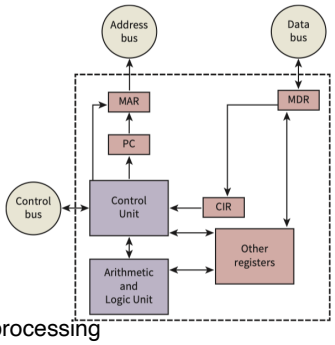
ALevel CS C05 Processor fundamenta

Von Neumann features:

1. There is a processor - CPU(Center processing unit)
2. The processor has direct access to memory
3. The memory contains a 'stored program' and the data required by the program
4. The stored program consists of individual instructions
5. The processor executes instructions sequentially



ALU(Arithmetic Logic Unit): arithmetic or logic processing requirements of the instructions in a running program

Control Unit:

1. controlling the flow of data throughout the processor and the rest of the while computer system
2. ensuring that program instructions are handle correctly

Internal clock: controls the cycles of activity within the processor
System clock: controls the cycles of activity outside the processor

Registers

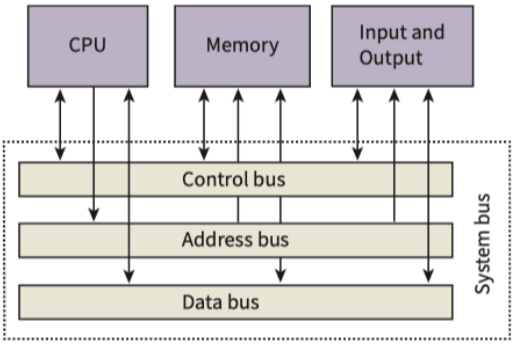
feature:

1. placed very close to ALU, allow very short access time
2. limited storage capacity

Accumulator(ACC): a general-purpose register that **stores a value** before and after the execution of an instruction by the ALU
Current instruction register(CIR): Stores the current instruction while it is being decoded and executed
Index register(IX): Stores a value; only used for **indexed addressing**
Memory address register(MAR): Stores the address of a memory location or an I/O component which is about to have a value read from or written to
Memory data register(MDR): Stores data that has just been read from memory or is just about to be written to memory
Program counter(PC): Stores the address of where the next instruction is to be read from
Status register(SR): Contains bits that are either set or cleared which can be referenced individually

BUS

Address Bus: a component that carries an address. This can be to the memory controller to identify a location in memory which is to be read from or written to or it can be to the I/O system to identify the source or destination of the data
The address bus is a '**one-way street**'. It can only be used to send an address to a memory controller or an I/O controller.
Data Bus: a component that can carry data from the processor to the memory or to an output device or can carry data from the memory or from an input device
The data bus is **two-way(bidirectional)**.
Control Bus: transmit a signal from the control unit to any other system component or transmits a signal to the control unit.
The control bus is a **bidirectional bus**.
A major use of control bus is to **carry timing signals**.



Factors contributing to system performance

Processor clock speed: One clock cycle defines the shortest possible time that any action can take.
Cores: Performance improves with increasing number of cores.
Cache memory: Cache memory is the fastest component of the IAS. Performance improves with increased storage size for the cache and with increased rate of access.
Address bus width: Special techniques are used when the storage capacity of the memory is too large for direct addressing. Their use affects system performance.

I/O port

Each I/O device is connected to an interface called a port.
Each port is connected to the I/O or device controller.
Universal Serial Bus(USB):
The **plug-and-play** concept was only fully realised by the creation of the USB standard.
Specialised multimedia ports:
VGA port: provides high-resolution screen display. **not support audio component**
HDMI port: allow the transmission of high-quality video **including the audio component.**

The fetch-execute(F-E) cycle (fetch, decode and execute cycle)

1. The **PC holds the address** of the next instruction to be loaded.
2. address in the PC transferred within the **CPU to MAR**
3. the instruction held in the address pointer to by the MAR is **fetched into MDR**
4. The instruction stored in the MDR is transferred within the CPU to the **CIR**
5. In the final step the **PC is incremented by 1**.

The clock cycle is the one controlled by the system clock which will have settings that allow one data transfer from memory to take place in the time defined for one cycle.

Register transfer notation:
MAR ← [pc]
PC ← [pc] + 1; MDR ← [[MAR]]
CIR ← [MDR]

The content of the MAR is an address; it is the content of that address which is being transferred to the MDR.

Interrupt handling
reasons for an interrupt to be generated:

1. a fatal error in a program
2. a hardware fault
3. a need for I/O processing to begin
4. user interaction
5. a timer signal

interrupt handle step:

1. The content of PC and other register are stored safe in the memory
2. The appropriate interrupt handler or ISR program is initiated by loading its start address into the program counter
3. check if further interrupt need handling.
4. Further interrupts are dealt with by repeated execution of the ISR program.
5. If there are no further interrupts, the safely stored contents of the registers are restored to the CPU and the originally running program is resumed.