BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

I Semester 2017-2018

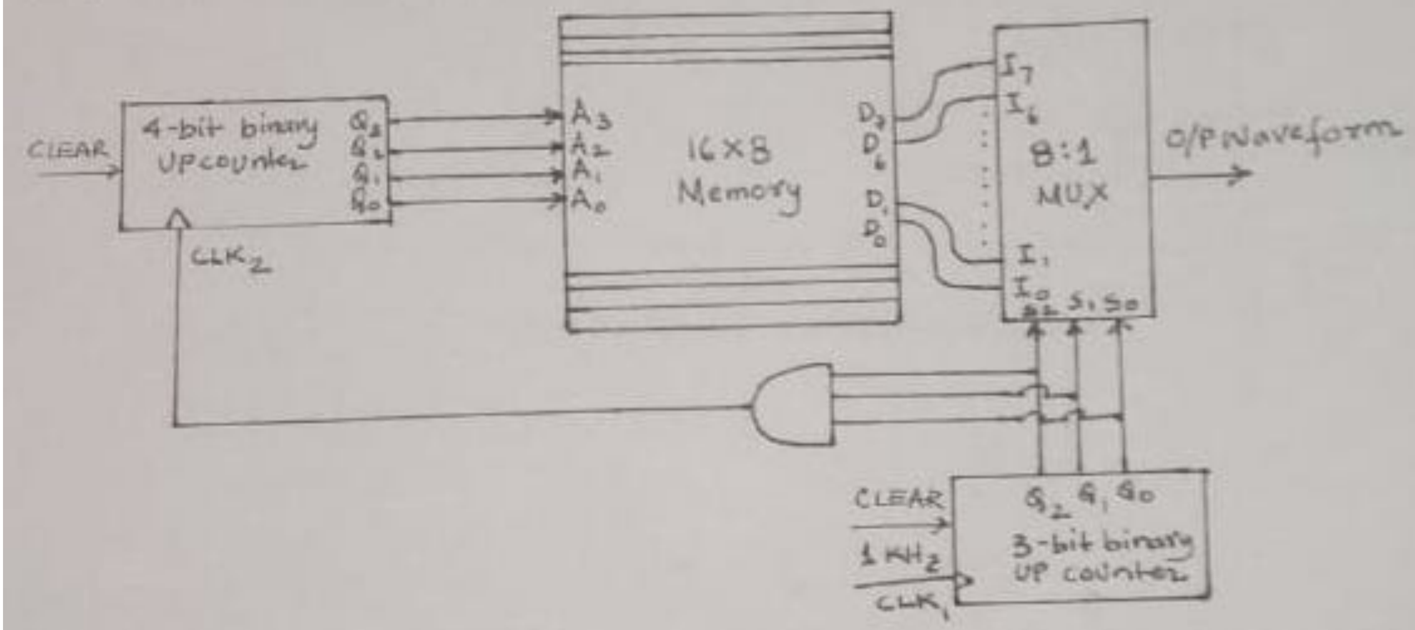
CS F342 Computer Architecture

Lab Exam

Date: 19/11/2017 TIME: 90 Minutes MM: 33

INSTRUCTIONS: (i) Create a folder on desktop, rename it to your IDNO. (ii) Rename the project with your IDNO (iii) Write your IDNO and name in .v file. *Write all modules in a single .v file.* Put this .v file in the folder you created on desktop.

Q. Implement a waveform generator as shown below:



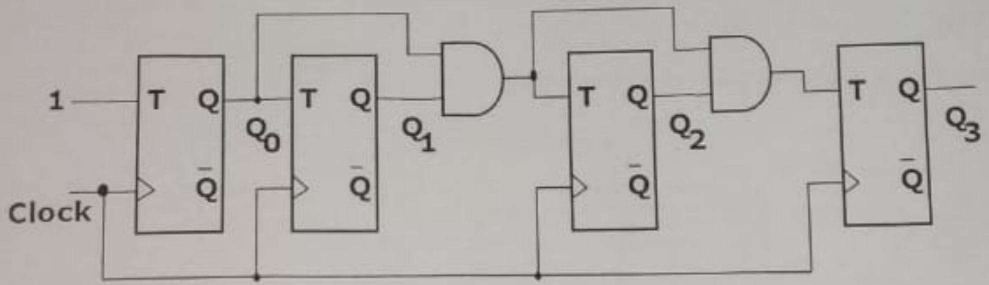
The Waveform generator consists of 2 binary counters, one memory of size 16x8, one MUX of size 8:1 and one AND gate. Different waveforms can be generated by initializing the 16 memory locations with different 8- bit values. Use the following modules to implement the above circuit (use all signal names as shown in circuit above):

1. **MUX_SMALL:** This module implements the functionality of a 2:1 MUX. Implement this module using dataflow modeling.
2. **MUX_BIG:** This module implements the functionality of an 8:1 MUX. Build this module using appropriate numbers of MUX_SMALL modules.
3. **TFF:** This module implements the functionality of a T-Flip flop with an asynchronous clear. Implement this module using behavioral modeling.
4. **COUNTER_4BIT:** This module implements the functionality of a 4-bit synchronous binary UP counter with an asynchronous clear. Implement this module as shown in figure below. Use TFF modules for your implementation. Include a CLEAR input in the circuit.
5. **COUNTER_3BIT:** This module is similar to the COUNTER_4BIT module except that it implements the functionality of a 3-bit synchronous binary UP counter with an asynchronous clear. Use TFF modules for your implementation. Include a CLEAR input in the circuit.
6. **MEMORY:** This module implements the functionality of a 16x8 memory. Initialize the 16 memory registers with values 0xCC, 0xAA, 0xCC,0xAA, 0xCC,0xAA, 0xCC,0xAA, 0xCC,0xAA, 0xCC,0xAA, 0xCC,0xAA, 0xCC,0xAA, respectively. i.e. Register0 is initialized with value 0xCC, Register1 is initialized with value 0xAA, Register2 is initialized with value 0xCC and so on.
7. **INTG:** This module integrates all above modules into a working circuit.

