Embedded Systems

Prof. Sedighi

Amirkabir University of Technology
(Tehran Polytechnic)

Department of Computer Engineering

Reza Adinepour ID: 402131055

Homework 2

Chapter 3 - Discrete Dynamics

March 25, 2024

Embedded Systems

Homework 2

Reza Adinepour ID: 402131055

Department of
Computer Engineering

Question 2

Consider a variant of the thermostat of example 3.5. In this variant, there is only one temperature threshold, and to avoid chattering the thermostat simply leaves the heat on or off for at least a fixed amount of time. In the initial state, if the temperature is less than or equal to 20 degrees Celsius, it turns the heater on, and leaves it on for at least 30 seconds. After that, if the temperature is greater than 20 degrees, it turns the heater off and leaves it off for at least 2 minutes. It turns it on again only if the temperature is less than or equal to 20 degrees.

1. Design an FSM that behaves as described, assuming it reacts exactly once every 30 seconds.

Solution

- initial state: **colling**
- if $temp \leq 20°C$ → heat on → wait for 30 S
- if $temp > 20°C$ → heat off → wait for 20 min on heating state

we can draw FSM as bellow:

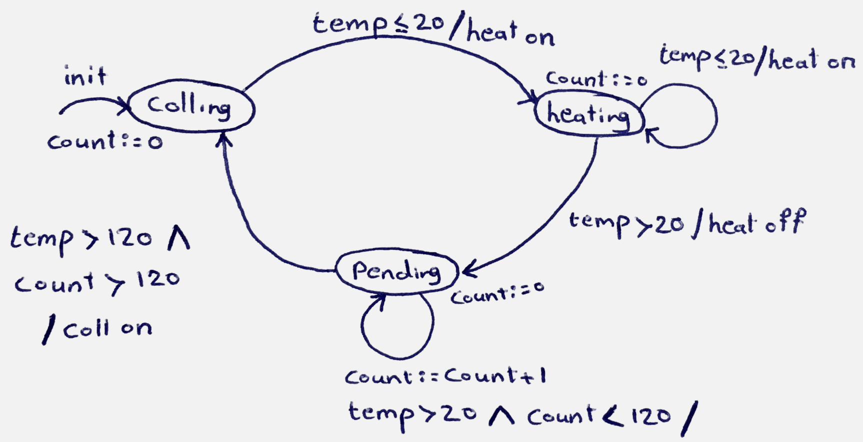


Figure 1: FSM solution of part 1

2. How many possible states does your thermostat have? Is this the smallest number of states possible?

Solution

I try to draw the smallest FSM for this problem. This FSM contain a 3 state.

3. Does this model thermostat have the time-scale invariance property?

Solution

The model does not have the hysteresis property because the timeout is a fixed amount of time, so varying the time scale of the input will yield distinctly different behavior.

Question 3

Consider the following state machine:

output: y: $\{0, 1\}$

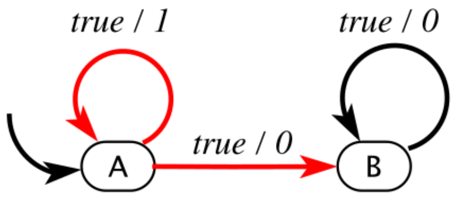


Figure 2: State machine of question 2

Determine whether the following statement is true or false, and give a supporting argument:

The output will eventually be a constant 0, or it will eventually be a constant 1. That is, for some n   N, after the n-th reaction, either the output will be 0 in every subsequent reaction, or it will be 1 in every subsequent reaction.

Note that Chapter 13 gives mechanisms for making such statements precise and for reasoning about them.

Solution

Due to non-deterministic behavior of the system in state A so the destination after this state is not clear but it could be seen that eventually it is possible to reaching the state B. In state B because of deterministic behavior, it remained in this stage.

Question 4

How many reachable states does the following state machine have?

