Assignment 3 - SOEN 331 Section U-UB
By Harrison Ianatchkov ID: 26607403 and Justin Yip ID: 27032870
Due March 30, 2015

Safety Critical System

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

Q = {dormant, safe_shutdown, init, idle, monitoring, error_diagnosis}
Σ_1 = {kill, sleep, start, init_ok, begin_monitoring, idle_rescue, idle_crash, retry_init, monitor_crash, monitor_rescue, init_crash, shutdown}
Σ_2 = {broadcast idle_err_msg, retry++, broadcast moni_err_msg, broadcast init_err_msg, system clean up}
q_0 : dormant
V : retry : $\mathbb{N}_0$; inlockdown : Boolean.
Λ : Transition specifications

1. $\rightarrow dormant$
2. $dormant \xrightarrow{kill} exit$
3. $dormant \xrightarrow{start\ /\ retry = 0} init$
4. $init \xrightarrow{init\_ok} idle$
5. $init \xrightarrow{init\_crash/\ broadcast\ init\_err\_msg} error\_diagnosis$
6. $idle \xrightarrow{begin\_monitoring\ /\ inlockdown = false} monitoring$
7. $idle \xrightarrow{idle\_crash\ /\ broadcast\ idle\_err\_msg} error\_diagnosis$
8. $monitoring \xrightarrow{monitor\_crash\ [not(inlockdown)]\ /\ broadcast\ moni\_err\_msg} error\_diagnosis$
9. $error\_diagnosis \xrightarrow{retry\_init\ [retry < 3]\ /\ retry++} init$
10. $error\_diagnosis \xrightarrow{idle\_rescue} idle$
11. $error\_diagnosis \xrightarrow{moni\_rescue} monitoring$
12. $error\_diagnosis \xrightarrow{shutdown\ [retry \geq 3]\ /\ system\ clean\ up} safe\_shutdown$
13. $safe\_shutdown \xrightarrow{sleep} dormant$

init

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

Q = {boot_hw, senchk, tchk, psichk, ready}
Σ_1 = {hw_ok, senok, t_ok, psi_ok}
Σ_2 = {}
q_0 : boot_hw
V = {}
Λ: Transition specifications

1. $\rightarrow boot\_hw$
2. $boot\_hw \xrightarrow{hw\_ok} senchk$
3. $senchk \xrightarrow{sen\_ok} tchk$
4. $tchk \xrightarrow{t\_ok} psichk$
5. $psichk \xrightarrow{psi\_ok} ready$

lockdown

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

Q = {prep_vpurge, alt_temp, alt_psi, risk_assess, safe_status}
Σ_1 = {initiate_purge, tcyc_comp, psicyc_comp}
Σ_2 = {lock_doors, unlock_doors}
q_0 : prep_vpurge
V : risk: percentage
Λ : Transition specifications

1. $\rightarrow prep\_vpurge$
2. $prep\_vpurge \xrightarrow{initiate\_purge/lock\_doors} alt\_temp$
3. $prep\_vpurge \xrightarrow{initiate\_purge/lock\_doors} alt\_psi$
4. $alt\_temp \xrightarrow{tcyc\_comp} risk\_assess$
5. $alt\_psi \xrightarrow{psicyc\_comp} risk\_assess$
6. $risk\_assess \xrightarrow{[risk \geq 1\%]} prep\_vpurge$
7. $risk\_assess \xrightarrow{[risk < 1\%]} safe\_status$
8. $safe\_status \rightarrow exit$

error_diagnosis

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

Q = {error_rcv, applicable_rescue, reset_module_data}
Σ_1 = {apply_protocol_rescues, reset_to_stable}
Σ_2 = { }
q_0 : error_rcv
V : err_protocol_def: Boolean
Λ : Transition specifications

1. $\rightarrow error\_rcv$
2. $error\_rcv \xrightarrow{[err\_protocol\_def]} applicable\_rescue$
3. $error\_rcv \xrightarrow{[err\_protocol\_def]} reset\_module\_data$
4. $reset\_module\_data \xrightarrow{reset\_to\_stable} exit$
5. $applicable\_rescue \xrightarrow{apply\_protocol\_rescues} exit$

monitoring

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

Q = {monidle, regulate_environment, lockdown}
Σ_1 = {no_contagion, after_100ms, purge_succ, contagion_alert}
Σ_2 = {inlockdown = false, broadcast FACILITY_CRIT_MESG and inlockdown = true}
q_0 : monidle
V = {}
Λ : Transition specifications

1. $monidle \xrightarrow{no\_contagion} regulate\_environment$
2. $monidle \xrightarrow{contagion\_alert \,/\, broadcast\ FACILITY\_CRIT\_MESG\ and\ inlockdown = true} lockdown$
3. $regulate\_environment \xrightarrow{after\_100ms} monidle$
4. $lockdown \xrightarrow{purge\_succ \,/\, inlockdown = false} monidle$