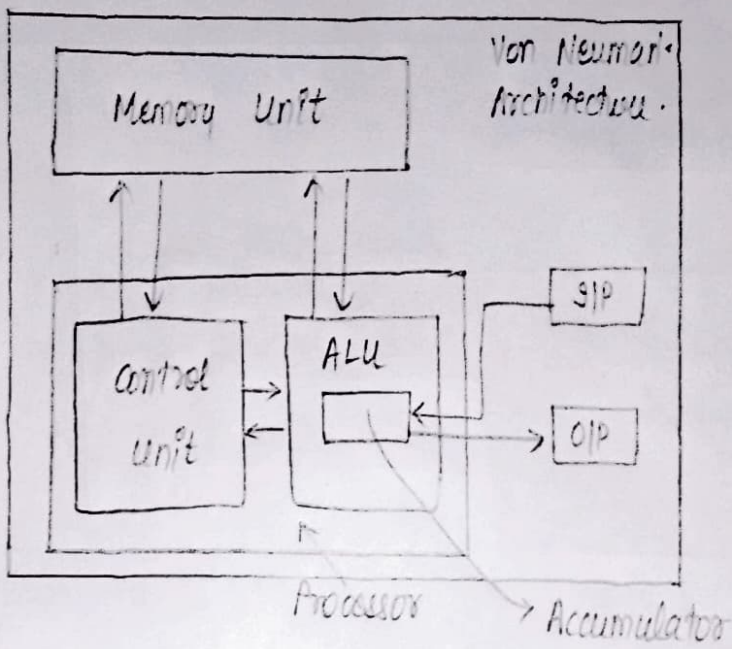
Design & Implementation of Control Unit:-

Control Unit :→

Control Unit is the part of the Computer's central Processing Unit (CPU), which directs the operation of the processor.

→ It was included as part of Von Neumann Architecture by John Von Neuman. It is the responsibility of Control Unit to tell computer's memory, arithmetic / logic unit & Input & output devices how to respond to the instructions that have been sent to the processor.



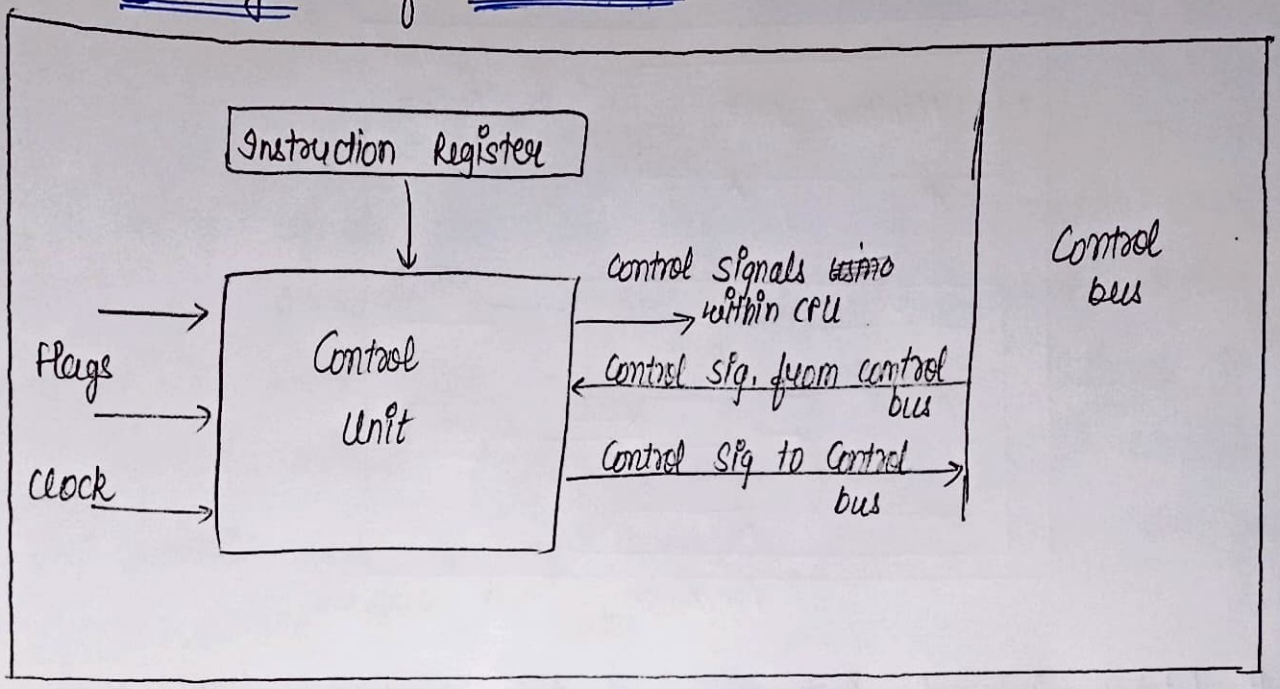
→ It fetches internal instructions of the programs from main memory to the processor instruction register (IR), & based on IR's content, the control unit generates a control signal that supervises the execution of these instructions.

