Computer Organization

- It is sometimes convenient to use the address bits of an instruction code not as an address but as the actual operand.
- When the second part of an instruction code specifies an operand, the instruction is said to have an immediate operand.
- When the second part specifies the address of an operand, the instruction is said to have a direct address.
- This is in contrast to a third possibility called indirect address, where the bits in the second part of the instruction designate an address of a memory word in which the address of the operand is found.

Computer Organization

- One bit of the instruction code can be used to distinguish between a direct and an indirect address.
- As an illustration of this configuration, consider the instruction code format shown in Fig. 5-2(a). It consists of a 3-bit operation code, a 12-bit address, and an address mode bit designated by I.
- The mode bit is 0 for a direct address and 1 for an indirect address.
- A direct address instruction is shown in Fig. 5-2(b).

Computer Organization

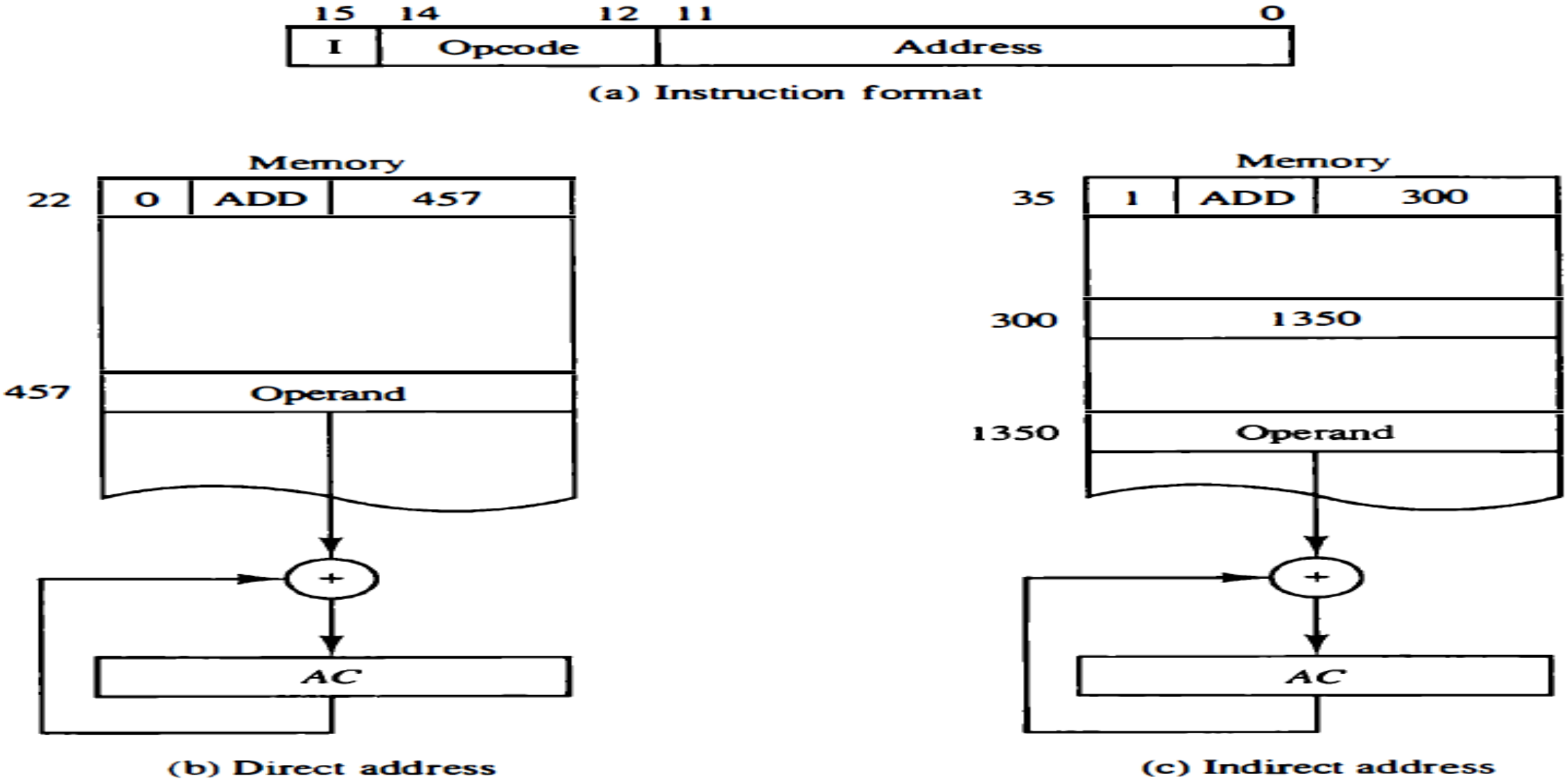


Figure 5.2 Demonstration of direct and indirect address

Computer Organization

- The I bit is 0, so the instruction is recognized as a direct address instruction. The opcode specifies an ADD instruction, and the address part is the binary equivalent of 457.

- The control finds the operand in memory at address 457 and adds it to the content of AC .

- Fig. 5-2(c) has a mode bit I = 1. Therefore, it is recognized as an indirect address instruction.

Computer Organization

- Fig. 5-2(c) has a mode bit I = 1. Therefore, it is recognized as an indirect address instruction. The address part is the binary equivalent of 300.
- The control goes to address 300 to find the address of the operand. The address of the operand in this case is 1350.
- The operand found in address 1350 is then added to the content of AC .
- The indirect address instruction needs two references to memory to fetch an operand.
- The first reference is needed to read the address of the operand; the second is for the operand itself.
- Thus the effective address in the instruction of Fig. 5-2(b) is 457 and in the instruction of Fig 5-2(c) is 1350.

Computer Registers

- Computer instructions are normally stored in consecutive memory locations and are executed sequentially one at a time.
