TOPIC: Micro-programmed Control Unit (Lecture 13)

By – Dr Anchal Thakur (Asst. Professor, DICE)

Introduction

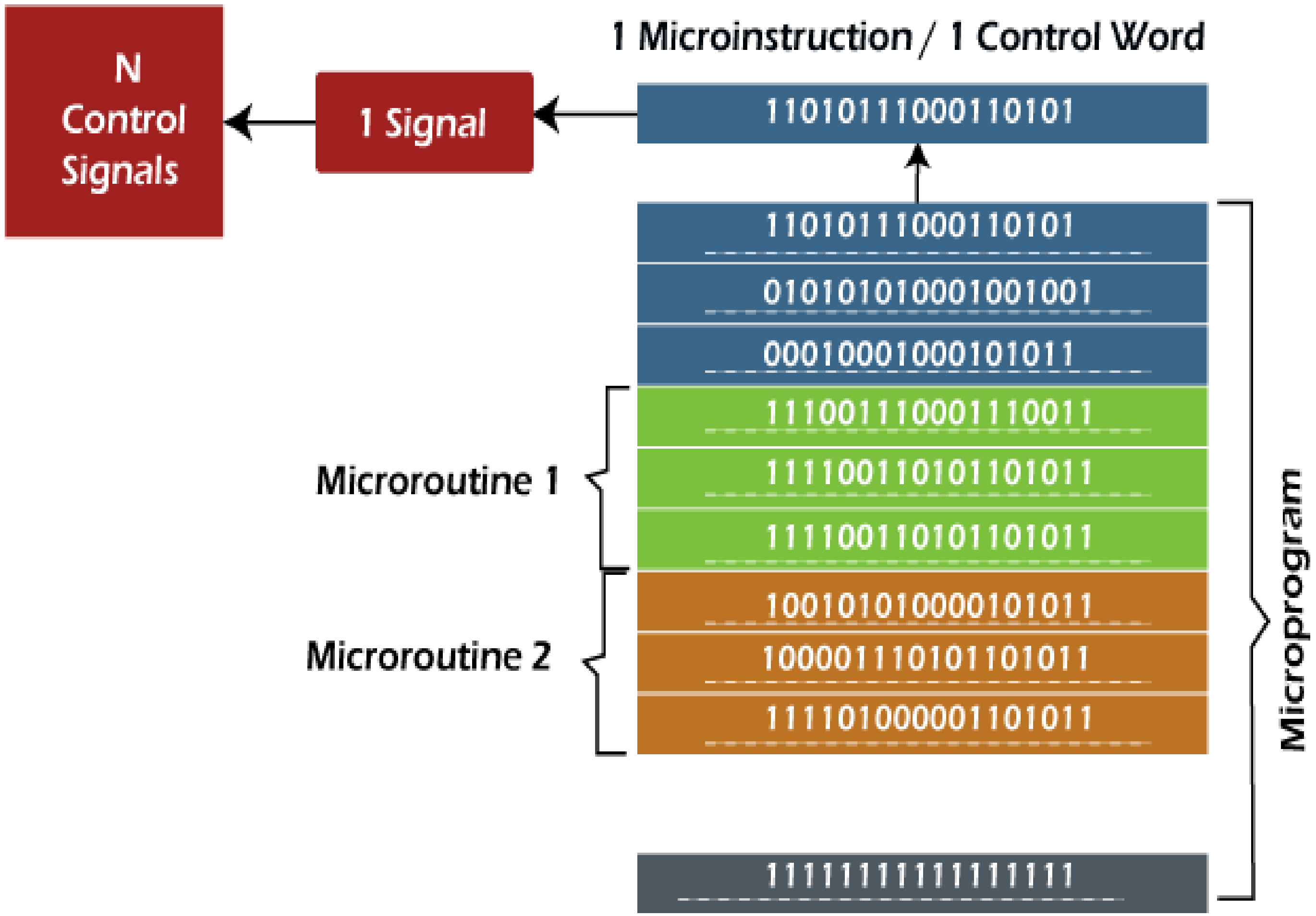
The CPU usually has two main systems: **control unit** (CU) and **arithmetic and logic unit** (ALU). The control unit (CU) is used to synchronize the tasks with the help of sending timings and control signals. On the other hand, mathematical and logical operations can be handled with the help of ALU. Micro programmed control units and hardwired control units can be called two types of control units. We can execute an instruction with the help of these two control units.