→ Basically, Control unit controls all the I/P, O/P & other memory flow operations.

Working of Control Unit :→

* A control unit receives input information to which it converts into controls signals, which are then sent to central processor.
* The computer's processor then tells the attached hardware what operations to perform.
* The functions that a control unit performs are dependent on the type of CPU because the architecture of CPU varies from manufacturer to manufacturer

Block diagram of the Control Unit :→



Functions of Control Unit :→

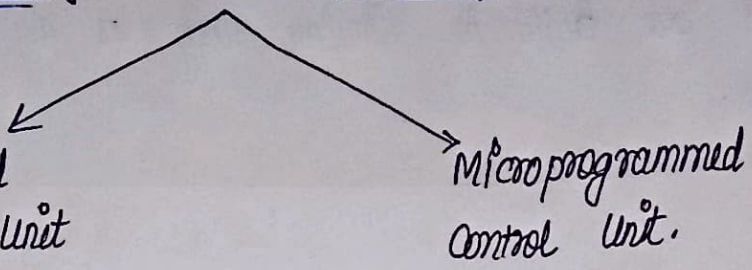
1) It coordinates the sequence of data movement into, out of & b/w a processor's many sub-units.
2) It interprets instructions.
3) It controls data flow inside the processor.

4) It receives external instructions or commands to which it converts to sequence of control signals.

5) It controls many execution units (ALU, data buffer & registers) contained within a CPU.

6) It also handles multiple tasks, such as fetching, decoding, execution handling & storing results.

⊕ There are two methods of Designing & Implementing a CU or we can say there are two types of CU :→



① Hardwired control Unit :→

→ The control signals that are important for instruction execution control are generated by specially designed hardware logical circuits, in which we can not modify the signals generation method without physical change of the circuit structure.

→ The operation code of an instruction contains the basic ~~unit~~ data for control signal generation.

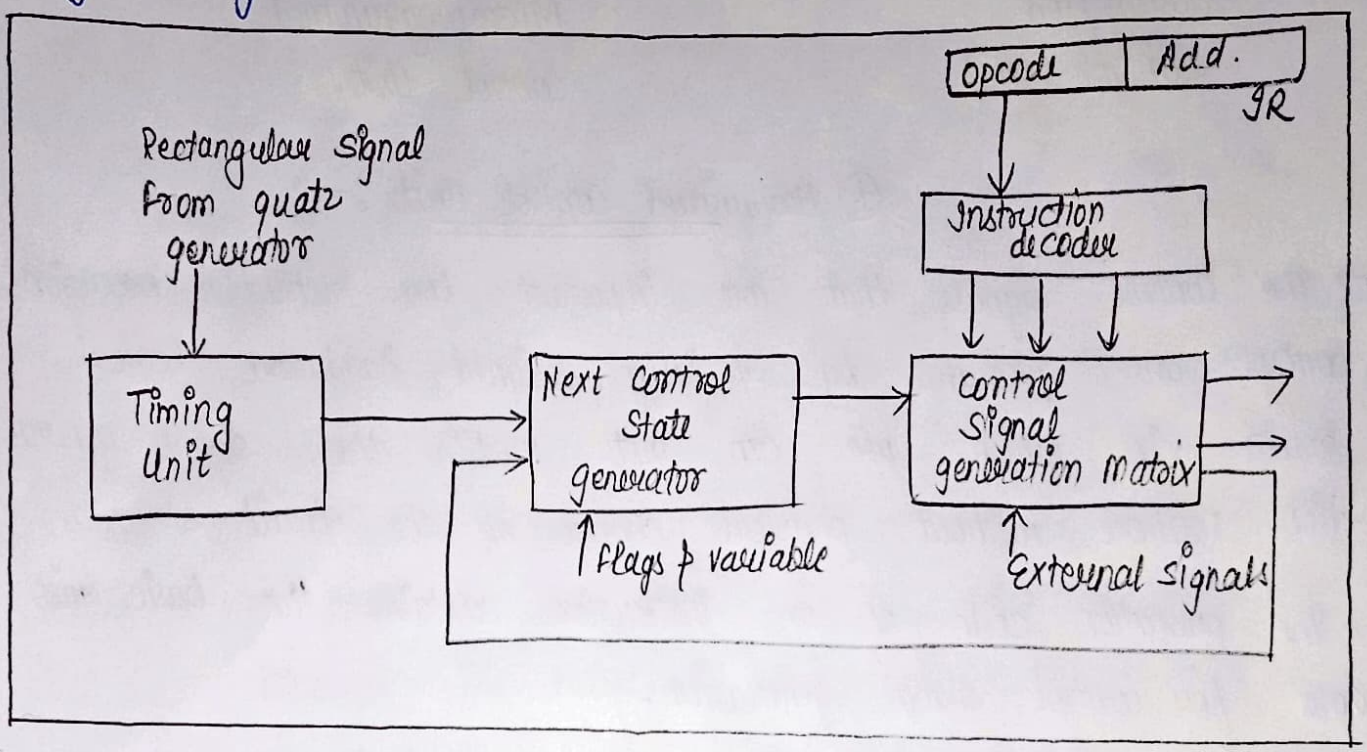
→ In Instruction decoder, the operation code is decoded. The Instruction decoder constitutes a set of many decoders that decode different fields of Instruction opcode.

