Unit 3: Numericals

Qn1.

A computer uses a memory unit with 256K words of 32 bits each. A binary instruction code is stored in one word of memory. The instruction has four parts: an indirect bit, an operation code, a register code part to specify one of 64 registers, and an address part.

a. How many bits are there in the operation code, the register code part, and the address part?
b. Draw the instruction word format and indicate the number of bits in each part.
c. How many bits are there in the data and address inputs of the memory?

Solution:

$256\ K = 2^8 \times 2^{10} = 2^{18}$

$64 = 2^6$

(a) Address: 18 bits
Register code: 6 bits
Indirect bit: 1 bit
25 $32 - 25 = 7$ bits for opcode.

(b) 1 7 6 18 = 32 bits

I	opcode	Register	Address

(c) Data; 32 bits; address: 18 bits.

Qn2.

What is the difference between a direct and an indirect address instruction? How many references to memory are needed for each type of instruction to bring an operand into a processor register?

Solution:

A direct address instruction needs two references to memory: (1) Read instruction; (2) Read operand.

An indirect address instruction needs three references to memory:
(1) Read instruction; (2) Read effective address; (3) Read operand.

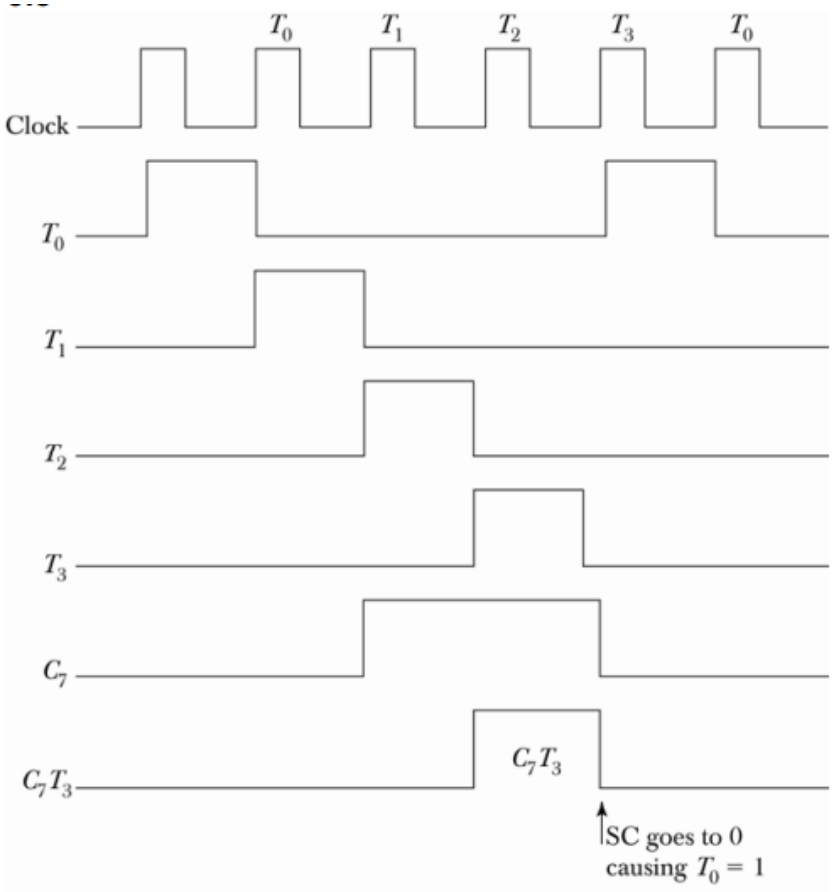
Qn3.

Draw a timing diagram assuming that SC is cleared to 0 at time T_3 if control signal C_7 is active.

$$C_7T_3: \quad SC \leftarrow 0$$

C_7 is activated with the positive clock transition associated with T_1.

Solution:

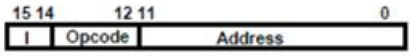


Qn4.

An instruction at address 021 in the basic computer has $I = 0$, an operation code of the AND instruction, and an address part equal to 083 (all numbers are in hexadecimal). The memory word at address 083 contains the operand B8F2 and the content of AC is A937. Go over the instruction cycle and determine the contents of the following registers at the end of the execute phase: PC, AR, DR, AC, and IR. Repeat the problem six more times starting with an operation code of another memory-reference instruction.

Solution:

Memory-Reference Instructions (OP-code = 000 ~ 110)



Symbol	Hex Code I = 0	Hex Code I = 1	Description
AND	0xxx	8xxx	AND memory word to AC
ADD	1xxx	9xxx	Add memory word to AC
LDA	2xxx	Axxx	Load AC from memory
STA	3xxx	Bxxx	Store content of AC into memory
BUN	4xxx	Cxxx	Branch unconditionally
BSA	5xxx	Dxxx	Branch and save return address
ISZ	6xxx	Exxx	Increment and skip if zero

No need

	PC	AR	DR	AC	IR
Initial	021	—	—	A937	—
AND	022	083	B8F2	A832	0083
ADD	022	083	B8F2	6229	1083
LDA	022	083	B8F2	B8F2	2083
STA	022	083	—	A937	3083
BUN	083	083	—	A937	4083
BSA	084	084	—	A937	5083
ISZ	022	083	B8F3	A937	6083

Qn5.

Show the complete logic of the interrupt flip-flops R in the basic computer. Use a JK flip-flop and minimize the number of gates.

$$(T_0 + T_1 + T_2)'\,(IEN)\,(FGI + FGO) : \quad R \leftarrow 1$$
$$RT_2 : \quad R \leftarrow 0$$