PTO→

Also write a test bench module which:

a. Clears the two counters
b. Applies a clock of 1 KHz at input of the 3- bit counter.

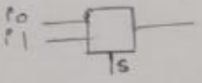


4-bit synchronous up counter

s2 s1 s1 ; i[0]
; i[1]
; i[1]
; i1

16:1 4:1

s2 s3



16

S ; i1
1
0 ; i0

s0 s1 s2

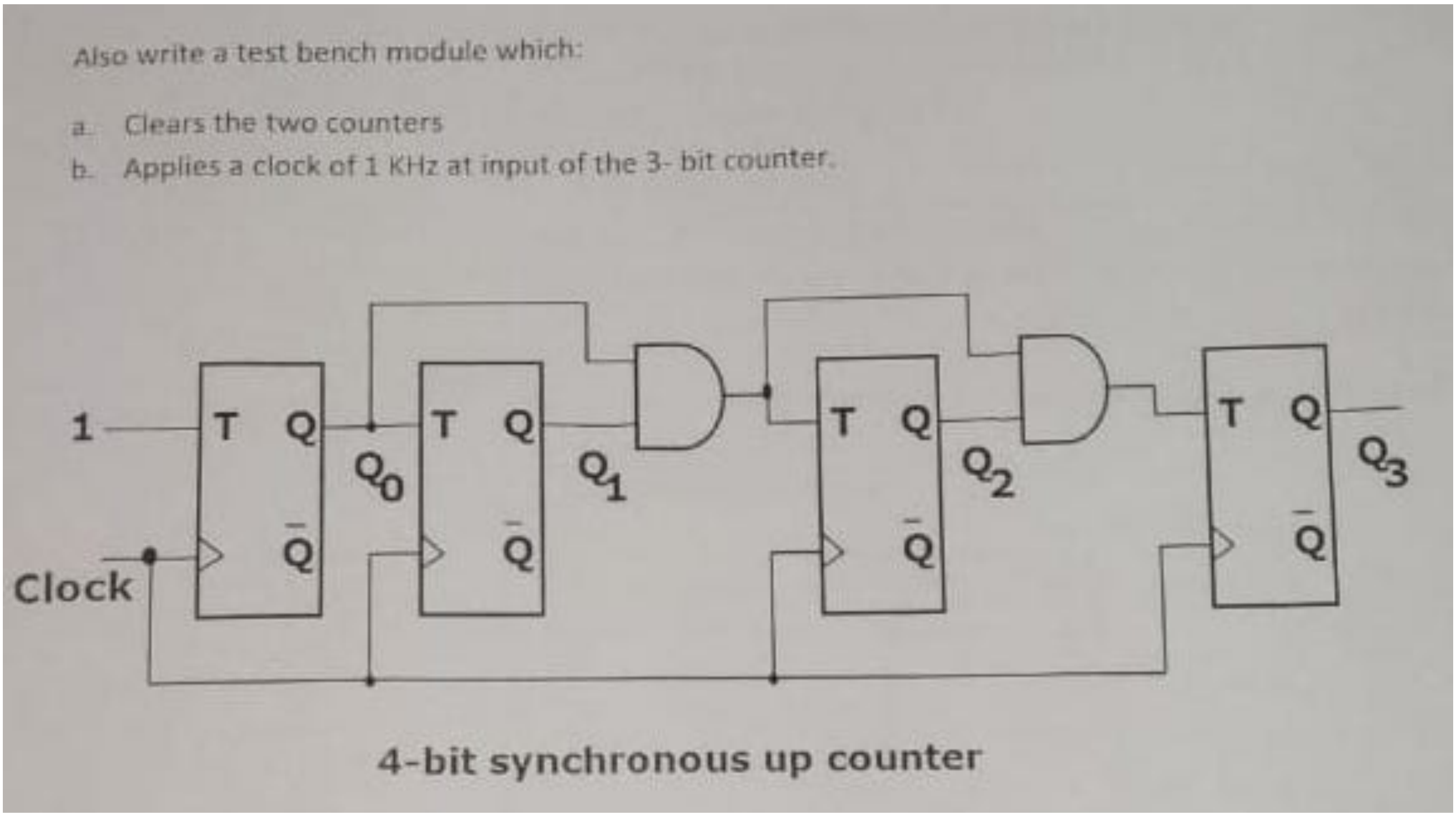
4/2 → 1

111 ; i[0]

a[0] → s1 s3

s0 s1

a s0 s1



Also write a test bench module which:

a. Clears the two counters

b. Applies a clock of 1 KHz at input of the 3- bit counter.

4-bit synchronous up counter