→ As a result, few ~~outline~~ output lines going out from the Instruction decoder obtains active signal values.

These output lines are connected to the inputs of the matrix that generates control signals for executive units of the computer.

→ This matrix implements logical combinations of the decoded signals from the instruction opcode with the outputs from the matrix that generates signals representing consecutive control unit states & with signals coming from the outside of the processor (interrupt signals).

→ These matrices are build in similar way as a programmable logic arrays.



→ control signals for an instⁿ. execution have to be generated not in a single time point but during entire time interval that corresponds to instruction execution cycle.

① Description of above block diagram :-

1. A no. of signals generated by control signal generator matrix are sent back to inputs of next control signal generator matrix. (CSGM).
2. This matrix combines these signals with the timing signals, which are generated by timing unit based on the rectangular patterns usually supplied by quartz generator.
3. When a new Instn arrives at control unit, the CUs is in the initial state of new Instn. fetching.
4. Instn decoding allows the CU enters the first relating execution of the new Instn, which lasts as long as timing signals & other I/P signals as flags & state info. of computer remain unaltered.
5. A change of any of earlier mentioned signals stimulates the change of the control unit state.
This causes that a new respective I/P is generated for the control signal GM.
6. When an interrupt appears, the CU takes entry into a next control state that is the state concerned with the reactions to this interrupt. The value of flags & state variables of computer are used to select suitable states for Instn execution cycle.

