VIT-AP UNIVERSITY

ECE-2002 Computer Organization and Architecture

Dr. Kritika Bansal

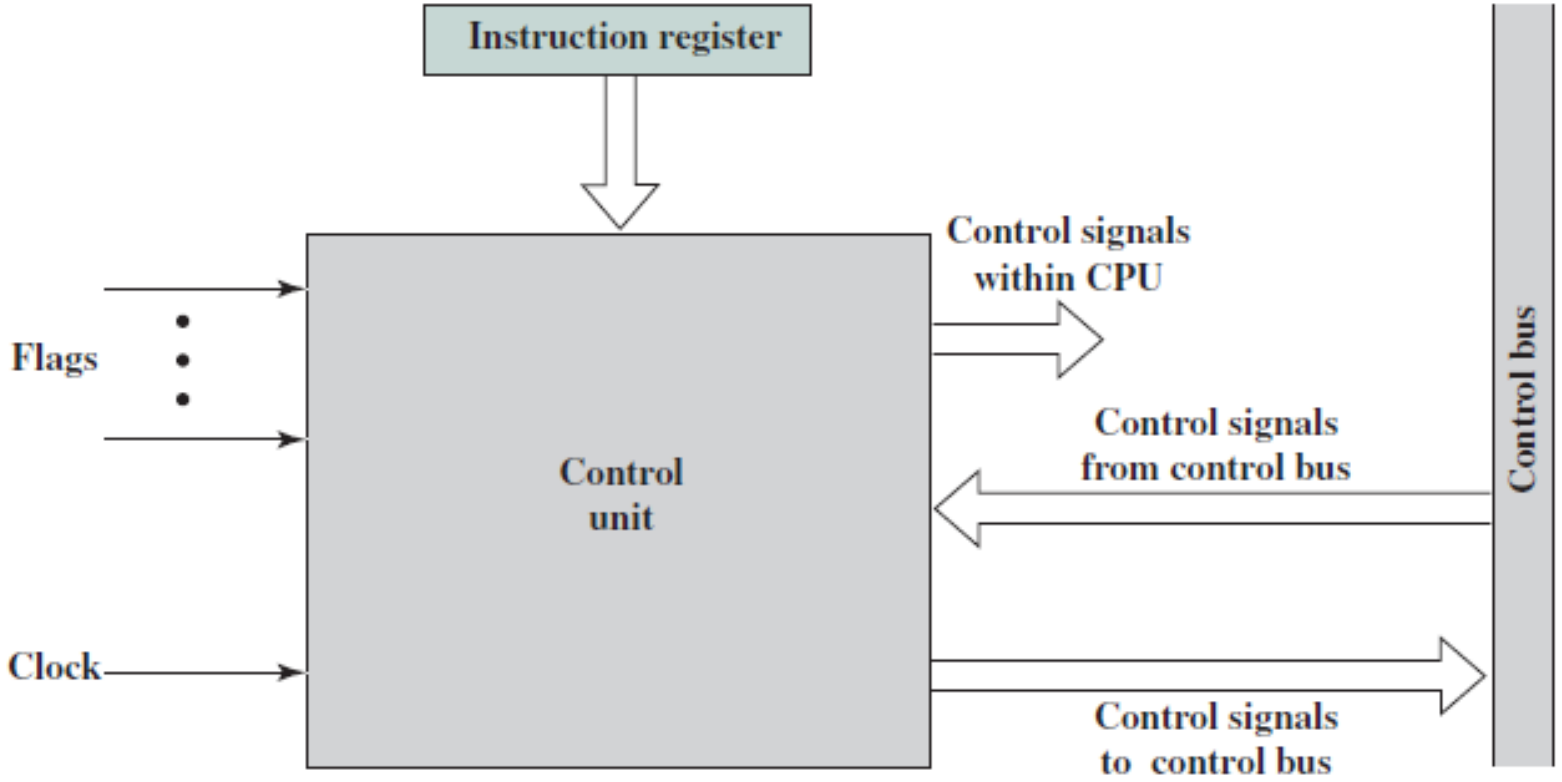
School of Electronics Engineering, VIT-AP

Control Unit

- The control unit performs two basic tasks:
  - **Sequencing**: The control unit causes the processor to step through a series of micro-operations in the proper sequence, based on the program being executed.
  - **Execution**: The control unit causes each micro-operation to be performed.
- Implementation of Control unit is broadly of two types
  - **Hardwired implementation**
  - **Microprogrammed implementation**