In the **hardwired control unit**, the execution of operations is much faster, but the implementation, modification, and decoding are difficult. In contrast, implementing, modifying, decoding **micro-programmed control units** is very easy. The micro-programmed control unit is also able to handle complex instructions. With the help of control signals generated by micro-programmed and hardwired control units, we are able to fetch and execute the instructions.

Micro-programmed Control Unit

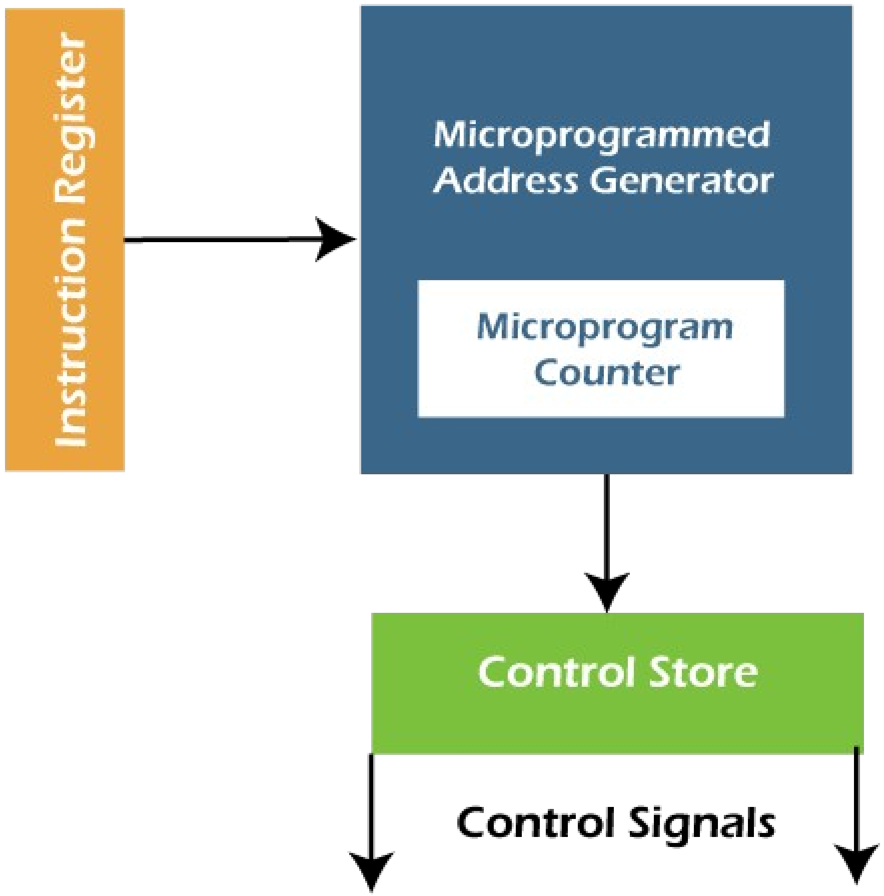
A micro-programmed control unit can be described as a simple logic circuit. We can use it in two ways, i.e., it is able to execute each instruction with the help of generating control signals, and it is also able to do sequencing through microinstructions. It will generate the control signals with the help of programs. At the time of evolution of CISC architecture in the past, this approach was very famous. The program which is used to create the control signals is known as the "Micro-program". The micro-program is placed on the processor chip, which is a type of fast memory. This memory is also known as the control store or control memory.

Microprogramed control unit



Continue..

A micro-program is used to contain a set of microinstructions. Each microinstruction or control word contains different bit patterns. The n bit words are contained by each microinstruction. On the basis of the bit pattern of a control word, every control signals differ from each other.



Microprogrammed Control Unit Organization

Organization of Micro-programmed CU

The flow of instruction execution with the help of instruction execution steps, which are described as follows:

- **Instruction** fetch is the **first step**. In this step, the instruction is fetched from the IR (Instruction Register) with the help of a Microinstruction address register.
- **Decode** is the **second step**. In this step, the instructions obtained from the instruction register will be decoded with the help of a microinstruction address generator. Here we will also get the starting address of a micro-routine. With the help of this address, we can easily perform the operation, which is mentioned in the instruction. It will also load the starting address into the micro-program counter.
- **Increment** is the third step. In this step, the control word, which corresponds to the starting address of a micro-program, will be read. When the execution proceeds, the value of the micro-program counter will be increased so that it can read the successive control words of a micro-routine.
- **End bit** is the fourth step. In this step, the microinstruction of a micro-routine contains a bit, which is known as the end bit. The execution of the microinstruction will be successfully completed when the end bit is set to 1.
- This is the last step, and in this step, the micro-program address generator will again go back to **Step 1** so that we can fetch a new instruction, and this process or cycle goes on.

