SOEN331: Introduction to Formal Methods for Software Engineering
Assignment 1 on extended finite state machines

Tarek Ait Hamouda (40044119), Abhijit Gupta (40066502),
Ethel Narra Pangan (40061530)

February 11, 2019

1 Room temperature control formal specification

The EFSM of the room temperature control is the tuple $S = (Q, \Sigma_1, \Sigma_2, q_0, V, \Lambda)$, where

$Q = \{idle, warmup, configuration\}$

$\Sigma_1 = \{shutoff, setup, interrupt, after(3min), after(2min), after\ 1\ min\ inactive, cancel, completed\}$

$\Sigma_2 = \{fan\ on, fan\ off, furnace\ on, furnace\ off, prolonged\ beep\ sound, double\ beep\ sound, click\ sound\}$

$q_0 : idle$

$V : \{C, T, D, Tf, input\ list\}$

Λ: Transition specifications

1. $\rightarrow idle$
2. $idle \xrightarrow{\text{shut off/ (fan off; furnace off)}} off$
3. $idle \xrightarrow{\text{after(2min)[input list not empty; C} \geq \text{D]}} idle$
4. $idle \xrightarrow{\text{after(2min)[input list not empty; C} \leq \text{D-1] / (fan off; furnace on)}} warming\ up$
5. $idle \xrightarrow{\text{setup/(}beep\ sound\text{; }led\ light\ switch\ on\text{)}} configuration$
6. $warming\ up \xrightarrow{\text{after(3min)[T.F} < \text{D+1]}} warming\ up$
7. $warming\ up \xrightarrow{\text{after(3min)[T.F = D+1]/(fan on; furnace off; click sound)}} idle$
8. $warming\ up \xrightarrow{\text{interrupt/(furnace off; beep sound; led light switch on)}} configuration$

9. $configuration \xrightarrow{\text{after 1 min inactive / led light switch off}} idle$
10. $configuration \xrightarrow{\text{cancel/(prolonged beep sound; led light switch off)}} idle$
11. $configuration \xrightarrow{\text{completed/(double beep sound; led light switch off)}} idle$

The UML state diagram is shown in Figure 1.

The EFSM of the configuration is the tuple $S = (Q, \Sigma_1, \Sigma_2, q_0, V, \Lambda)$, where

$Q = \{input, add, override\}$

$\Sigma_1 = \{register, repeat\}$

$\Sigma_2 = \{\}$

$q_0 : input$

$V : triplet$

Λ: Transition specifications

1. $\rightarrow input$
2. $input \xrightarrow{\text{register[triplet does not exist in input list]}} add$
3. $input \xrightarrow{\text{register[triplet exists in input list]}} override$
4. $add \xrightarrow{\text{repeat}} input$
5. $override \xrightarrow{\text{repeat}} input$

The UML state diagram is shown in Figure 2.

