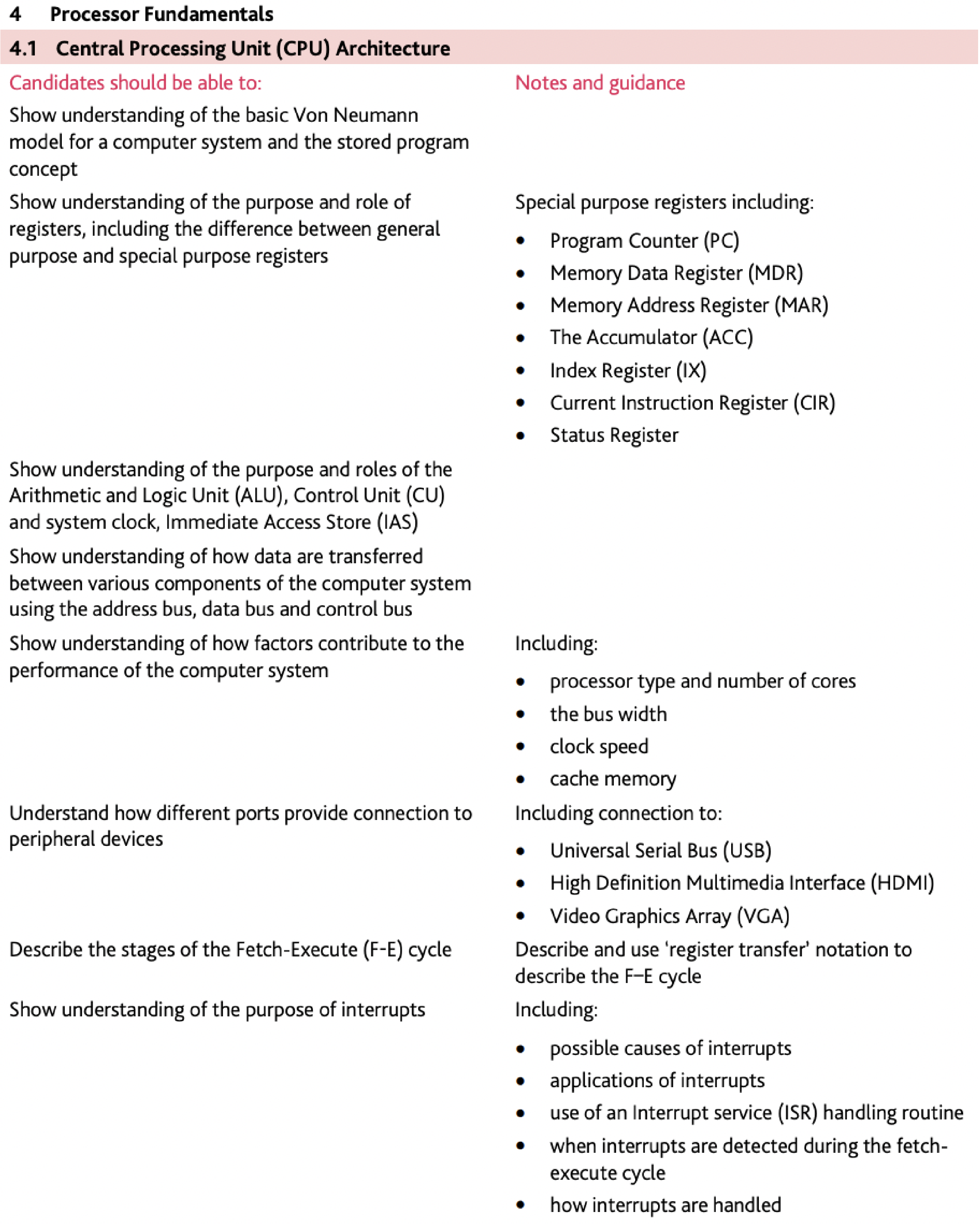
1. Central Processing Unit (CPU) Architecture



4 Processor Fundamentals

4.1 Central Processing Unit (CPU) Architecture

Candidates should be able to:	Notes and guidance
Show understanding of the basic Von Neumann model for a computer system and the stored program concept	
Show understanding of the purpose and role of registers, including the difference between general purpose and special purpose registers	Special purpose registers including: • Program Counter (PC) • Memory Data Register (MDR) • Memory Address Register (MAR) • The Accumulator (ACC) • Index Register (IX) • Current Instruction Register (CIR) • Status Register
Show understanding of the purpose and roles of the Arithmetic and Logic Unit (ALU), Control Unit (CU) and system clock, Immediate Access Store (IAS)	
Show understanding of how data are transferred between various components of the computer system using the address bus, data bus and control bus	
Show understanding of how factors contribute to the performance of the computer system	Including: • processor type and number of cores • the bus width • clock speed • cache memory
Understand how different ports provide connection to peripheral devices	Including connection to: • Universal Serial Bus (USB) • High Definition Multimedia Interface (HDMI) • Video Graphics Array (VGA)
Describe the stages of the Fetch-Execute (F-E) cycle	Describe and use 'register transfer' notation to describe the F–E cycle
Show understanding of the purpose of interrupts	Including: • possible causes of interrupts • applications of interrupts • use of an Interrupt service (ISR) handling routine • when interrupts are detected during the fetch-execute cycle • how interrupts are handled