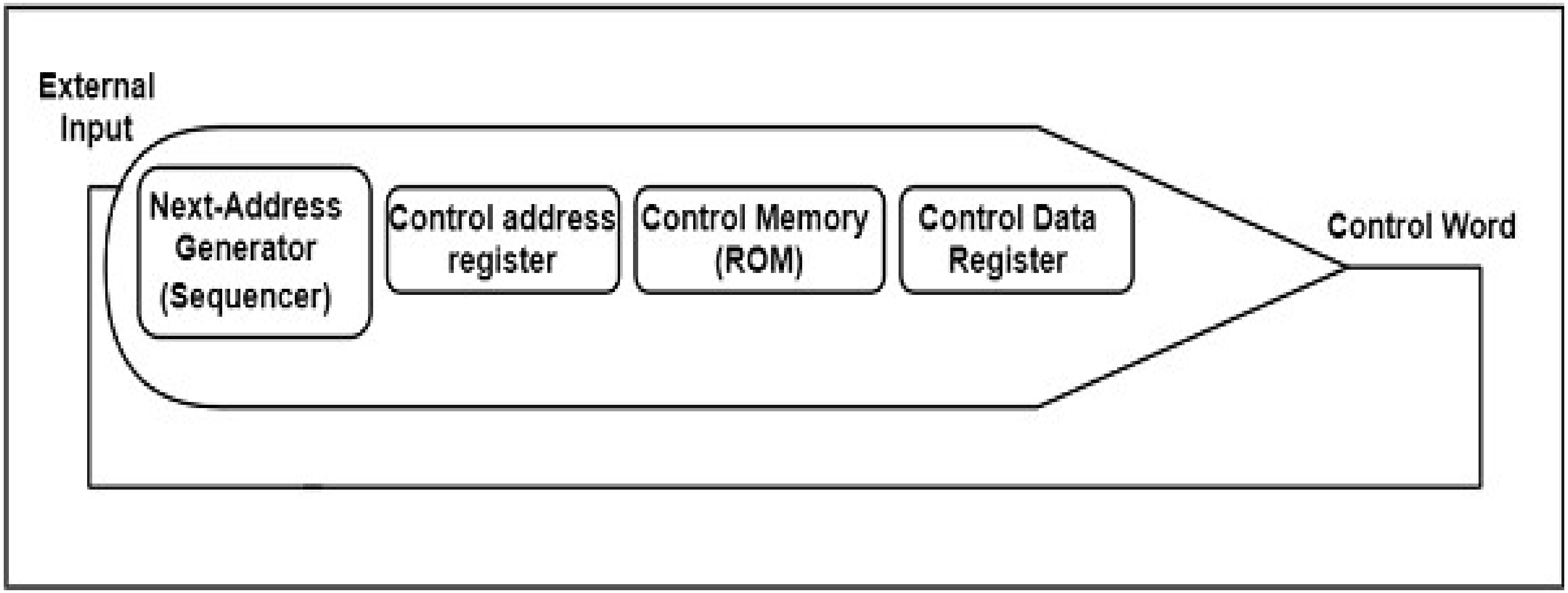
Control Memory

- A control memory is a part of the control unit. Any computer that involves microprogrammed control consists of two memories. They are the main memory and the control memory. Programs are usually stored in the main memory by the users. Whenever the programs change, the data is also modified in the main memory. They consist of machine instructions and data.
- The control memory consists of microprograms that are fixed and cannot be modified frequently. They contain microinstructions that specify the internal control signals required to execute register micro-operations.
- The machine instructions generate a chain of microinstructions in the control memory. Their function is to generate micro-operations that can fetch instructions from the main memory, compute the effective address, execute the operation, and return control to fetch phase and continue the cycle.

Continue..

- The figure shows the general configuration of a microprogrammed control organization.