2 UML state diagrams

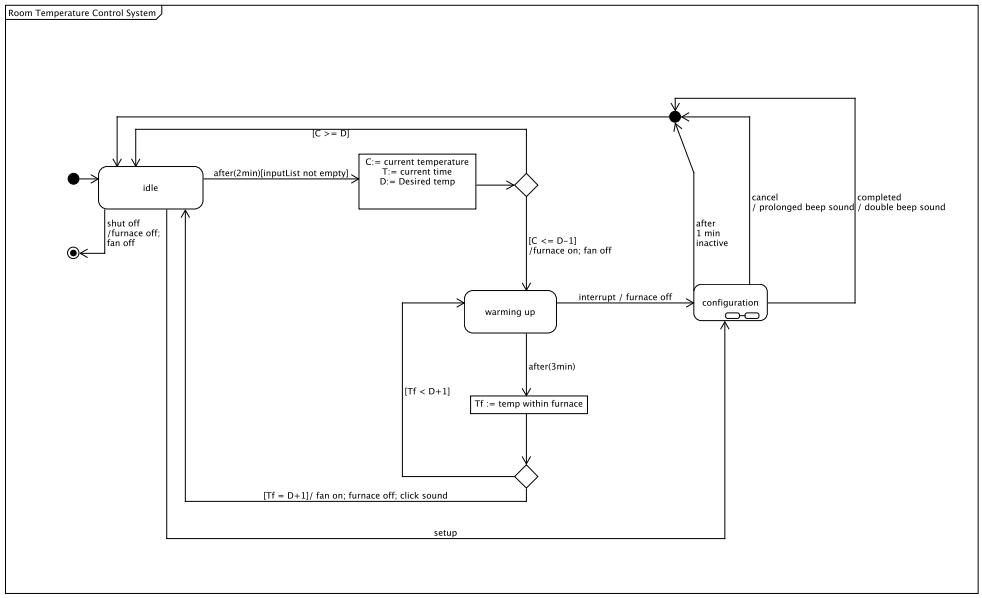

Figure 1: Room Temperature Control System UML State Diagram

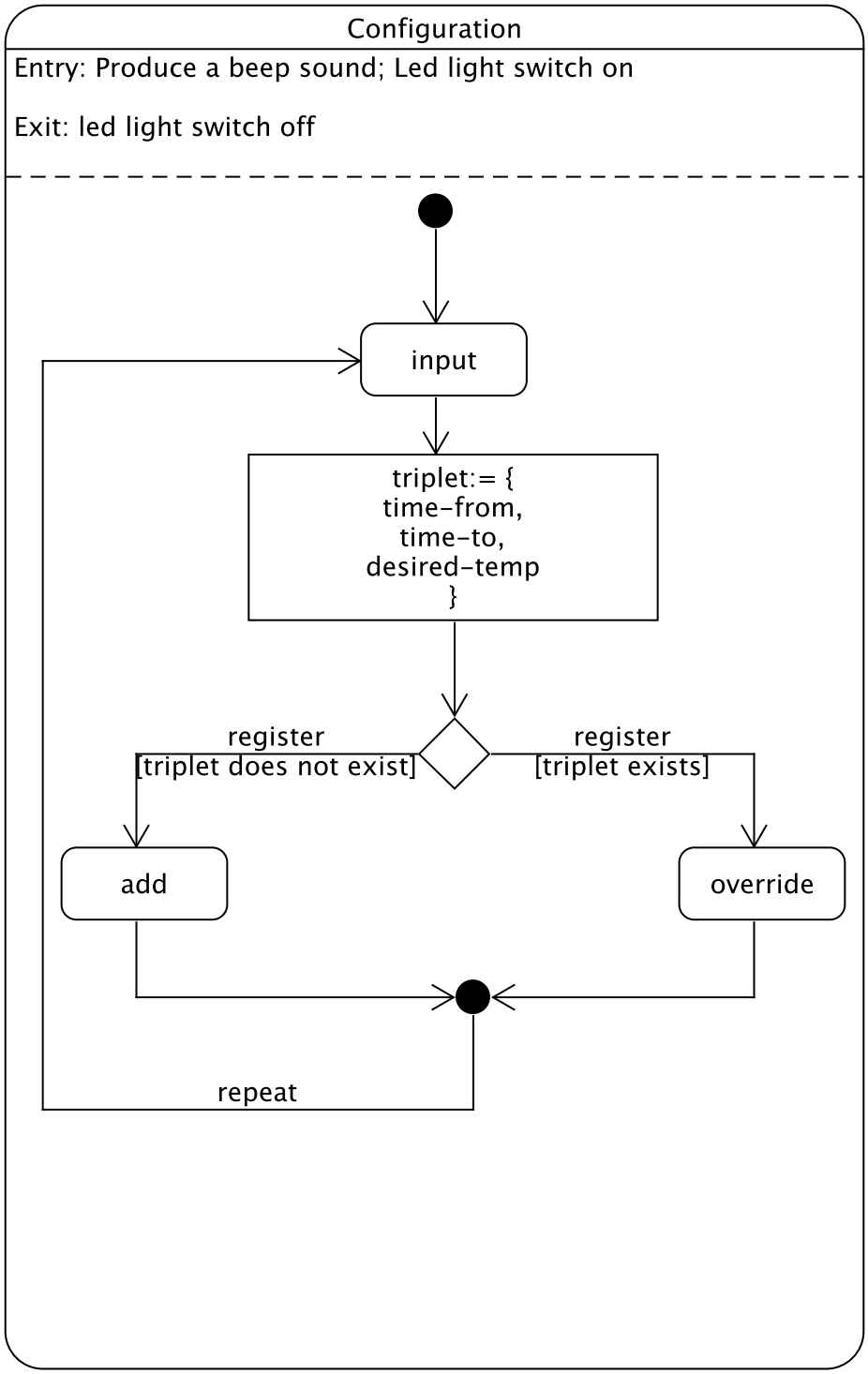

Figure 2: Configuration UML State Diagram