- The control reads an instruction from a specific address in memory and executes it.
- It then continues by reading the next instruction in sequence and executes it, and so on.
- This type of instruction sequencing needs a counter to calculate the address of the next instruction after execution of the current instruction is completed.

Computer Registers

- It is also necessary to provide a register in the control unit for storing the instruction code after it is read from memory.
- The computer needs processor registers for manipulating data and a register for holding a memory address.
- These requirements dictate the register configuration shown in Fig. 5-3.
- The registers are also listed in Table 5-1 together with a brief description of their function and the number of bits that they contain.

Computer Registers

- The memory unit has a capacity of 4096 words and each word contains 16 bits.
- Twelve bits of an instruction word are needed to specify the address of an operand.
- This leaves three bits for the operation part of the instruction and a bit to specify a direct or indirect address.

Computer Registers

- The data register (DR) holds the operand read from memory.
- The accumulator (AC) register is a general purpose processing register.
- The instruction read from memory is placed in the instruction register (IR).
- The temporary register (TR) is used for holding temporary data during the processing.

Computer Registers

TABLE 5-1 List of Registers for the Basic Computer

Register symbol	Number of bits	Register name	Function
DR	16	Data register	Holds memory operand
AR	12	Address register	Holds address for memory
AC	16	Accumulator	Processor register
IR	16	Instruction register	Holds instruction code
PC	12	Program counter	Holds address of instruction
TR	16	Temporary register	Holds temporary data
INPR	8	Input register	Holds input character
OUTR	8	Output register	Holds output character

Computer Registers

- The memory address register (AR) has 12 bits since this is the width of a memory address.
- The program counter (PC) also has 12 bits and it holds the address of the next instruction to be read from memory after the current instruction is executed.
- Two registers are used for input and output.
- The input register (INPR) receives an 8-bit character from an input device.
- The output register (OUTR) holds an 8-bit character for an output device.