Hardwired Control Unit

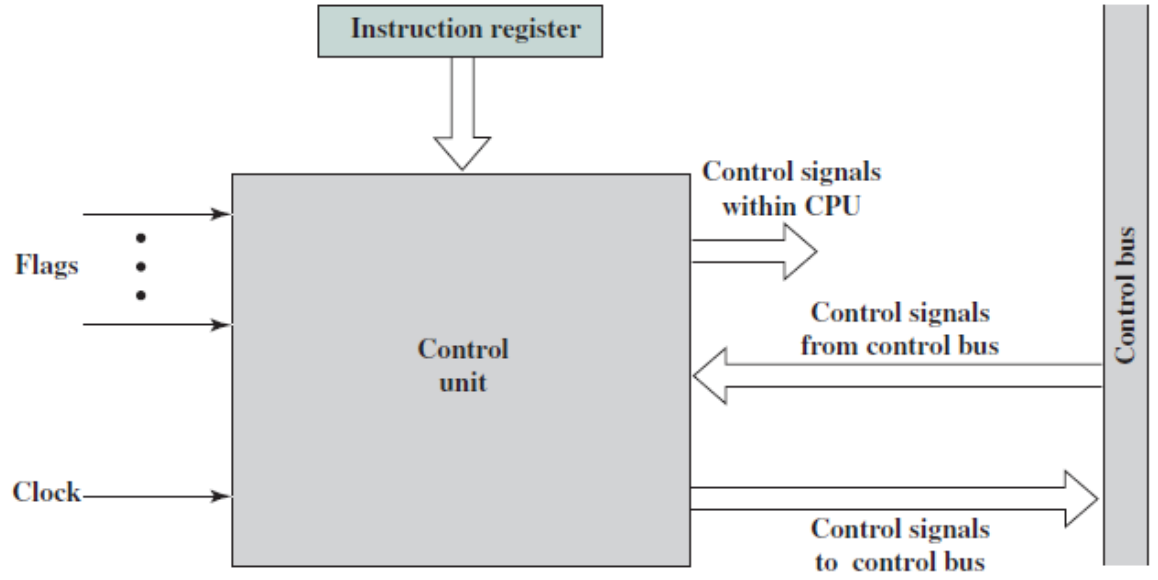
Control Unit



➢The inputs are:

Clock: The control unit causes one micro-operation (or a set of simultaneous micro-operations) to be performed for each clock pulse.

Control Unit

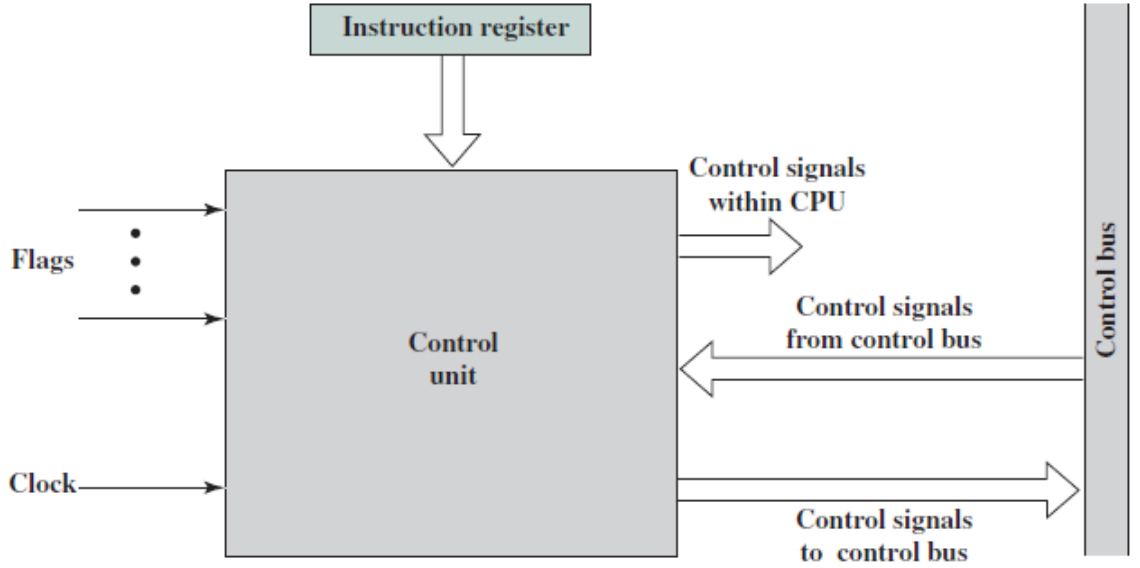


Instruction register: The opcode and addressing mode of the current instruction are used to determine which micro-operations to perform during the execute cycle.

