Control Unit Functional Requirements

By reducing the operation of the processor to its most fundamental level we are able to define exactly what it is that the control unit must cause to happen

Three step process to lead to a characterization of the control unit:

Define the basic elements of processor

Describe the micro-operations processor performs

Determine the functions that the control unit must perform to cause the micro-operations to be performed

The control unit performs two basic tasks:

- Sequencing
- Execution

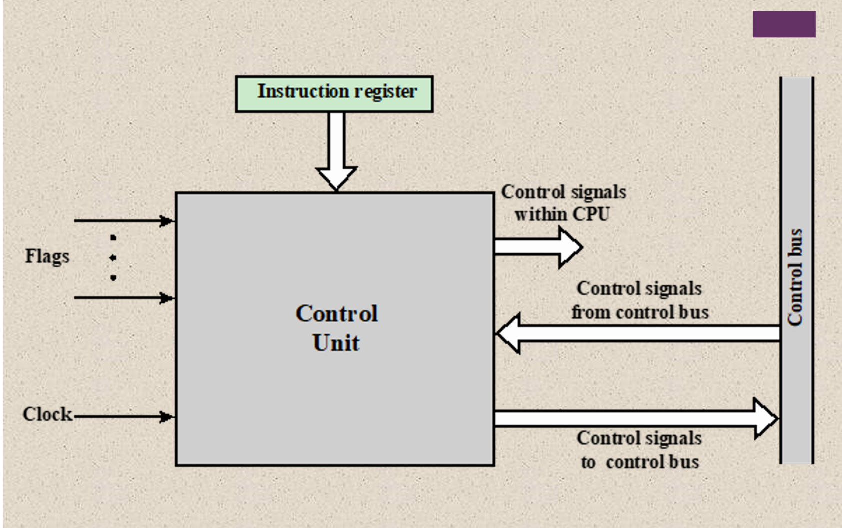

Figure 20.4 Block Diagram of the Control Unit

Control Signals

We have defined the elements that make up the processor (ALU, registers, data paths) and the micro-operations that are performed. For the control unit to perform its function, it must have inputs that allow it to determine the state of the system and outputs that allow it to control the behavior of the system. These are the external specifications of the control unit. Internally, the control unit must have the logic required to perform its sequencing and execution functions.

The inputs are:

Clock: This is how the control unit “keeps time.” The control unit causes one micro-operation (or a set of simultaneous micro-operations) to be performed for each clock pulse. This is sometimes referred to as the processor cycle time, or the clock cycle time.

Instruction register: The opcode and addressing mode of the current instruction are used to determine which micro-operations to perform during the execute cycle.

Flags: These are needed by the control unit to determine the status of the processor and the outcome of previous ALU operations. For example, for the increment-and-skip-if-zero (ISZ) instruction, the control unit will increment the PC if the zero flag is set.

Control signals from control bus: The control bus portion of the system bus provides signals to the control unit.

The outputs are as follows:

Control signals within the processor: These are two types: those that cause data to be moved

from one register to another, and those that activate specific ALU functions.

Control signals to control bus: These are also of two types: control signals to memory, and control signals to the I/O modules.

Three types of control signals are used: those that activate an ALU function; those that activate a data path; and those that are signals on the external system bus or other external interface. All of these signals are ultimately applied directly as binary inputs to individual logic gates. Let us consider again the fetch cycle to see how the control unit maintains control. The control unit keeps track of where it is in the instruction cycle. At a given point, it knows that the fetch cycle is to be performed next. The first step is to transfer the contents of the PC to the MAR. The control unit does this by activating the control signal that opens the gates between the bits of the PC and the bits of the MAR. The next step is to read a word from memory into the MBR and increment the PC. The control unit does this by sending the following control signals simultaneously: A control signal that opens gates, allowing the contents of the MAR onto the address bus; A memory read control signal on the control bus; A control signal opens the gates, allowing the contents of the data bus to be stored in the MBR; Control signals to logic that add 1 to the contents of the PC and store the result back to the PC.

Following this, the control unit sends a control signal that opens gates between the MBR and the IR.This completes the fetch cycle except for one thing: The control unit must decide whether to perform an indirect cycle or an execute cycle next. To decide this, it examines the IR to see if an indirect memory reference is made. The indirect and interrupt cycles work similarly. For the execute cycle, the control unit begins by examining the opcode and, on the basis of that, decides which sequence of micro-operations to perform for the execute cycle.

Data paths: The control unit controls the internal flow of data. For example, on instruction fetch, the contents of the memory buffer register are transferred to the IR. For each path to be controlled, there is a switch (indicated by a circle in the figure). A control signal from the control unit temporarily opens the gate to let data pass.

ALU: The control unit controls the operation of the ALU by a set of control signals. These signals activate various logic circuits and gates within the ALU.

System bus: The control unit sends control signals out onto the control lines of the system bus

(e.g., memory READ).

