Problem set 4.

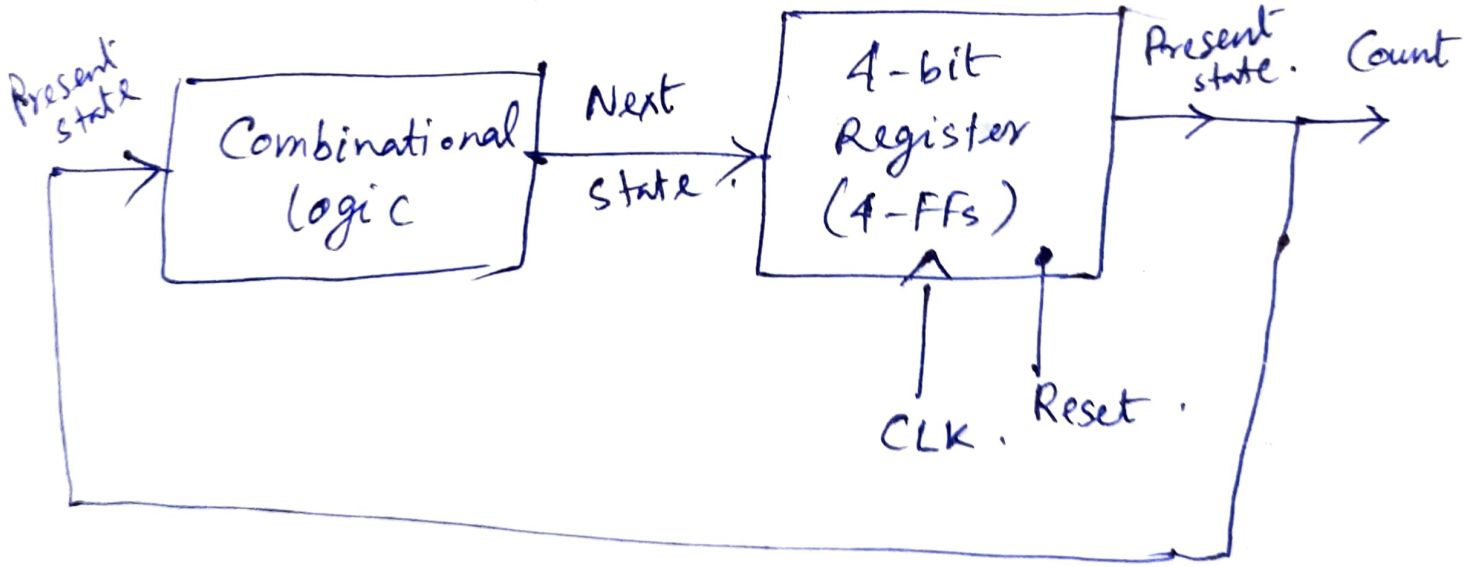
(1) Design the following arbitary sequence counter using VHDL. Verify it's behaviour by writing a test-bench.

$0 \rightarrow 1 \rightarrow 4 \rightarrow 2 \rightarrow 6 \rightarrow 5 \rightarrow 9 \rightarrow 12 \rightarrow 0 \rightarrow$ ... Repeat

[Don't forget to include an asynchronous Reset i/p.]

Hints:-

(i) Consider the concept of Moore machine to design this counter.



(ii) Combinational logic is defined by the State-transition graph.

Next_state = f (Present state)

Count <= present state

(iii) Write two concurrent processes within the architecture. to.

- One for the registers => Process(clk, rst)
- Another one for the Combinational logic.

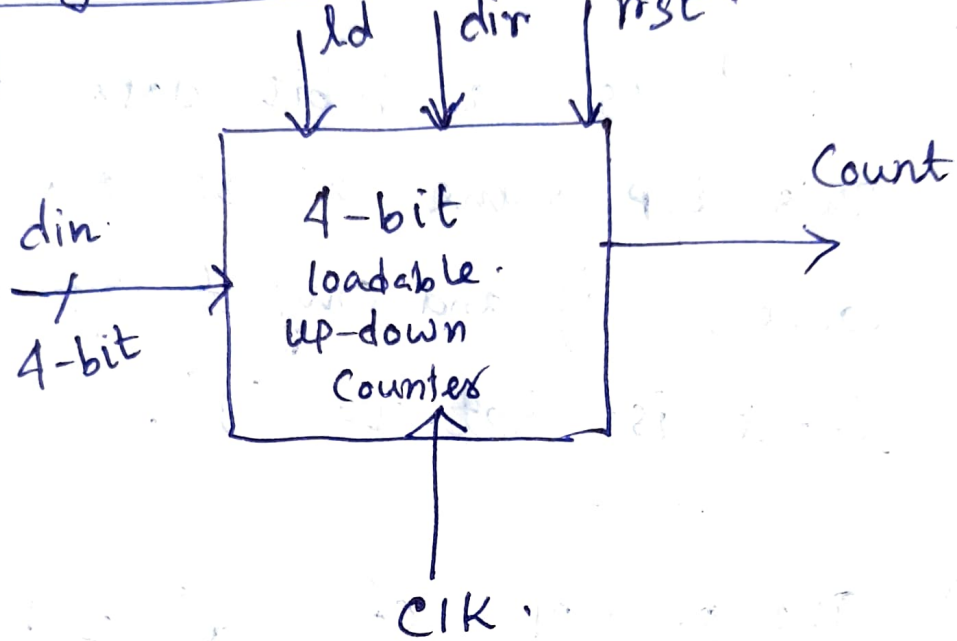
↓↓

process(present_state).