Computer Registers

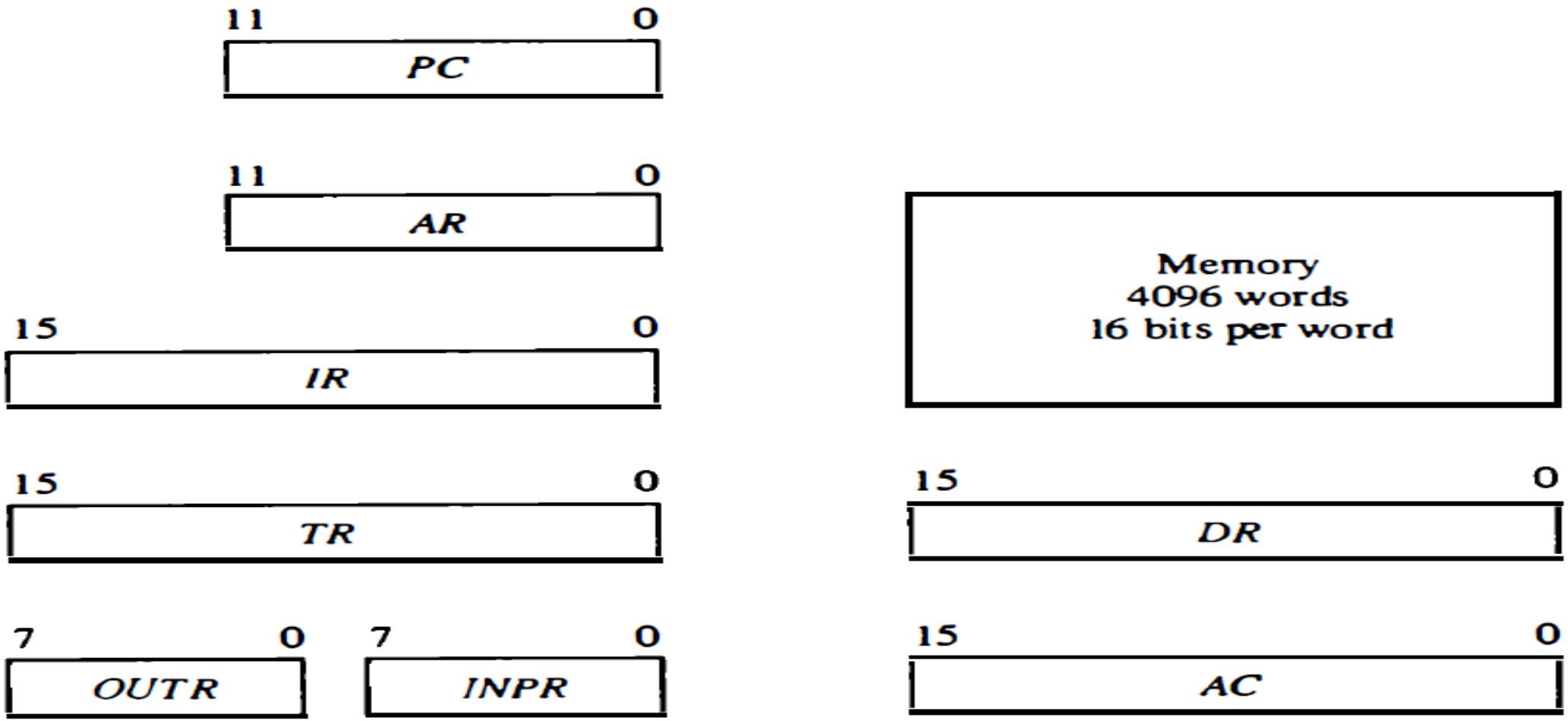


Figure 5-3 Basic computer registers and memory.

Tutorial

- What do you mean by sequential circuits ?

Difference between the combinational circuits and sequential circuits are given below

	Combinational Circuits	Sequential Circuits
1)	The outputs of the combinational circuit depend only on the present inputs.	The outputs of the sequential circuits depend on both present inputs and present state(previous output).
2)	The feedback path is not present in the combinational circuit.	The feedback path is present in the sequential circuits.
3)	In combinational circuits, memory elements are not required.	In the sequential circuit, memory elements play an important role and require.
4)	The clock signal is not required for combinational circuits.	The clock signal is required for sequential circuits.