Flags: These are needed by the control unit to determine the status of the processor and the outcome of previous ALU operations.

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

Control Unit



- The outputs are :

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.

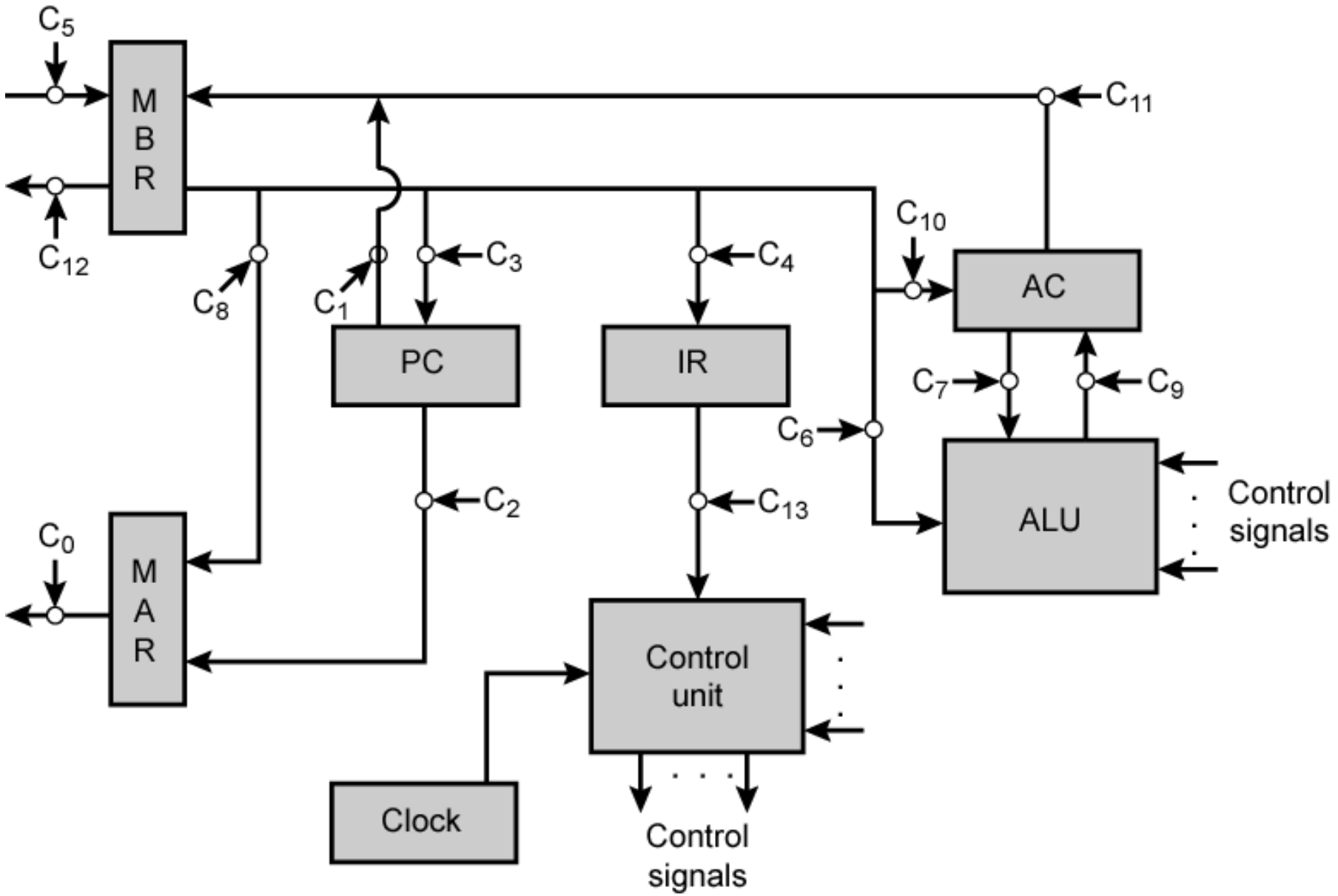
Control signals to control bus: These are also of two types: control signals to memory, and control signals to the I/O modules.