(2) Design a 4-bit up-docon loadable counter with synchronous i/Ps : 'ld', 'dir', & 'din'.

'ld' → 1-bit, 'dir' → 1-bit, 'din' → 4-bit.

and one asynchronous i/p : 'rst' ; with following Specifications:



(i). When 'ld' will be set to 1, the data i/p (din) will be loaded into the Counter and the count value will be updated from the loaded value based on the 'dir' signal. 'ld' is a synchronous i/p.

(ii) { 'dir' = '1' → enable 'up-count'
{ 'dir' = '0' → " down-count.

→ Synchronous i/p.

(iii) 'rst' is a normal asynchronous reset i/p of the Counter.

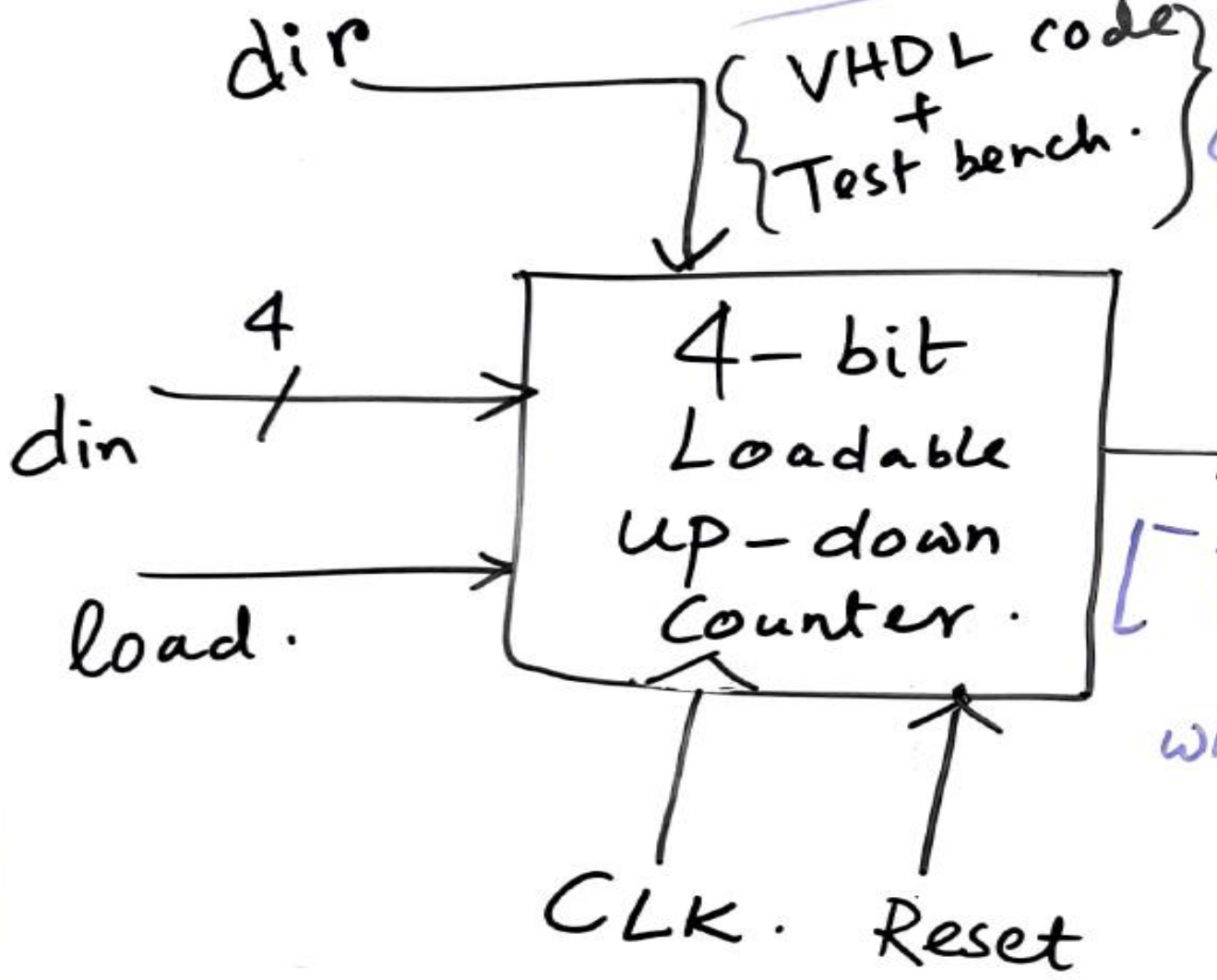
Reset

(iv) 'din' is a 4-bit data input line. This i/p value will be loaded into the Counter if and only if synchronous to 'ld' input is set to '1'.

(v) The priority of 'ld' is higher than that of 'din'.

Verify your design by writing an appropriate test-bench code.

Priority: load > dir



dir = '1' ⇒ up-count.
dir = '0' ⇒ down-count.
↳ Synchronous.

[Reset ⇒ asynchronous.
load ⇒ synchronous.
When load = '1'
din is loaded into
Counter.]