Tutorial

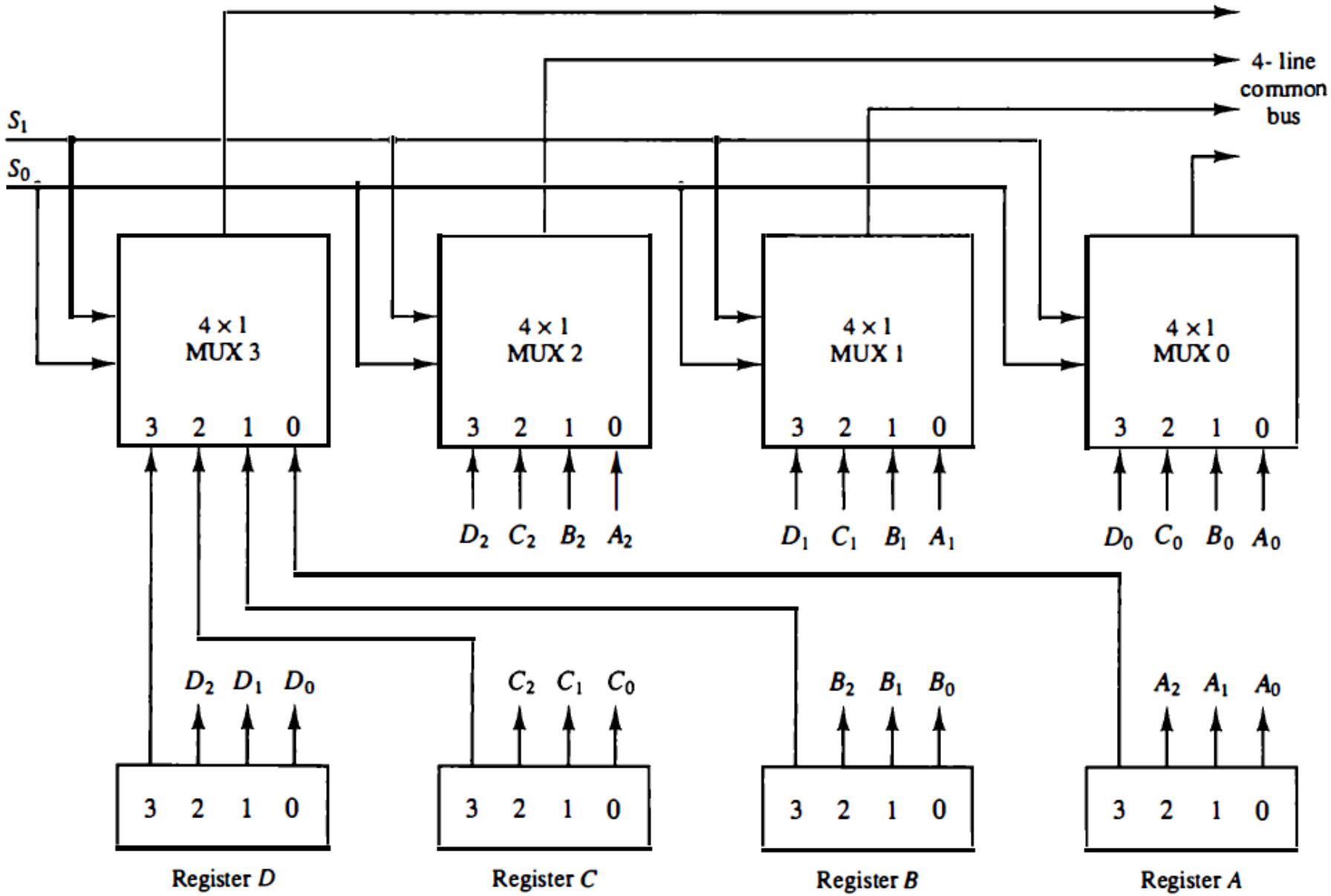
- What is bus system with 4 registers and 4 mux ?

Bus and Memory transfer

- A typical digital computer has many registers, and paths must be provided to transfer information from one register to another.
- The number of wires will be excessive if separate lines are used between each register and all other registers in the system.
- A more efficient scheme for transferring information between registers in a multiple-register configuration is a common bus system.

