

ICS Problem #10

Sheet #10

CS'22

Problem 10.1. a)

#	Machine Code	Assembly Code	Description
0	001 1 0001	Load #1	Load value 1 in accumulator
1	010 0 1111	STORE 15	Store the value of accumulator in location 15
2	001 1 0000	Load #0	Load value 0 in accumulator
3	101 1 0100	Equal #4	Skip next instruction if accumulator value equals to 4
4	110 1 0110	Jump #6	Jump to instruction 6
5	111 1 0000	HALT	Stop execution
6	001 0 0011	Load 3	Load the value of memory location 3 into accumulator
7	100 1 0001	Sub #1	Subtract the value 1 from accumulator
8	010 0 0011	STORE 3	Store the value of accumulator in location 3
9	001 0 1111	Load 15	Load the value of memory location 15 into accumulator
10	011 0 1111	ADD 15	Add the value of the memory location 15 to accumulator
11	010 0 1111	STORE 15	Store the value of accumulator in memory location 15.
12	110 1 0010	Jump #2.	Jump to instruction 2.
13	000 0 0000		
14	000 0 0000		
15	000 0 0000		

- b) Initially the program starts as it stores the value 1 in accumulator and then to location 15.

Then it ~~does the~~ ~~same~~ loads 0 and compares it with number 4. If true it will skip next instruction else if false in this case it just goes to location 6.

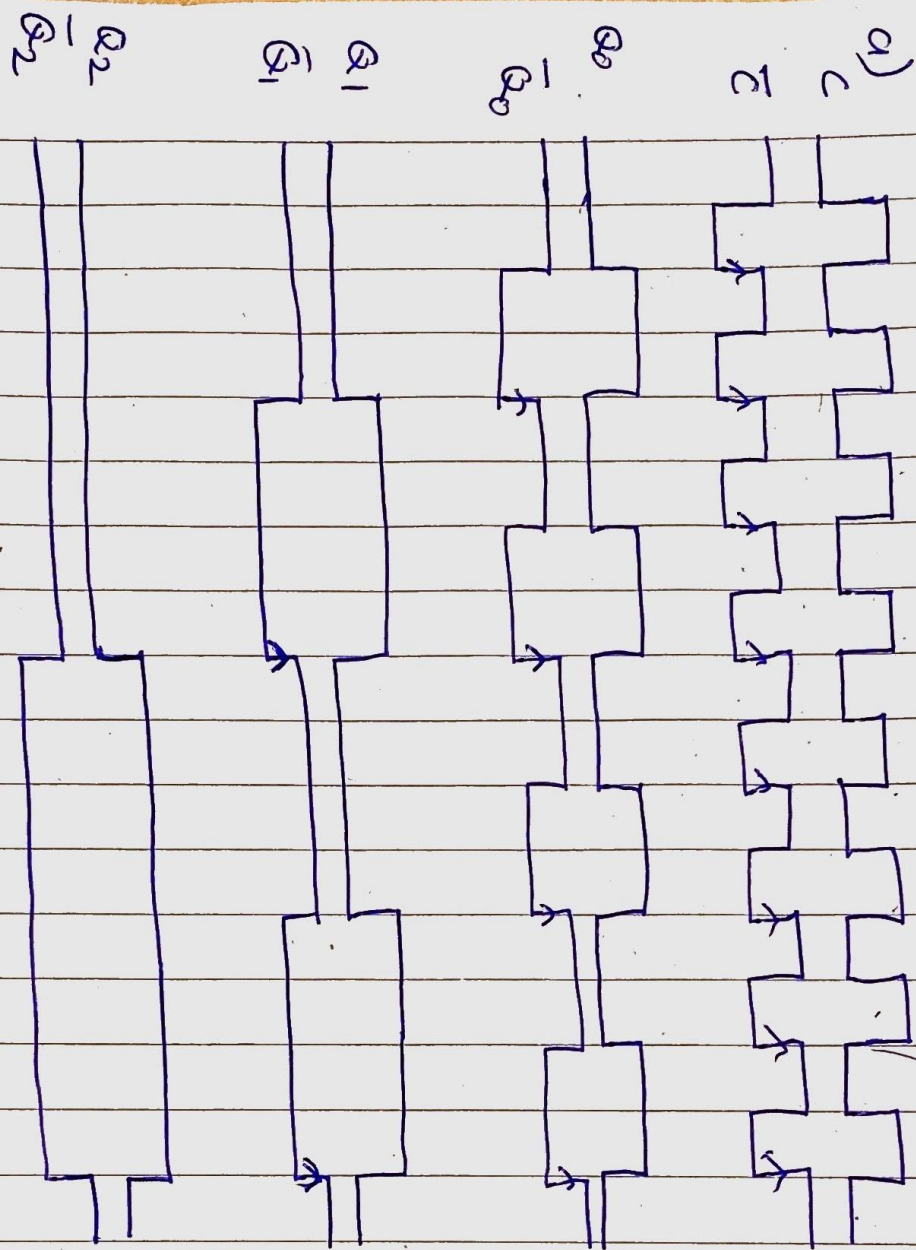
Then value in 3 which is ~~Equal to~~ 4 is loaded into accumulator and decreased by one. Then ~~its~~ value from memory location 15 which is 1 is loaded into accumulator and is increased by 1 (itself).

Then again it goes to instruction 2 and 0 is loaded to the accumulator and compared with 3 this time however. The process is repeated and we can figure out this makes the ~~loop~~ process run 4 times and value in memory location 15 changes from $1 \rightarrow 2 \rightarrow 4 \rightarrow 8 \rightarrow 16$ and finally when ① and Equal to 0 match it goes to address 5 and the program terminates.

- program
c) When ~~it~~ ends, the value 16 is left in accumulator.

- c) In general terms we can say it ~~is~~ as ~~increasing~~ 2^n where n runs from 0 to 4.

Problem 10.2



- b) You can not make ripple counters arbitrary long due to the presence of gate delays. ~~a therefore~~ The gate delays will shift as the index of Q increases, defeating the purpose of circuit.