Micro Programmed Control Organization



Addressing Sequencing in Computer Organization

The control memory is used to store the microinstructions in groups. Here each group is used to specify a routine. The control memory of each computer has the instructions which contain their micro-programs routine. These micro-programs are used to generate the micro-operations that will be used to execute the instructions. Suppose the address sequencing of control memory is controlled by the hardware. In that case, that hardware must be capable to branch from one routine to another routine and also able to apply sequencing of microinstructions within a routine. When we try to execute a single instruction of computer, the control must undergo the following steps:

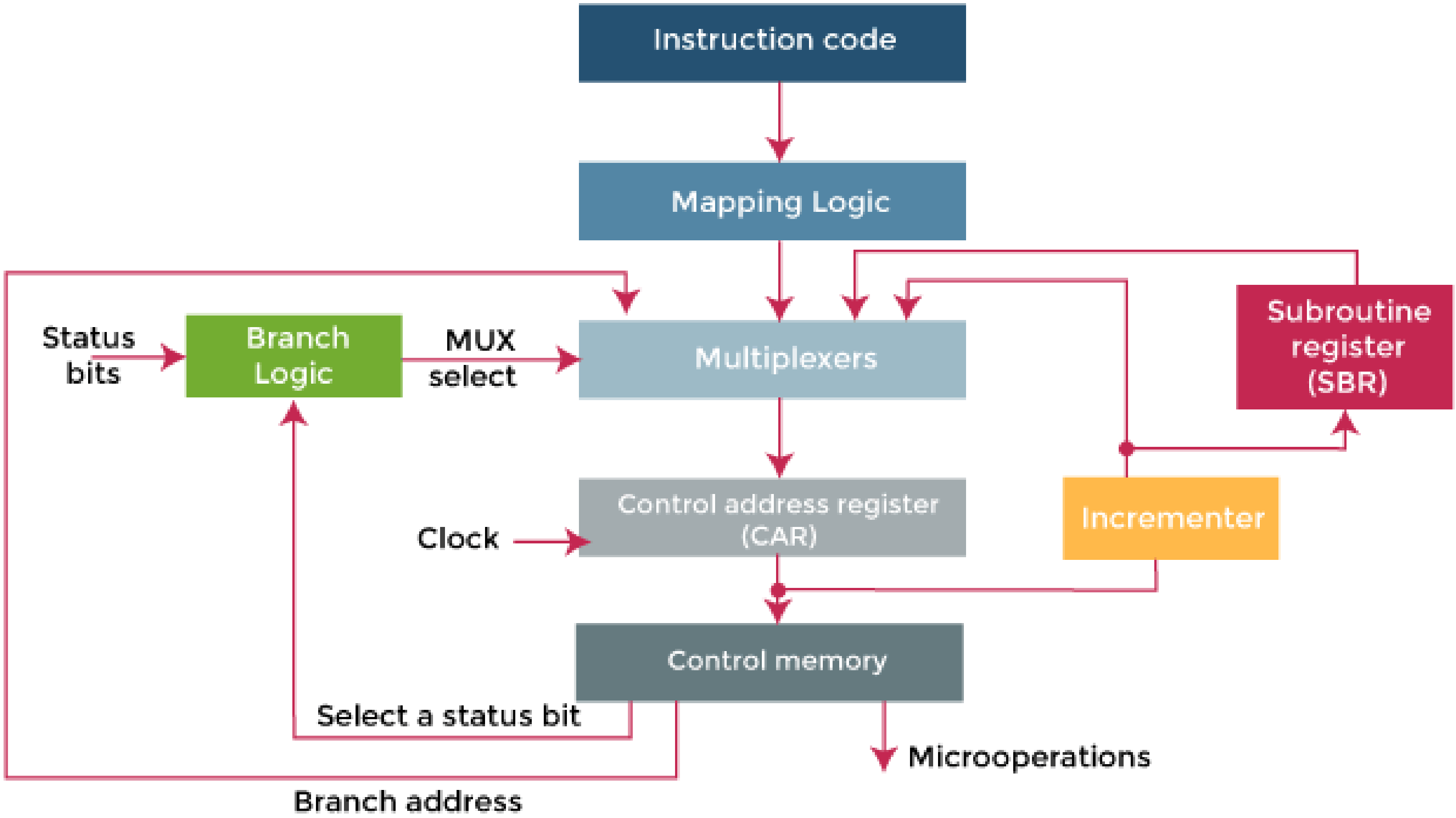
- When the power of a computer is turned on, we have to first load an initial address into the CAR (control address register). This address can be described as the first microinstruction address. With the help of this address, we are able to activate the instruction fetch routine.
- Then, the control memory will go through the routine, which will be used to find out the effective address of operand.
- In the next step, a micro-operation will be generated, which will be used to execute the instruction fetched from memory.

Continue..