- BUS STRUCTURE CONSISTS OF SET OF COMMON LINES, ONE FOR EACH BIT OF A REGISTER THROUGH WHICH BINARY INFORMATION IS TRANSFERRED ONE AT A TIME
- Have control circuits to select which register is the source, and which is the destination

Figure 4-3 Bus system for four registers.



S1	S0	Register selected
0	0	A
0	1	B
1	0	C
1	1	D

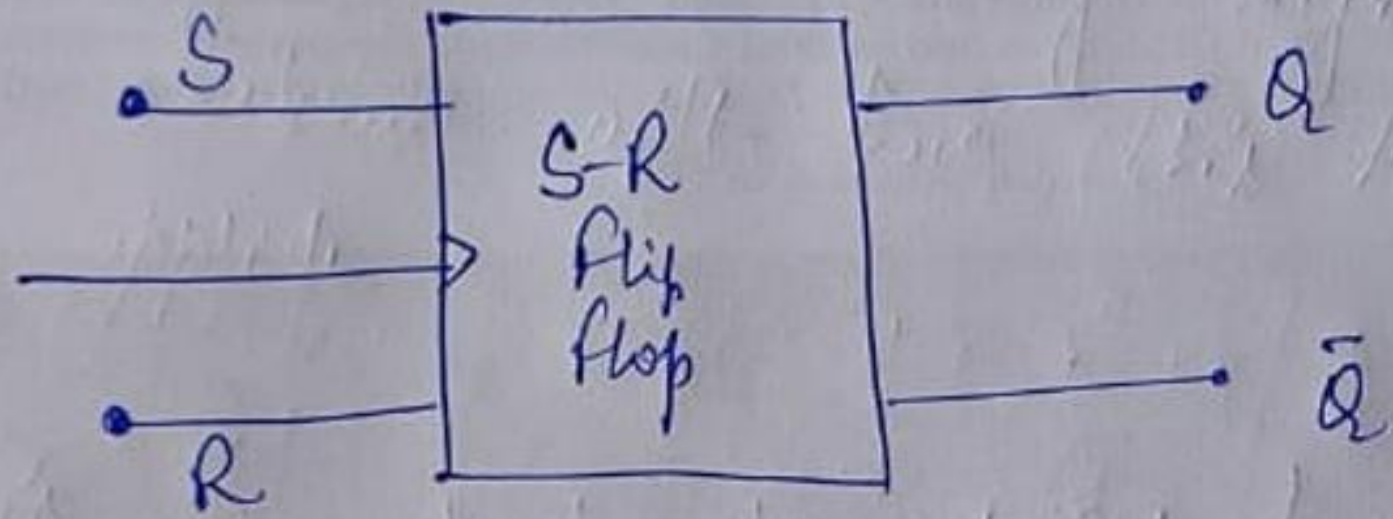
Tutorial

- What is SR Flip flop ? Explain with truth table ?