Example- Fetch Cycle

How control signal is used for fetch cycle

- **MAR <- PC**
  - Control unit activates signal to open gates between PC and MAR
- **MBR <- memory**
- **PC <- PC+1**
  - Control unit activates signal to open gates between MAR and address bus
  - Control unit activates Memory read control signal
  - Control unit activates signal to open gates between data bus and MBR
  - Control unit activates signal to logic that add 1 to the contents of PC
- **IR <- MBR**
  - Control unit activates signal to open gates between MBR and IR

Data Paths and Control Signals

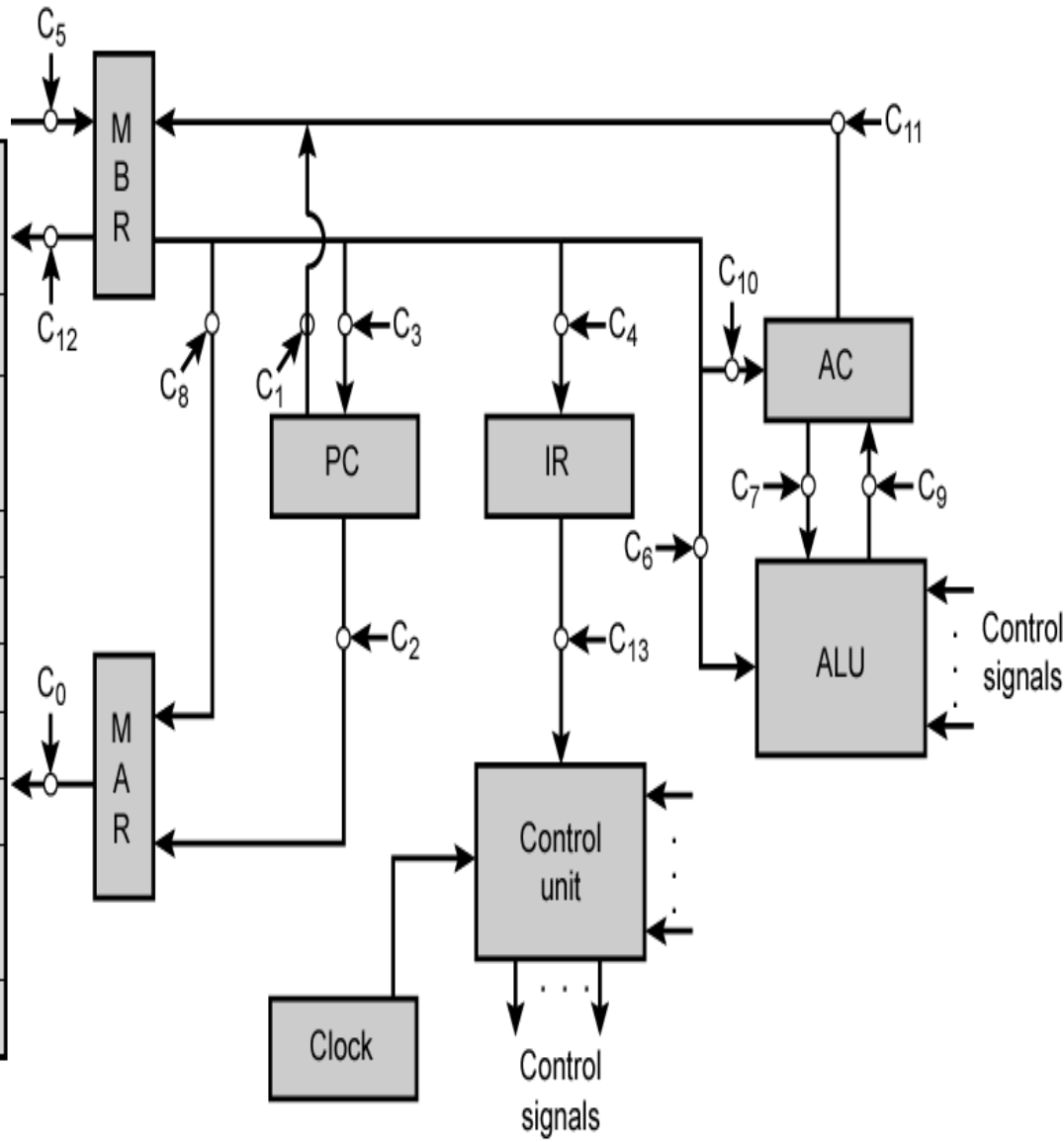


Data Paths, Micro-operations and Control Signals

	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

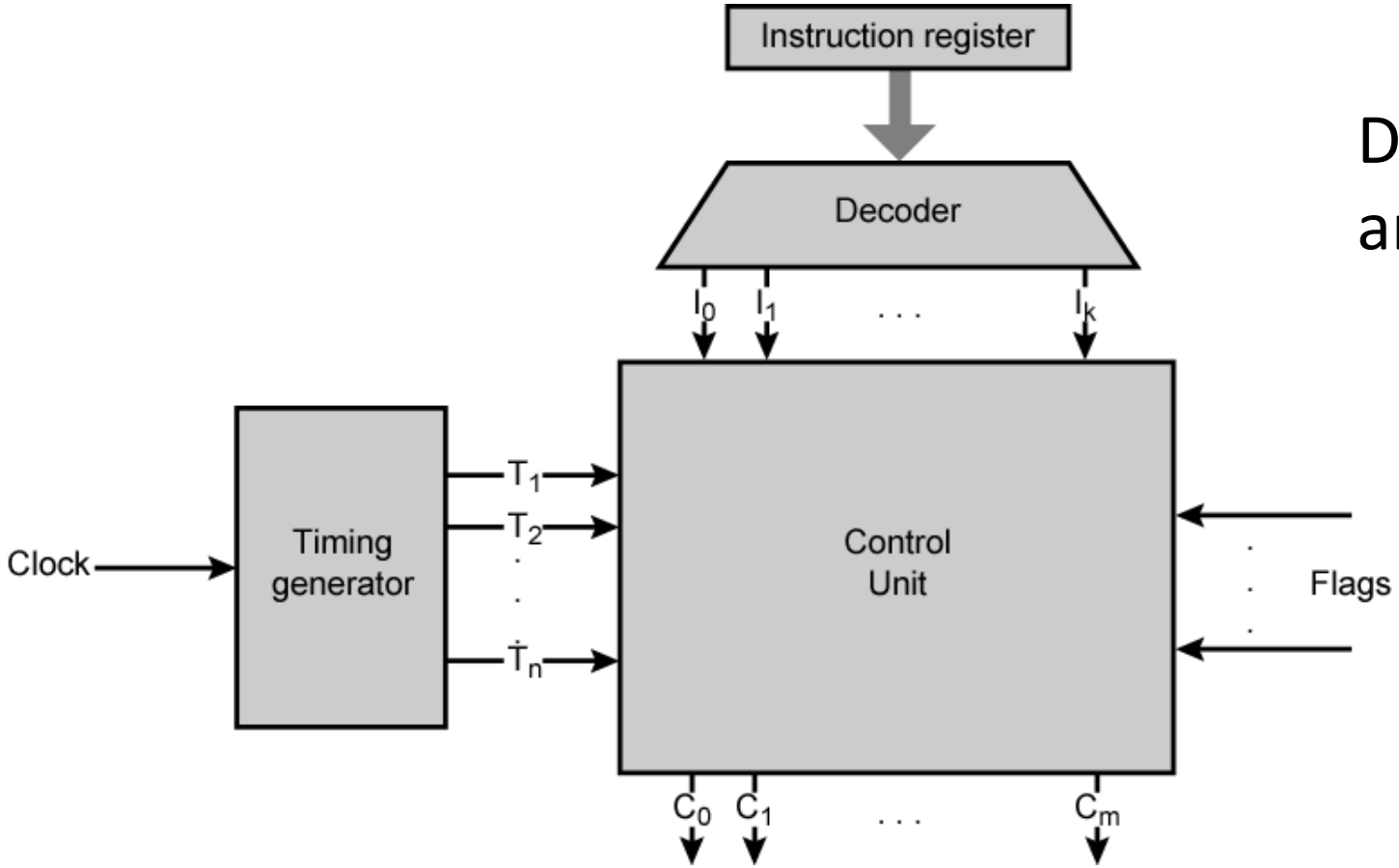
C_R = Read control signal to system bus.

C_W = Write control signal to system bus.