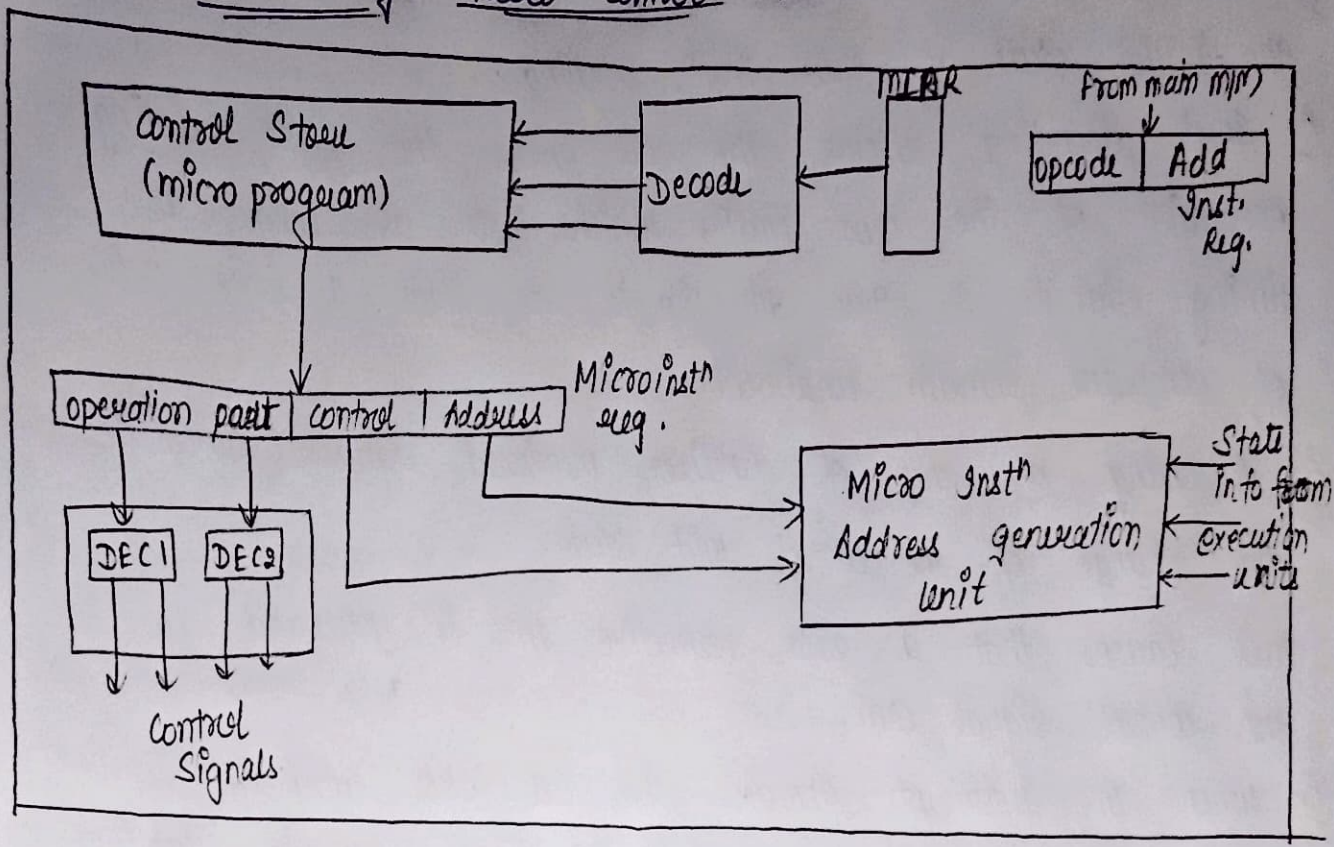
(2) Micro programmable control Unit :→

→ It contains control store that is used for storing words containing encoded control signals mandatory for Instn. execution.

In this CU, subsequent Instn. words are fetched into the MBR register in a normal way.

