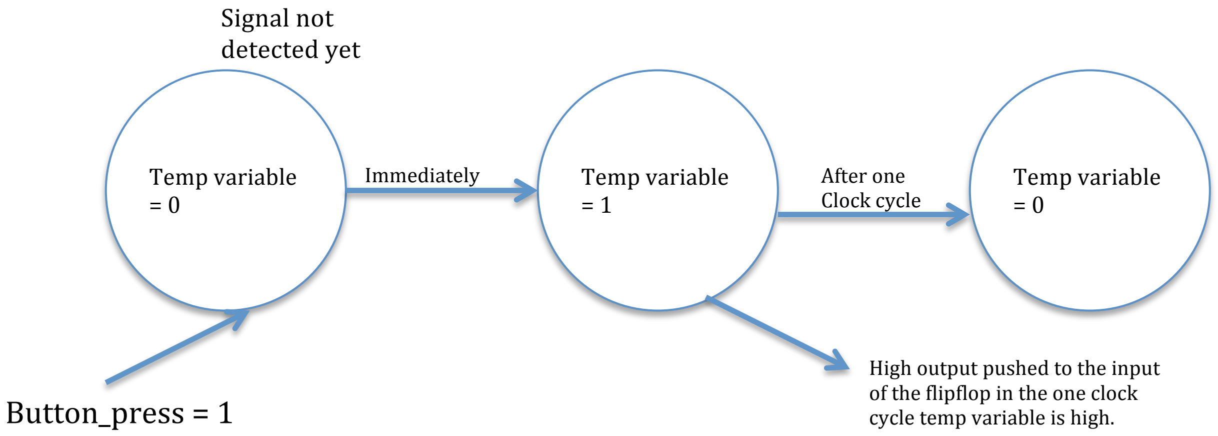
EE108a Lab 3 Prelab – Readme
Group 18, Fall 2012

a) If we are looking to design a "one-pulse" FSM that outputs a 1-cycle pulse every time the button is pressed, regardless of how long it is held down, then we have to implement a three-state FSM, as depicted below. The 1-cycle pulse is basically implemented in the same way that we implemented the "count-completion signal" in timer.v. We have a temporary variable (reg) that is set to 0 at the beginning of every clock cycle, and when the button is pressed, immediately comes on for the exact period of one clock cycle. The input of the flip flop is connected to this temp variable, and therefore gets set to high whenever temp variable comes on for one clock cycle.

b) If the case described in the prompt were to occur, where there was a Verilog case statement with no default entry, there will be an inferred latch, and synthesis will fail. If there were a default entry, the signal will enter the default case. If we appropriately write the default case such that we are able to quickly flag that there has been an instance of a default case (meaning that there is an error in the implementation), then we will be able to find the error and fix it.


Signal not detected yet
Temp variable = 0
Immediately
Temp variable = 1
After one Clock cycle
Temp variable = 0
Button_press = 1
High output pushed to the input of the flipflop in the one clock cycle temp variable is high.