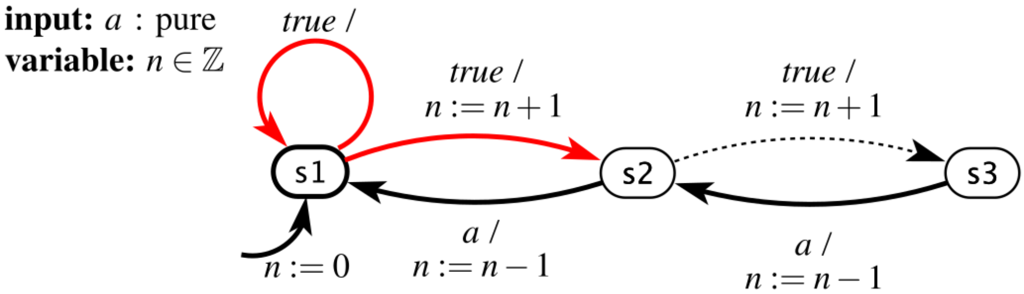


Figure 3: State machine of question 4

Solution

As we know the formula of reachable states define as follow:

$$|states| = np^m$$

So we have:

$$|states| = 3 \times 3^1 = 9$$

We have 9 reachable states in this state machine.

Question 5

Consider the deterministic finite-state machine in Figure 3.14 that models a simple traffic light.

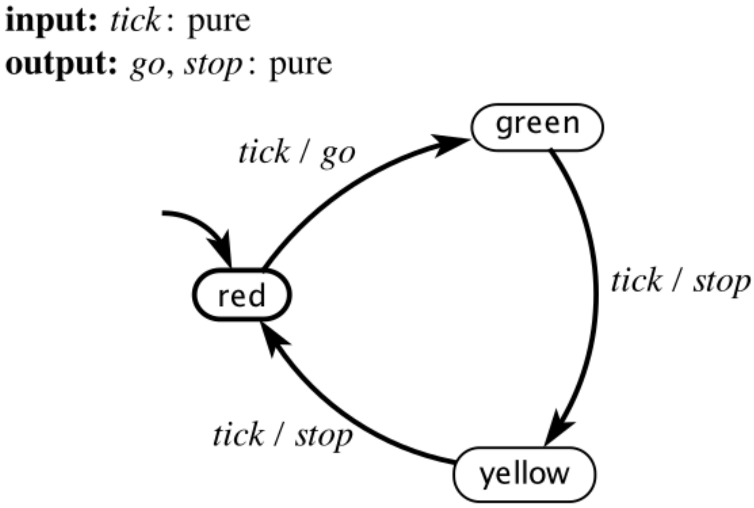


Figure 4: Deterministic finite-state machine for question 5

1. Formally write down the description of this FSM as a 5-tuple:
   (States, Inputs, Outputs, update, initialState)

Solution

States = $\{red,\ green,\ yellow\}$
Inputs = $\{tick \to \{present, absent\}\}$
Output = $\{\{go, stop\} \to \{present, absent\}\}$
Initial state = $\{red\}$
Update = $\{\{red, green\} \to \{tick = present\}, \{green, yellow\} \to \{tick = present\}, \{yellow, red\} \to \{tick = present\}\}$

2. Give an execution trace of this FSM of length 4 assuming the input *tick* is *present* on each reaction.

Solution

$red \xrightarrow{\text{tick=present/go}} green \xrightarrow{\text{tick=present/stop}} yellow \xrightarrow{\text{tick=present/stop}} red$

3. Now consider merging the red and yellow states into a single stop state. Tran sitions that pointed into or out of those states are now directed into or out of the new stop state. Other transitions and the inputs and outputs stay the same. The new stop state is the new initial state. Is the resulting state machine de terministic? Why or why not? If it is deterministic, give a prefix of the trace of length 4. If it is nondeterministic, draw the computation tree up to depth 4.