→ Opcode of each Instn is not directly decoded to enable immediate control signal generation but it comprises the initial address of a microprogram contained in the control store.

~~~→ with a single-level control store :→



→ Instn opcode from instn register is sent to the control store address register. Based on that address, 1st micro instn. of a micro program that interprets execution of this instn is read to the microinstn register.

→ This micro instn. contains in its operation part encoded control signals, normally as few bit fields. In a set microinstn field decoder, the fields are decoded.

→ The micro instn also contains the address of next micro instn of given instn microprogram & a control field used to

control activities of micro Instⁿ address generator.

→ In micro Instⁿs. along with conditional addressing mode, this address is refined by using processor condition flags that represent the status of computations in current program.

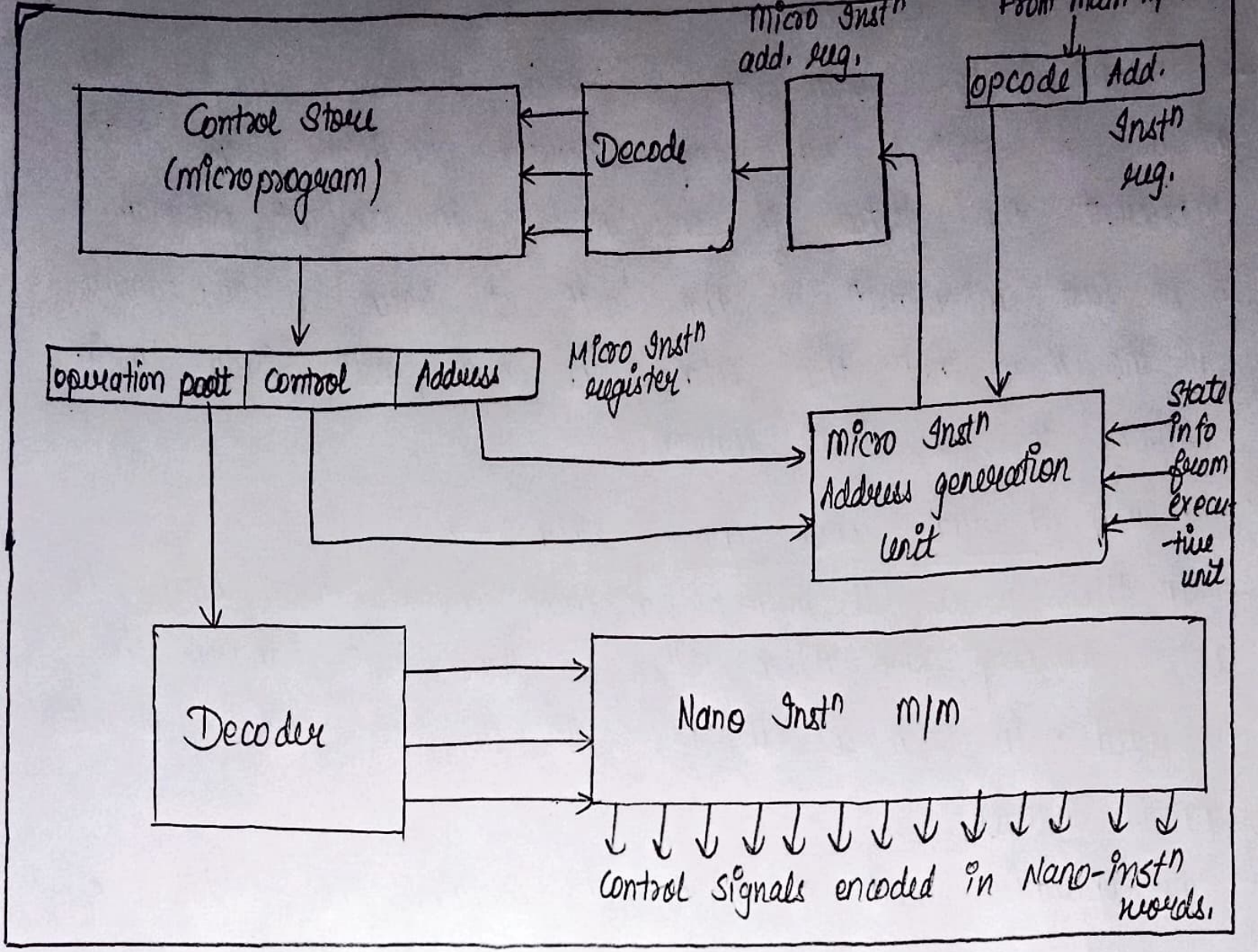
→ The last micro Instⁿ in the Instⁿ of given micro program is the micro Instⁿ that fetches the next Instⁿ from main memory to the Instⁿ register.

~~~ with a two-level control store :→

→ In this, in a CU with a two level control store, besides the control memory for micro Instⁿs, a Nano-Instⁿ memory is included. In such a CU, micro Instⁿ do not contain encoded control signals.

→ The operation part of micro Instⁿ contains the address of the word in Nano-Instⁿ m/m, which contains encoded control signals.

→ This Nano-Instⁿ m/m contains all combinations of control signals that appear in micro program that interprets the complete Instⁿ set of a given computer, written once in form of Nano-instⁿ.
~~~


Micro Instn add. reg.
opcode
Add.
Instn reg.
Control Store (microprogram)
Decode
operation part
Control
Address
Micro Instn register
micro Instn Address generation unit
State info from execu-tive unit
Decoder
Nano Instn m/m
Control signals encoded in Nano-instn words.

Hardwired Control Unit	Microprogrammed Control Unit
Hardwired control unit generates the control signals needed for the processor using logic circuits	Microprogrammed control unit generates the control signals with the help of micro instructions stored in control memory
Hardwired control unit is faster when compared to microprogrammed control unit as the required control signals are generated with the help of hardware's	This is slower than the other as micro instructions are used for generating signals here
Difficult to modify as the control signals that need to be generated are hard wired	Easy to modify as the modification need to be done only at the instruction level
More costlier as everything has to be realized in terms of logic gates	Less costlier than hardwired control as only micro instructions are used for generating control signals
It cannot handle complex instructions as the circuit design for it becomes complex	It can handle complex instructions
Only limited number of instructions are used due to the hardware implementation	Control signals for many instructions can be generated
Used in computer that makes use of Reduced Instruction Set Computers(RISC)	Used in computer that makes use of Complex Instruction Set Computers(CISC)