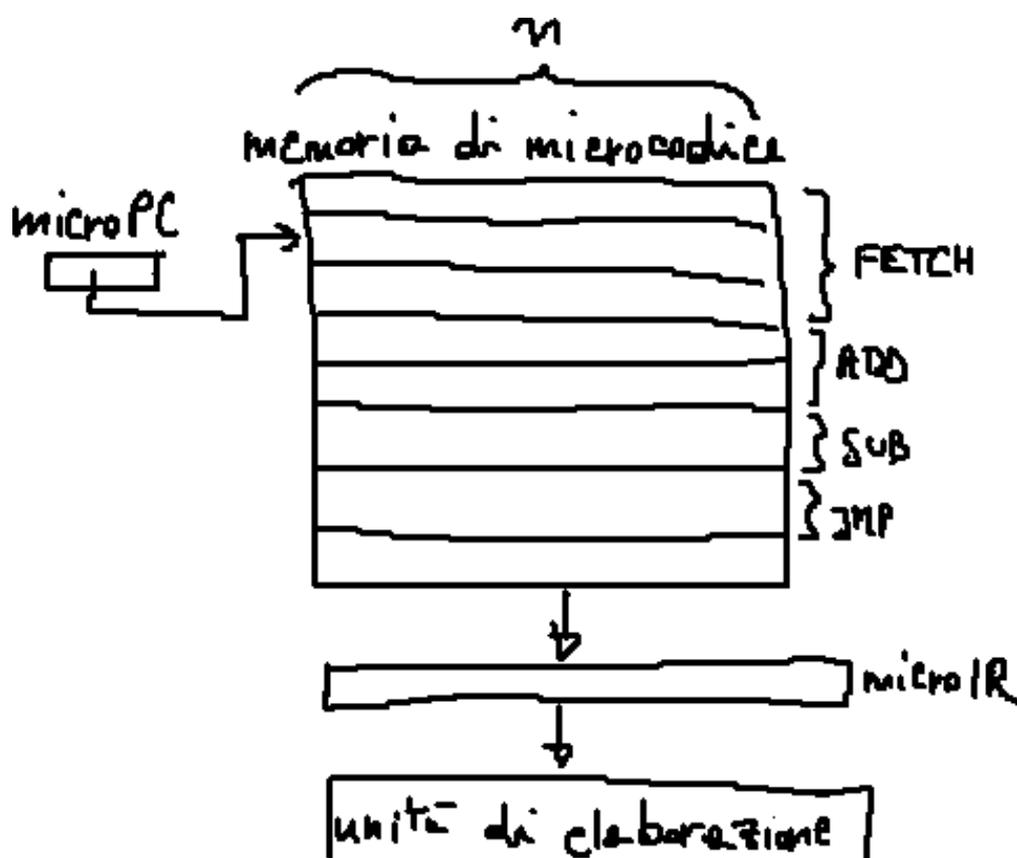


## Horizontal Microprogramming

In horizontal microprogramming, each bit is identified specifically with a single control point, which indicates that the corresponding micro-operation is to be executed. Since each microinstruction is capable enough to control several resources simultaneously, it has the potential advantage of more efficient hardware utilization and in addition, it requires smaller number of microinstructions per microprogram. It allows higher degree of parallelism with a minimum amount of encoding and separate control fields. However, developing microprograms that use resources optimally or efficiently is a complex task. Horizontal microprogramming offers great flexibility because each control bit is independent of each other. It has greater length so it typically contains more information than vertical microinstructions.



## Vertical Microprogramming

Vertical microprogramming employs a variable format and higher degree of encoding, as opposed to horizontal microprogramming. It not only shortens the length of the microinstruction, but also prevents the increasing memory capacity from directly affecting the microinstruction length. Each vertical microinstruction generally represents a single micro-operation. A code is used for each micro-operation to be performed and the decoder translates

the code into individual control signals. Because only the micro-operation to be performed is specified, the microinstruction fields are fully utilized. Plus vertical microprograms are easier to write than their horizontal counterparts. Vertical microinstruction resembles the conventional machine language format comprising one operation and a few operands. It is consequently easy to use for microprogramming. It generally consists of four to six fields that require approximately 16 to 32 bits per instruction.

---

## **Difference between Horizontal and Vertical Microprogramming**

---

### **Encoding**

---

Vertical microprogramming employs a variable format and a higher degree of encoding, as opposed to horizontal microprogramming. In vertical microprogramming, the control bits are encoded with each code being used for each action to be performed and an instruction decoder decodes the code into multiple control signals. On the contrary, horizontal microprogramming involves horizontal microinstructions that use no encoding at all. They represent each control bit in the datapath assigned with a separate bit in the microinstruction format. Every bit in the control field is attached to a control line.

---

### **Sequence**

---

Horizontal microprogramming generally follows a sequential approach to specify the next microinstruction in a microprogram, similar to conventional machine language format. Each bit is identified specifically with a single control point, which indicates that the corresponding micro-operation is to be executed. Special conditional and unconditional branch microinstructions are then required to break the sequence. Vertical microprogramming may use a relative addressing scheme in which a few bits are required to specify a relative forward or a backward jump. This requires address computation at every step.

---

### **Design**

---

– Vertical microprograms have a better code density which is beneficial for the size of the control store. Vertical microinstruction resembles the conventional machine language format comprising one operation and a few operands. Each vertical microinstruction represents a single micro-operation, while operands may specify the data sink and source. Horizontal microprograms, on the other hand, generally represent multiple micro-operations that are

executed at the same time. In extreme cases, each horizontal microinstruction controls several hardware resources simultaneously.

---

## **Flexibility**

---

– Horizontal microprograms offer improved flexibility because each control bit is independent of each other. It has greater length so it typically contains more information than vertical microinstructions. Horizontal microinstructions with 48 or more bits are quite common. Horizontal microprograms have the potential advantage of utilizing hardware more efficiently and on top of it, it requires smaller numbers of microinstructions per microprogram. Vertical microinstructions, on the other hand, are more compact but less flexible than horizontal microinstructions. The vertical approach is consequently easy to use for microprogramming.

# Horizontal VS Vertical Microprogramming

### Comparison Chart

Horizontal Microprogramming	Vertical Microprogramming
Horizontal microprogramming involves horizontal microinstructions that use no encoding at all.	The control bits are encoded in vertical microprogramming.
Horizontal microprogramming generally follows a sequential approach to specify the next microinstruction in a microprogram.	Vertical microprogramming may use a relatively addressing scheme.
Horizontal microprograms generally represent multiple micro-operations that are executed at the same time.	Each vertical microinstruction represents a single micro-operation, while operands may specify the data sink and source.
Horizontal microprograms offer improved flexibility because each control bit is independent of each other.	Vertical microinstructions are more compact but less flexible than horizontal microinstructions are easier to write.

## **Summary of Horizontal and Vertical Microprogramming**

As opposed to horizontal microinstructions, the vertical microinstruction represents single micro-operations. Horizontal microprograms allow higher degree of parallelism with a minimum amount of encoding and separate control fields whereas the control bits are encoded in vertical microprograms. The choice between the two approaches needs to be made carefully. However, in practical, designers use a combination of horizontal and vertical microinstruction formats so that the resulting structure is compact yet efficient.