The control memory required the capabilities of address sequencing, which is described as follows:

- On the basis of the status bit conditions, the address sequencing selects the conditional branch or unconditional branch.
- Addressing sequence is able to increment the CAR (Control address register).
- It provides the facility for subroutine calls and returns.
- A mappings process is provided by the addressing sequence from the instructions bits to a control memory address.

Continue..



Selection of Address for Control Memory

Conditional Branching

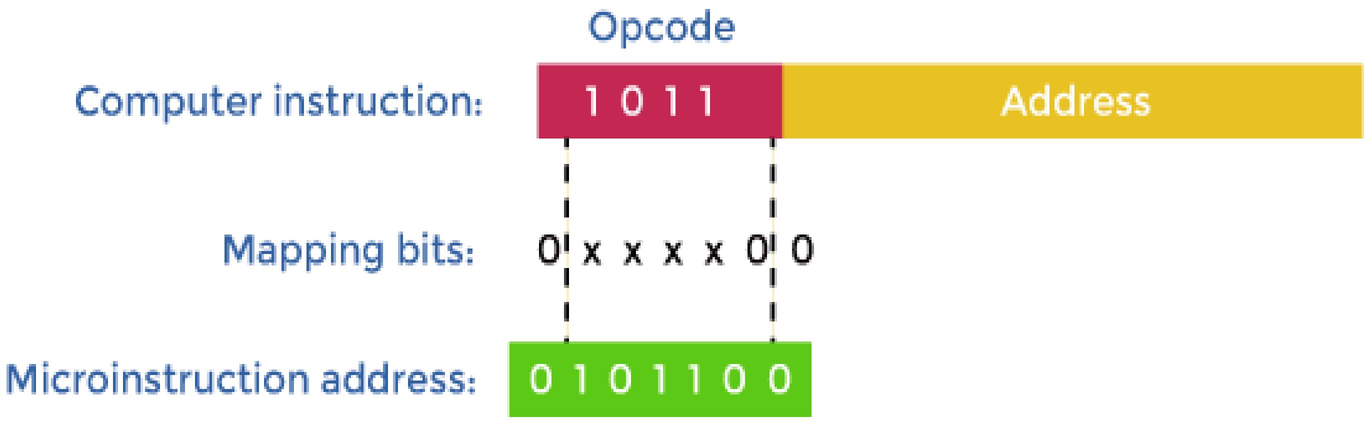
- In the above diagram, the **branch logic** is used to provide the decision-making capabilities in the control unit. There are **special bits** in the system which is described by the status conditions. These bits are used to provide the parameter information such as mode bits, the sign bit, carry-out, and input or output status.

- If these status bits come together with the microinstruction field, they are able to control the decision of a conditional branch, which is generated in the branch logic. Here the microinstruction field is going to specify a branch address. The multiplexer is used to implement the **branch logic hardware**. If the condition is met, it will be branch to the initial address. Otherwise, it will increment the address register.

- If we load the branch address into the control address register from the control memory, we are able to implement the unconditional branch microinstruction. If the condition is true, it will go to the branch, which is referred to as the address from the next address field of the current microinstruction. Otherwise, it will fall through. There are various types of conditions that need to be tested: Z(zero), C(carry), O(overflow), N(negative), etc.