A Control Signals Example

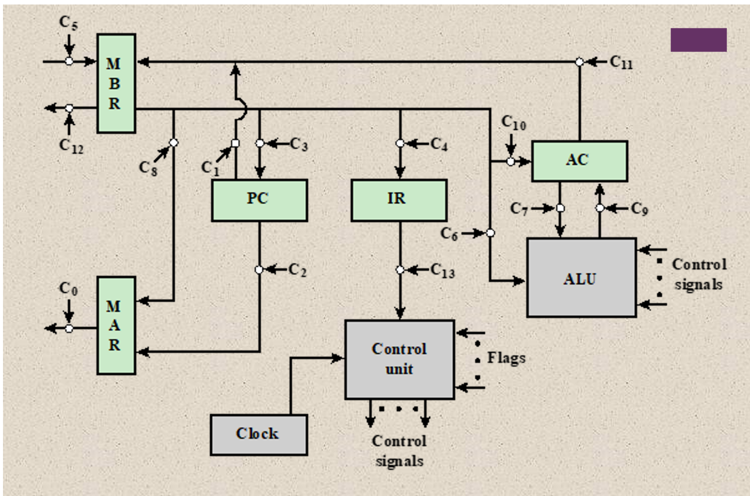

Figure 20.5 Data Paths and Control Signals

Table 20.1 Micro-operations and Control Signals

	Micro-operations	Active Control Signals
Fetch	t_1: MAR ← (PC)	C_2
	t_2: MBR ← Memory PC ← (PC) + 1	C_5, C_R
	t_3: IR ← (MBR)	C_4
Indirect	t_1: MAR ← (IR(Address))	C_8
	t_2: MBR ← Memory	C_5, C_R
	t_3: IR(Address) ← (MBR(Address))	C_4
Interrupt	t_1: MBR ← (PC)	C_1
	t_2: MAR ← Save-address PC ← Routine-address	
	t_3: Memory ← (MBR)	C_{12}, C_W

C_R = Read control signal to system bus.
C_W = Write control signal to system bus.

(Table can be found on page 719 in the textbook.)

HORIZONTAL AND VERTICAL MICRO-PROGRAMMED CONTROL UNIT

The **control unit (CU)** is the engine that runs the entire functions of a computer with the help of control signals in the proper sequence. In the **micro-programmed** control unit approach, the control signals that are associated with the operations are stored in special memory units. It is convenient to think of sets of control signals that cause specific micro-operations to occur as being "micro-instructions". The sequences of micro-instructions could be stored in an internal "*control*" memory.

The micro-programmed control unit can be classified into two types based on the type of Control Word stored in the Control Memory, viz., Horizontal micro-programmed control unit and Vertical micro-programmed control unit.

- In the *Horizontal micro-programmed* control unit, the control signals are represented in the decoded binary format, i.e., 1 bit/CS. Here 'n' control signals require n bit encoding. On the other hand.
- In a *Vertical micro-programmed* control unit, the control signals are represented in the encoded binary format. Here 'n' control signals require $\log_2 n$ bit encoding.

Difference between Horizontal and Vertical micro-programmed Control Unit:

S. No	Horizontal μ-programmed CU	Vertical μ-programmed CU

1.	It supports longer control word.	It supports shorter control word.
2.	It allows a higher degree of parallelism. If degree is n, then n Control Signals are enabled at a time.	It allows a low degree of parallelism i.e., the degree of parallelism is either 0 or 1.
3.	No additional hardware is required.	Additional hardware in the form of decoders is required to generate control signals.
4.	It is faster than a Vertical micro-programmed control unit.	it is slower than a Horizontal micro-programmed control unit.
5.	It is less flexible than a Vertical micro-programmed control unit.	It is more flexible than a Horizontal micro-programmed control unit.
6.	A horizontal micro-programmed control unit uses horizontal micro-instruction, where every bit in the control field attaches to a control line.	A vertical micro-programmed control unit uses vertical micro-instruction, where a code is used for each action to be performed and the decoder translates this code into individual control signals.
7.	The horizontal micro-programmed control unit makes less use of ROM encoding than the vertical micro-programmed control unit.	The vertical micro-programmed control unit makes more use of ROM encoding to reduce the length of the control word.

What is a Hardwired Control Unit?

A hardwired control is an approach to generating control signals utilising Finite State Machines (FSM) appropriately.

What is a Microprogrammed Control Unit?

A Microprogrammed control unit generates the control signals to retrieve and execute instructions. This control unit generates sequencing via microinstructions.

Difference between Hardwired and Microprogrammed Control Unit

S.No.	Hardwired Control Unit	Microprogrammed Control Unit
1.	The hardwired control unit induces the control signals required for the processor.	The microprogrammed control unit induces the control signals through microinstructions.
2.	Hardwired control unit is quicker than a microprogrammed control unit.	Microprogrammed control unit is slower than a hardwired control unit.
3.	It is hard to modify.	It is easy to modify.
4.	It is more expensive as compared to the microprogrammed control unit.	It is affordable as compared to the hardwired control unit.
5.	It faces difficulty in managing the complex instructions because the design of the circuit is also complex.	It can easily manage complex instructions.
6.	It can use limited instructions.	It can generate control signals for many instructions.