Computation II: embedded system design (5EIB0)

Part 2d: 2022-04-21 Finite State Machine (Design Moore AC controller - 6pt)

Due date: Thursday, 21 April 2022, 5:45 PM
Requested files: AC_Controller.v (Download)
Type of work: Individual work

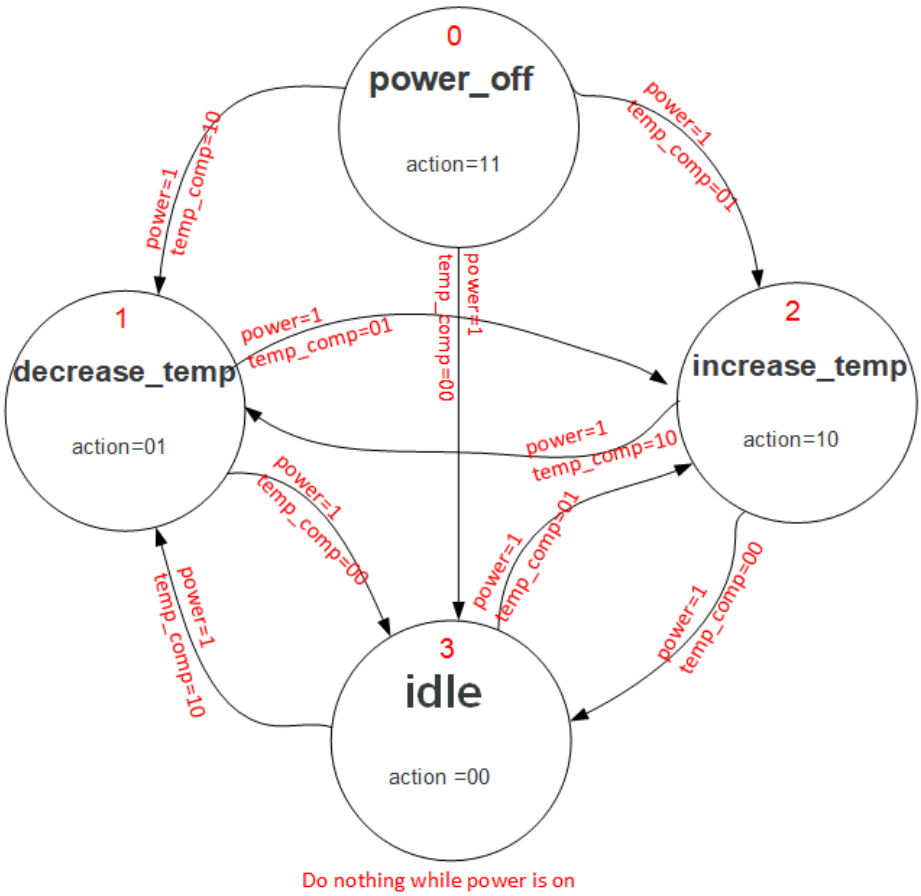
Introduction

In this assignment, you will create a Verilog implementation of a Finite State Machine (FSM), which controls an air conditioner (AC).

It has the following inputs and outputs:

- **power**, a one-bit input indicates if AC is on (=1) or off (=0).
- **temp_comp**, a two-bit input that encodes that the current room's temperature is lower (="01"), higher (="10"), or equal (="00") to the set temperature.
- **state_display**, a two-bit output that indicates the current state of acc
- **action**, a two-bit output that shows if AC needs to increase (="10") or decrease (="01") the temperature or do nothing (="00" in the idle state or "11" in the power-off state).

State diagram



The diagram above illustrates the transitions of the FSM that should be implemented to achieve this. It shows a Moore Machine where the output is defined only by the state. The machine has four states.

Assignment

Write an FSM module named **AC_Controller**, which implements the above state machine. Besides the input **power** and **temp_comp** and the outputs **state_display** and **action** as described above, it must have a clock input **clk** and a **reset** input.

The state machine must only transition to the next state (on rising edges of **clk**) when **power** is on (='1'); otherwise, it should return to the **power_off** state (the block diagram for the state machine does not include this transition due to clarity of the diagram). Consider that having a high reset (on the rising edge of **clk**, regardless of the other inputs) forces the state machine to go to the **power_off** state.

Debug

Click on the symbol marked below to see the waveforms produced by your design. Please note that if your code has an error that prevents it from being simulated, it will not produce any waveforms.

Requested files

AC_Controller.v

```
1  `timescale 1ns / 100ps
2
3
4  module AC_Controller (
5      input  wire [0:0] clk,
6      input  wire [0:0] reset,
7      input  wire [0:0] power,
8      input  wire [1:0] temp_comp,
9      output reg  [1:0] action,
10     output reg  [1:0] state_display
11     );
12
13   // Add your FSM implementation here...
14
15 endmodule
```

VPL

◄ Part 2c - 2022-04-21 Finite State Machine (Debug FSM Moore Sequence Detector - 4pt)

Jump to...

Part 2e: 2022-04-21 Design a parametrized counter - 6pt ►