Mapping of Instructions

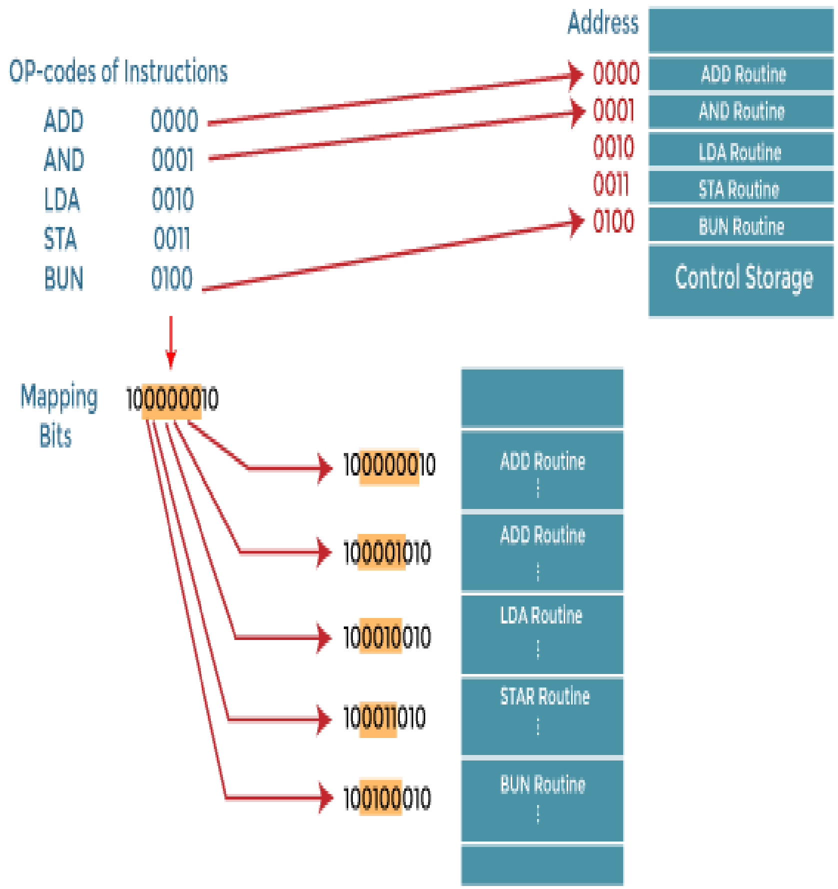
- In the control memory, if the microinstruction specifies a branch to the first work, in this case, there will be a special type of branch. Here an instruction contains their micro-program routine. For this special branch, the status bits will be the bits in the operation code, which is the part of instruction.



Mapping from instruction code to microinstruction address

Continue..

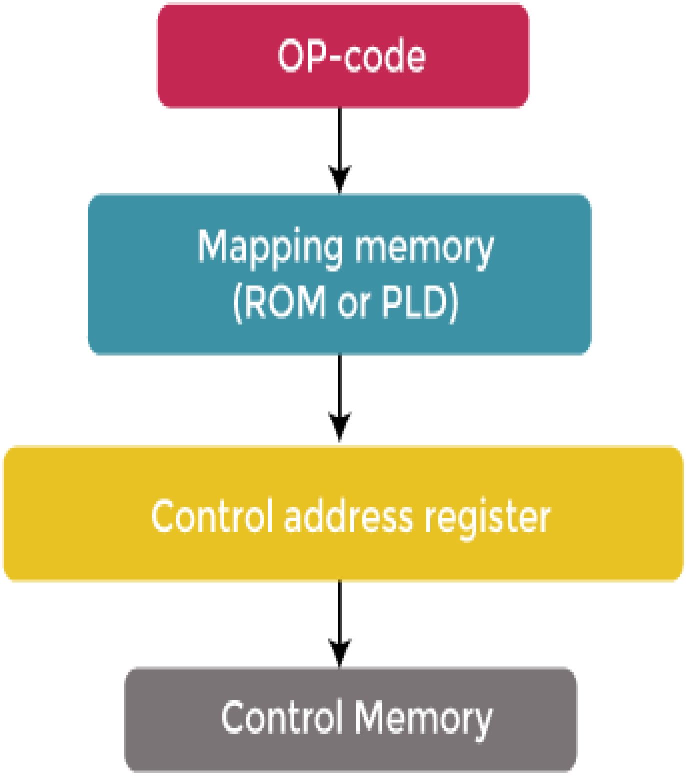
- The above image shows a type of easy mapping process which are going to convert the 4-bit operation code into the 7-bit address for control memory. In the mapping process, the 0 will be placed in the most significant bit of address. After that, the four operation code bits will be transferred. Lastly, the two least significant bits of CAR will be cleared.
- With the help of this process, a micro-program will be provided to each computer instruction. The micro-program contains the capacity of four microinstructions. If less than four microinstructions are used by the routine, the location of unused memory can be used for other routines. If more



Direct Mapping

Continue..

- This concept can be extended to a more general mapping rule with the help of PLD (Programmable logic device) or ROM (Read-only memory).
- The above image shows the mapping of address of microinstruction from the OP-code of an instruction. In the execution program, this microinstruction is the starting microinstruction.



Mapping Function Implemented by ROM and PLD

Subroutine

- Subroutines can be referred to as programs that are used to accomplish a particular task by the other routines. With the help of employing subroutines, we can save the microinstructions. These subroutines use the common sections of microcode, such as effective address computation.
- The main routine is able to get the address for the return with the help of a subroutine register. In another word, we can say that it becomes a source to transfer the address to a main routine. The register file is used to store the addresses for subroutines. These register files can be structured in a way that the register will be organized in the 'Last in first out' (LIFO) stack.