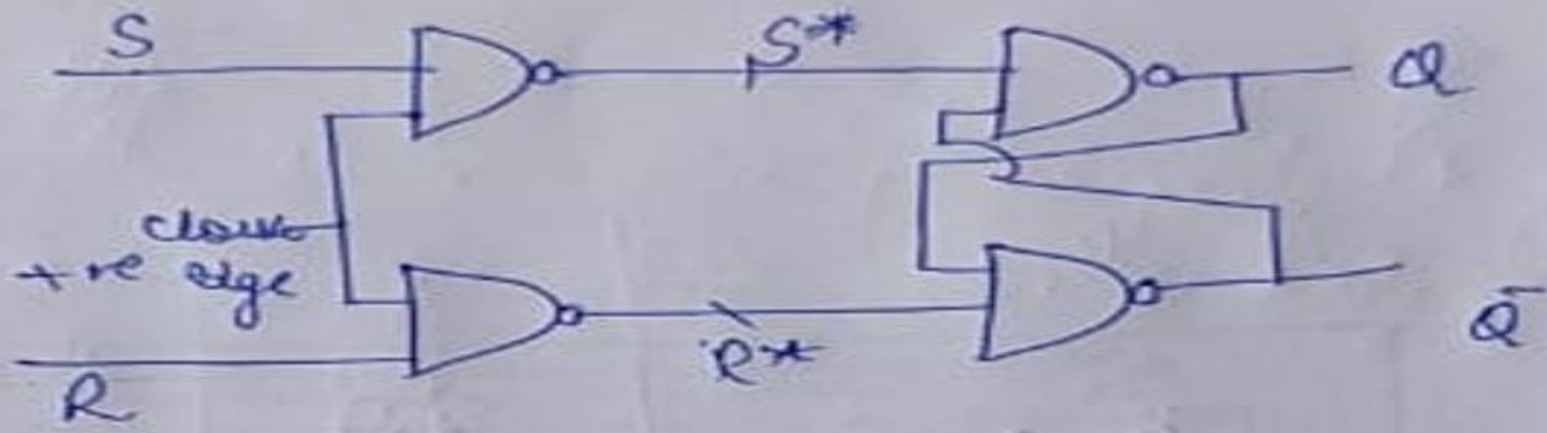
SR FLIP FLOP

- SR Flip Flop is the set reset flip flop. It consist of SR latch with clock circuit.
- It may be positive edge triggered or negative edge triggered.
- Triggering is the process of change of state of flop by applying input signal.

(1) S-R flip flop



positive edge triggered



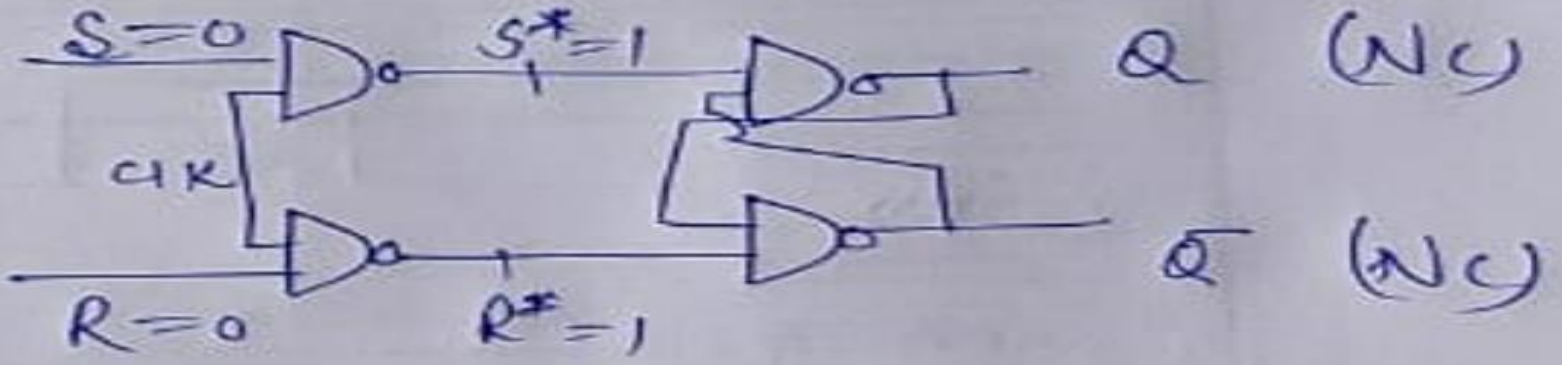
(SR NAND Latch)