Control Unit with Decoded Inputs

- Different control signals for different instructions
- To simplify, unique logic input for each opcode



Decoder: n binary inputs and 2^n binary outputs

Example

- For each control signal, to derive a Boolean expression of that signal as a function of the inputs
- Let us consider a **single control signal, C5**, which causes data to be read from the external data bus into the MBR
- Let us define two new **control signals,**

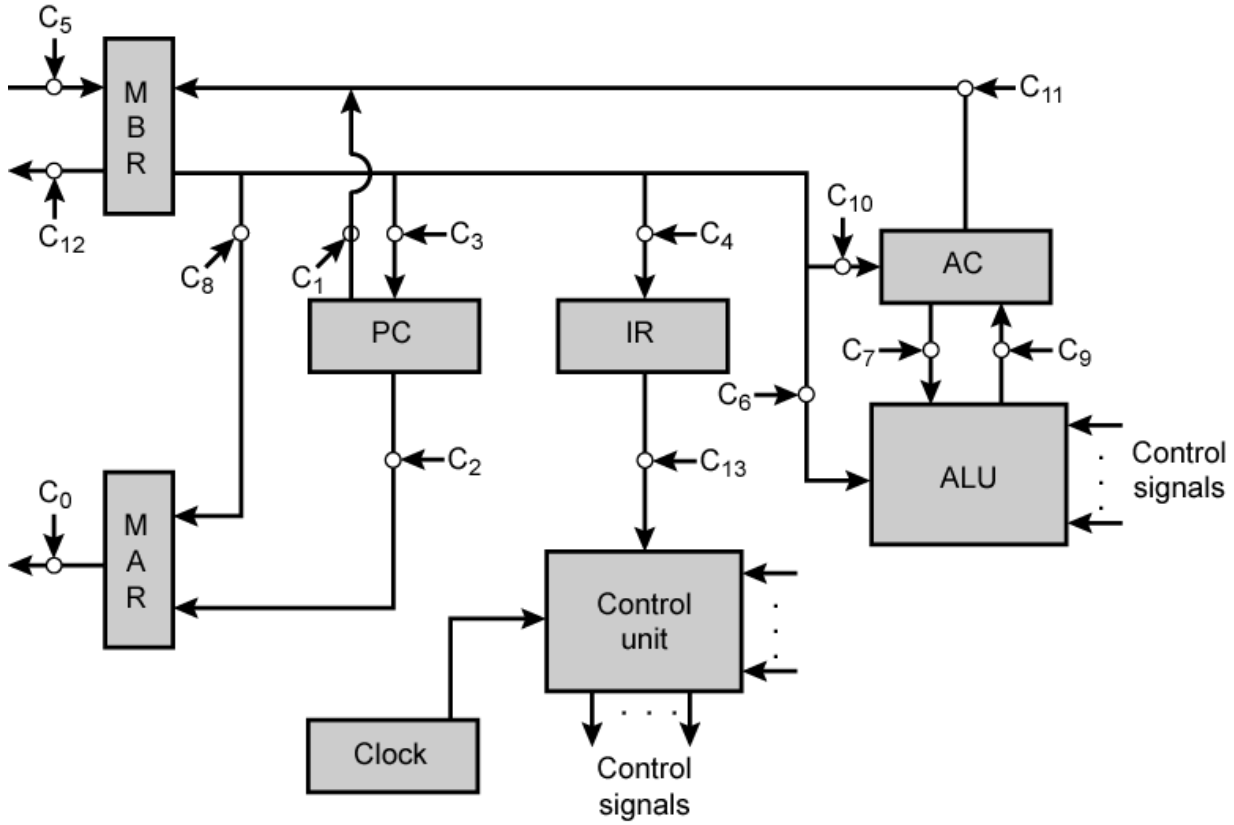
P and Q, that have the following interpretation:

PQ = 00 Fetch Cycle

PQ = 01 Indirect Cycle

PQ = 10 Execute Cycle

PQ = 11 Interrupt Cycle



Example

	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.

- Then C5 can be defined using the Boolean expression as:

$$C_5 = \overline{P} \cdot \overline{Q} \cdot T_2 + \overline{P} \cdot Q \cdot T_2$$

- That is, the control signal C5 will be asserted during the second time unit of both the fetch and indirect cycles.

PQ = 00 Fetch Cycle
PQ = 01 Indirect Cycle
PQ = 10 Execute Cycle
PQ = 11 Interrupt Cycle

Example

	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.

- C5 is also needed during the execute cycle.
- Assume that there are only three instructions that read from memory: LDA, ADD and AND.
- Then C5 can be defined using the Boolean expression as:

$$C_5 = \overline{P} \cdot \overline{Q} \cdot T_2 + \overline{P} \cdot Q \cdot T_2 + P \cdot \overline{Q} \cdot (LDA + ADD + AND) \cdot T_2$$

PQ = 00 Fetch Cycle
PQ = 01 Indirect Cycle
PQ = 10 Execute Cycle
PQ = 11 Interrupt Cycle