Lab 6: Finite State Machine

Objective

- Design a Finite State Machine to control a counter.
- Implement a Finite State Machine in Verilog

Introduction

A finite-state machine (FSM) is a mathematical model of computation used to design sequential logic circuits. It is conceived as an abstract machine that can be in one of a finite number of states. The machine is in only one state at a time; the state it is in at any given time is called the current state. It can change from one state to another when initiated by a triggering event or condition; this is called a transition. The Algorithmic State Machine (ASM) is a method for designing finite state machines. It is used to represent diagrams of digital integrated circuits. The ASM diagram is like a state diagram but less formal and thus easier to understand. An ASM chart is a method of describing the sequential operations of a digital system.

Lab Description

In this lab you should be design a finite state machine (FSM) which controls a counter. The FSM should wait for a start signal. After start is received the counter should count to 15 and then return to the INIT state. The count should return to 0 upon entering the INIT state. The done output should be 1 when the counter is idle and 0 when the counter is busy.
The following tables show the input and output of counter and FSM

Table1: Counter signals

Signal	Direction/Width	Description
clk	input/1-bit	clock for all sequential logic
reset	input/1-bit	reset signal for all sequential logic
cnten	input/1-bit	count enable; counter should in next clock cycle when this signal is 1.
cntclr	input/1-bit	count clear; force count to 0 in next clock cycle
count	output/8-bit	stores the current count value

Table 2: Finite state machine signals

Signal	Direction/Width	Description
clk	input/1-bit	clock for all sequential logic
reset	input/1-bit	reset signal for all sequential logic
start	input/1-bit	If start=1 leave INIT state and start counting process
done	output/1-bit	done=1 if counting process is complete. Done should stay 1 until the next start is received.
cnten	output/1-bit	count enable; counter should in next clock cycle when this signal is 1.
cntclr	output/1-bit	count clear; force count to 0 in next clock cycle

Homework's

Demo the following modules to the professor:

1. Draw an ASM chart for the finite state machine.
2. Write an Verilog module to FSM (Named *fsm.v*)
3. Write a module for counter(Named *counter.v*)
4. Create a top module named *top.v* (see figure 1)
5. Simulate the circuit (create testbench module named *test.v*)

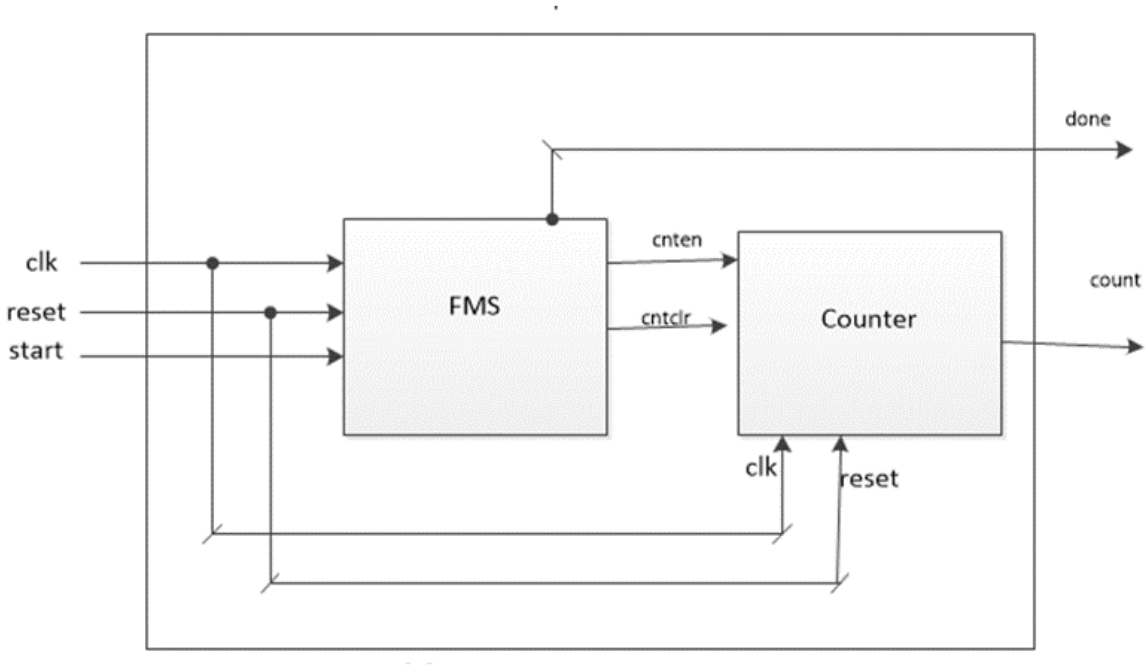

Figure 1: top module

Report and grading:

1. Demonstrate successful simulation and implementation in lab
2. Submit all files (*.v and *.bit) for each part. You should be show a table with the files and description for each one.
3. Final conclusions (list of problems and solutions, etc)