S*	R*	Q $\bar{Q}$
0	0	Invalid
0	1	1 0
1	0	0 1
1	1	NC

No change

Case I :-

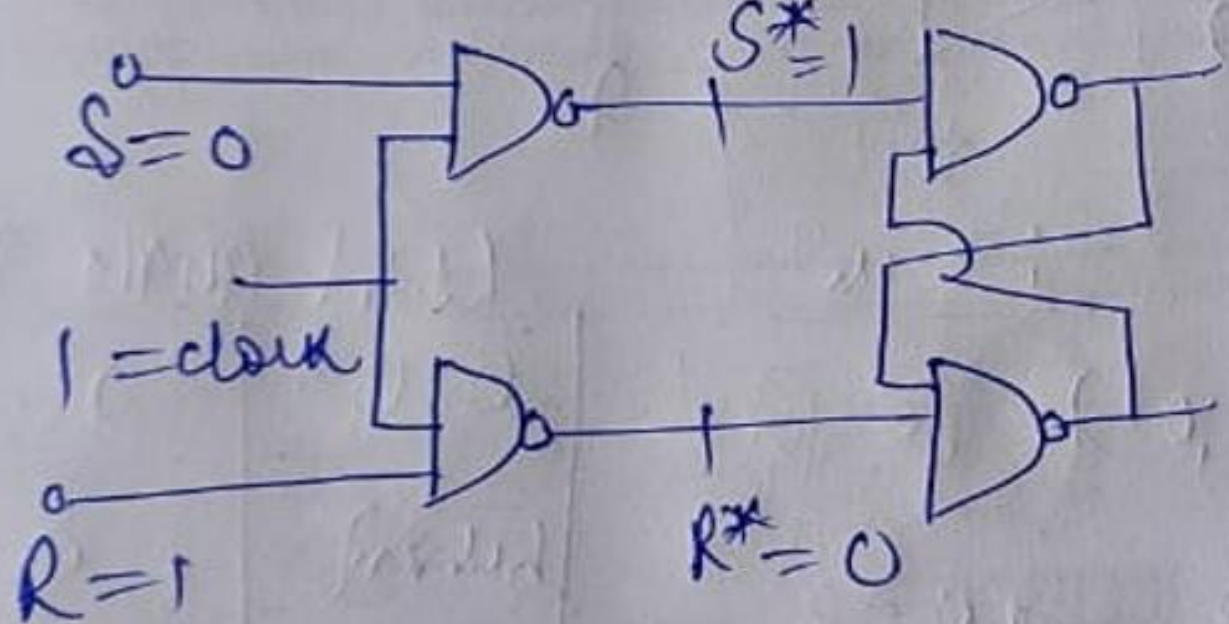
S=0, R=0, clock = ↑



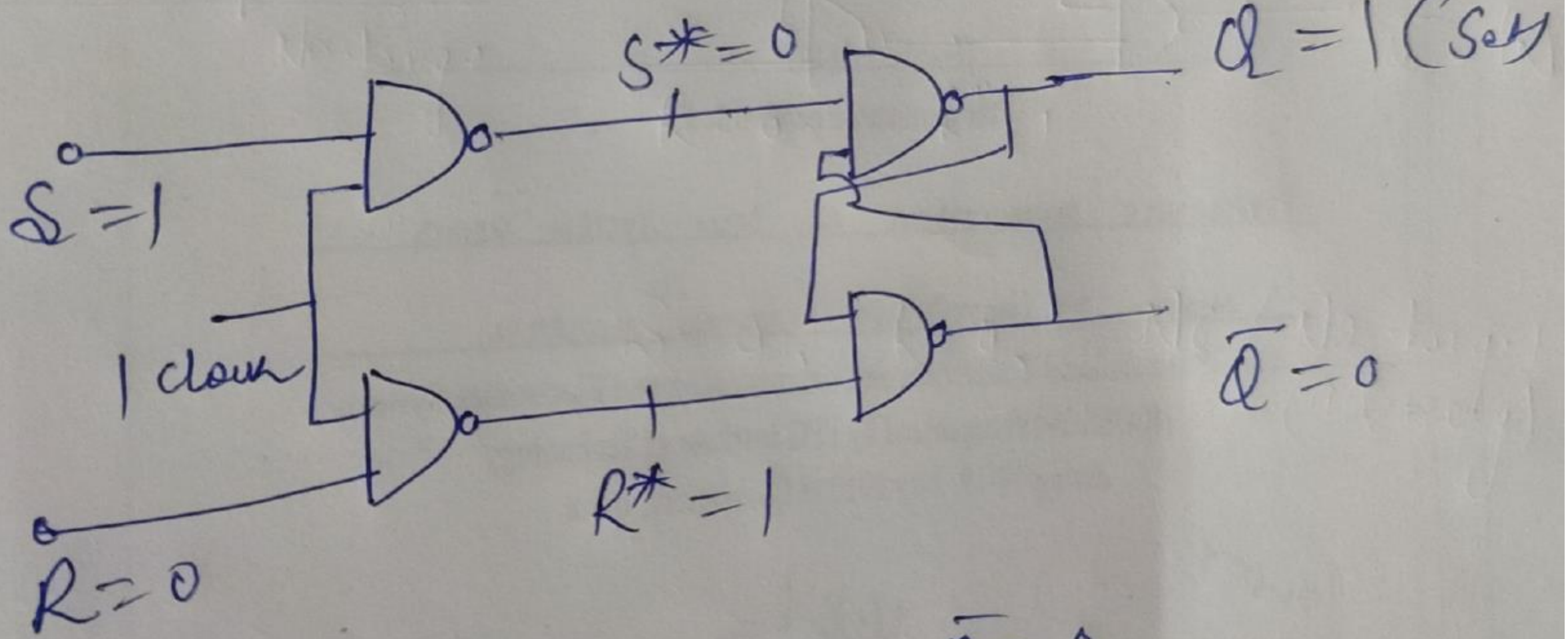
S=0, R=0, NC (No change)
(Memory)