- Von Neumann model

    - Single processor
    - Program consists of a sequence of stored instructions
    - Instructions and data are indistinguishable
    - … are stored in a contiguous block of main memory
    - Instructions are fetched and executed in sequence
- Stored-program concept
    - Program must be resident in main memory to be executed
    - Program consists of a sequence of instructions
    - Which occupy a contiguous block of main memory
    - Instructions and data are indistinguishable
    - Each instruction is fetched, decoded, then executed
    - Instruction fetch and data operation cannot occur at the same time
- Registers
    - General purpose registers
        - Hold data that is frequently used by the CPU
        - Can be used by the programmer when addressing the CPU
        - e.g. accumulator
    - Special purpose registers
        - Have a specific function within the CPU
        - Hold the program state
        - PC (Program Counter)
            - Stores the address of the next instruction to be fetched
        - MDR (Memory Data Register)
            - Stores the data in transit between memory and other registers
            - Holds the instruction before it is passed to the CIR
            - The location accessed is what is stored in MAR
        - MAR (Memory Address Register)
            - Stores the address of the memory location which is about to be accessed
        - ACC (Accumulator)
            - Stores the value of calculation by ALU
        - IX (Index Register)
            - Used to carry index addressing operation
        - CIR (Current Instruction Register)
          Stores the current instruction being executed and decoded

        - Status Register
            - Independent bits/flag
            - Each flag is set depending on an event
            - e.g. overflow, zero, negative, carry flag
- ALU (Arithmetic and Logic Unit)
    - Carries out arithmetic calculations
    - Carries out logic operations
    - Holds the result in ACC
- CU (Control Unit)
    - Sends and receives signals
    - Synchronizes operations
    - Control the execution of operations
    - e.g. input/output
- System clock
    - Sends out a number of pulses in a given time interval
    - Each processor operation takes several cycles to execute
    - The higher the clock frequency, the shorter the execution time and the better the performance
- IAS (Immediate Access Store)
    - Holds all the data and programs that the processor/CPU needs to access
- Bus
    - Address bus
        - Transfer address of memory and input/output position
        - Uni-directional
    - Data bus
        - Carries data between processor and memory
        - Bi-directional
    - Control bus
        - Transmit signals between the control unit
        - Dedicated bus since all timing signals are generated according to control signal
- Performance factors
    - Number of cores
        - Each cores processes one instruction per clock pulse

    - Multiple cores means that sequence of instructions can be split between them
    - … and so more than one instruction is executed per clock pulse
    - More cores decreases the time taken to complete task
  - Bus width
    - Bus width determines the number of bits that can be simultaneously transferred
    - Increasing bus width increases the number of bits that can be moved at one time
    - … hence improving processing speed as fewer transfers are needed
    - e.g. double the width of the data bus moves 2x data per clock pulse
  - Clock speed
    - Each instruction is executed on a clock pulse
    - … so the clock clock speed determines the number of instructions that can be run per second
    - The faster the clock speed the more instructions can be run per second
  - Cache memory
    - Cache memory uses SRAM while main memory uses DRAM
    - … so cache memory is faster than main memory
    - Cache memory stores frequently used instructions and data
    - When CPU reads memory, it first checks out cache and then moves on to main memory if the required data is not in cache memory
    - So this improve processor performance
- Ports
  - USB (Universal Serial Bus)
    - An asynchronous serial data transmission method
    - The computer automatically detects that a device is present (due to a small change in the voltage level)
    - The device is automatically recognized, and the appropriate device driver is loaded up
    - If a new device is detected, the computer will look for the device driver which matches the device. If this is not available, the user is prompted to download the appropriate software
  - HDMI (High Definition Multimedia Interface)

        - The computer automatically detects that a device is present (due to a small change in the voltage level)
        - Handshake between the display and the computer
        - The display sends meta data about itself (resolution, color depth, etc.)
    - VGA (Video Graphics Array)
        - The computer automatically detects that a device is present (due to a small change in the voltage level)
        - Handshake between the display and the computer
        - The display sends meta data about itself (resolution, color depth, etc.)
- F-E (Fetch-Execute) cycle
    - Fetch
        - The next instruction is fetched from the memory address in PC
        - This instruction in stored in CIR
        - The PC is incremented by 1
        - The instruction is decoded
    - Execute
      - The processor passes the decoded instruction as a set of control signals to the appropriate components within the computer system
      In register transfer notation:
    1. MAR←[PC] contents of PC copied into MAR
    2. PC←[PC]+1 PC is incremented by 1
    3. MDR←MAR data stored at address shown in MAR is copied into MDR
    4. CIR←[MDR] contents of MDR copied into CIR
- Interrupts
    - At the start oof each F-E cycle the processor checks for interrupts
    - Check if an interrupt flag is set in status register
    - Processor identifies source of interrupt
    - Processor checks priority of interrupt
    - If interrupt priority is high enough
    - Processor saves current contents of registers
    - Processor calls ISR (Interrupt Service Routine)
    - Address of ISR is loaded into PC
    - When servicing of interrupt complete, processor restores registers

- Processor continues with next F-E cycle