Solution

The new FSM as bellow:

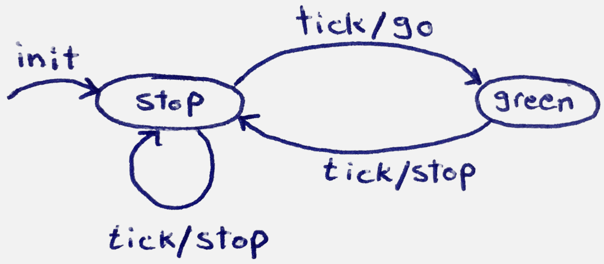


Figure 5: New FSM of part 3

This is the newest FSM. this is a non-deterministic FSM because two state of yellow and red are merged in new state of stop. so the stop state can do two transition with tick gard.

we can draw computation tree with depth 4 as bellow:

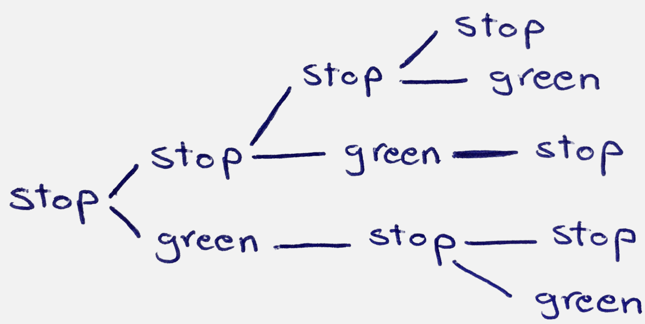
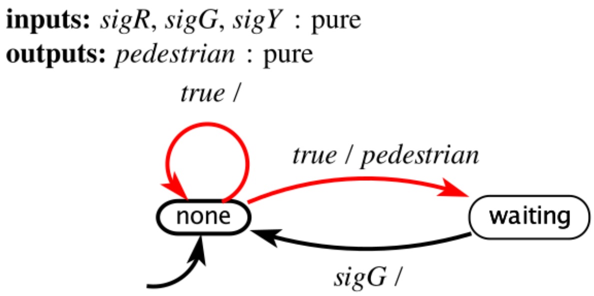


Figure 6: New FSM of part 3

Question 6

This problem considers variants of the FSM in Figure 3.11, which models arrivals of pedestrians at a crosswalk. We assume that the traffic light at the crosswalk is controlled by the FSM in Figure 3.10. In all cases, assume a time triggered model, where both the pedestrian model and the traffic light model react once per second. Assume further that in each reaction, each machine sees as inputs the output produced by the other machine *in the same reaction* (this form of composition, which is called synchronous composition, is studied further in Chapter 6).

1. Suppose that instead of Figure 3.11, we use the following FSM to model the arrival of pedestrians:

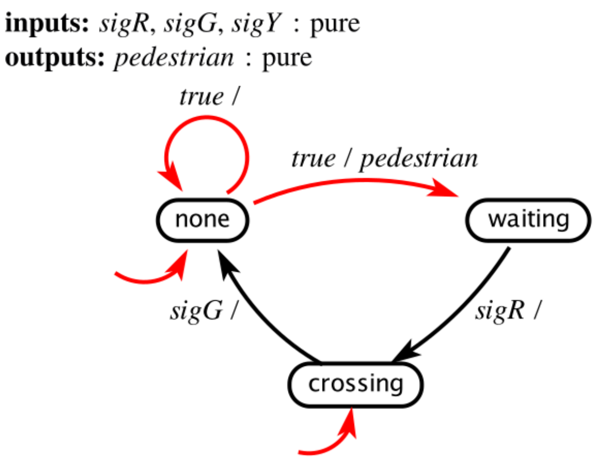


   Find a trace whereby a pedestrian arrives (the above machine transitions to waiting) but the pedestrian is never allowed to cross. That is, at no time after the pedestrian arrives is the traffic light in state red.

   Solution

   According to the pedestrian FSM, in its initial state it has a nondeterministic condition. Hence, we can suppose that, at first, we are in red and none states. Although the none state can repeat because of nondeterministic behavior, the red state after 60 seconds transfers to the green state. So, suppose that in the same time the red and none states transfer to the green and waiting states.