Case II :-

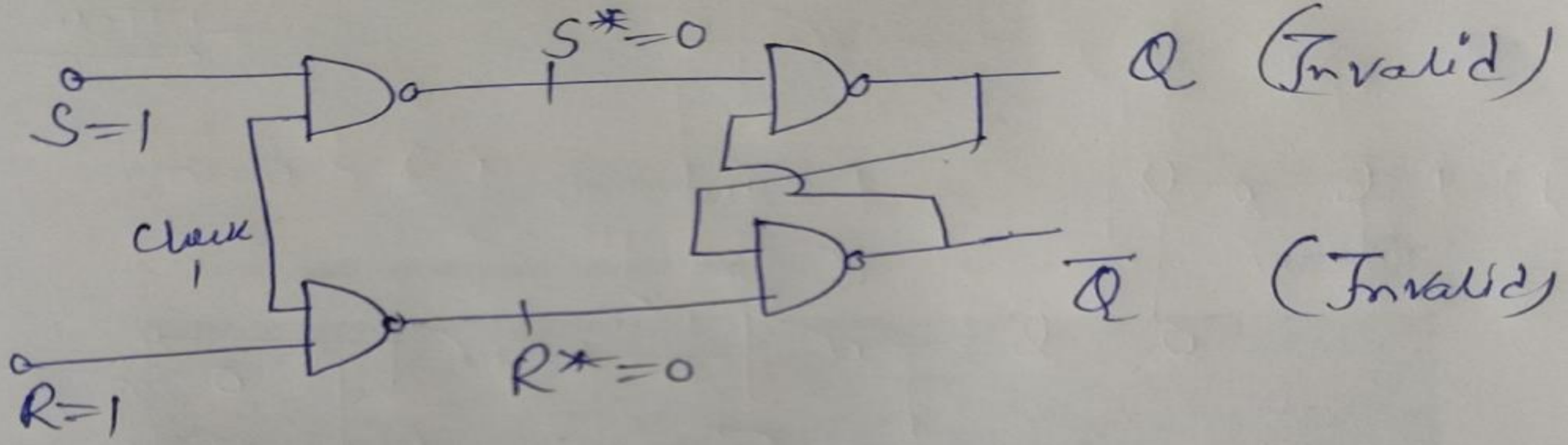
S=0 & R=1 , clock = 1 Q=0, $\bar{Q}$=1 (Reset)



Case III :-



S = 1; R = 0 Q = 1; $\bar{Q}$ = 0



Clock	S	R	Q	$\overline{Q}$	
↑	0	0	NC	NC	
↑	0	1	0	1	Reset
↑	1	0	1	0	Set
↑	1	1	Invalid		