Process of a Fetch Executive cycle is shown below:

1. At the next fetch-execute cycle, the interrupt register is checked bit by bit.
2. If an interrupt occurred during a previous cycle, the CPU would now service this interrupt or ignore it for now, depending on its priority.
3. Once the interrupt is serviced by the CPU, it stops its current task and stores the contents of its registers.
4. Control is now transferred to the interrupt handler (or interrupt service routine, ISR).
5. Once the interrupt is fully serviced, the register is reset and the contents of registers are restored.