2. Suppose that instead of Figure 3.11, we use the following FSM to model the arrival of pedestrians:



   Here, the initial state is nondeterministically chosen to be one of none or crossing. Find a trace whereby a pedestrian arrives (the above machine tran sitions from none to waiting) but the pedestrian is never allowed to cross. That is, at no time after the pedestrian arrives is the traffic light in state red.

Solution

Suppose that we are in state none and red for pedestrian and traffic light respectively. At first, it could transition to waiting and red states because of nondeterministically behavior. After that, the red state transfer to the green state.

Question 7

Consider the state machine in Figure 7 State whether each of the following is a behavior for this machine. In each of the following, the ellipsis "..." means that the last symbol is repeated forever. Also, for readability, *absent* is denoted by the shorthand a and *present* by the shorthand p.

1. $x = (p, p, p, p, p, ...)$, $y = (0, 1, 1, 0, 0, ...)$

   Solution

   This is not behavior for this machine

2. $x = (p, p, p, p, p, ...)$, $y = (0, 1, 1, 0, a, ...)$

   Solution

   This is behavior for this machine

3. $x = (a, p, a, p, a, ...)$, $y = (a, 1, a, 0, a, ...)$

   Solution

   This is not behavior for this machine

4. $x = (p, p, p, p, p, ...)$, $y = (0, 0, a, a, a, ...)$

   Solution

   This is behavior for this machine

5. $x = (p, p, p, p, p, ...)$, $y = (0, a, 0, a, a, ...)$

   Solution

   This is behavior for this machine

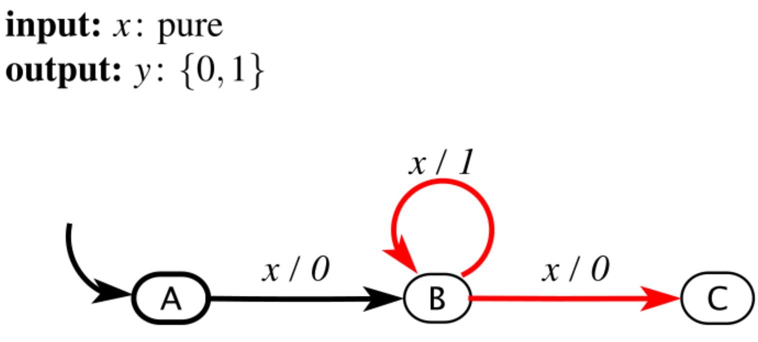


Figure 7: State machine for question 7

Question 8

(NOTE: This exercise is rather advanced.) This exercise studies properties of dis crete signals as formally defined in the sidebar on page 45. Specifically, we will show that discreteness is not a compositional property. That is, when combining two discrete behaviors in a single system, the resulting combination is not neces sarily discrete.

1. Consider a pure signal $x : \mathbb{R} \to \{present, absent\}$ given by:

$$x(t) = \begin{cases} present & \texttt{if } t \texttt{ is a non-negative integer} \\ absent & \texttt{otherwise} \end{cases}$$

   for all $t \in \mathbb{R}$. Show that this signal is discrete.

2. Consider a pure signal $y : \mathbb{R} \to \{present, absent\}$ given by:

$$y(t) = \begin{cases} present & \texttt{if } t = 1 - 1/n \texttt{ for any positive integer } n \\ absent & \texttt{otherwise} \end{cases}$$

   for all $t \in \mathbb{R}$. Show that this signal is discrete.

3. Consider a signal ω that is the merge of x and y in the previous two parts. That is, $w(t) = present$ if either $x(t) = present$ or $y(t) = present$, and is *absent* otherwise. Show that ω is not discrete.

4. Consider the example shown in Figure 3.1. Assume that each of the two signals *arrival* and *departure* is discrete. Show that this does not imply that the output *count* is a discrete signal.

Solution

I so sorry. I don't have any idea for this question.

End of Homework 2