***Part 1: Overall System***

Q = {dormant, init, idle, monitoring, error_diagnosis, safe_shutdown}
$q_o$ : dormant
$\Sigma_1$ = {kill, start, init_ok, init_crash, retry_init, begin_monitoring, idle_crash, idle_rescue, monitor_ crash, moni_rescue, shutdown, sleep}
$\Sigma_2$ = {activateDrivers, init_err_msg, triggerShutdown, incr_retry, idle_err_msg, moni_err_msg}
V = {driversLoaded : Boolean, retry : $\mathbb{R}$ = 3}
Λ = {

→ dormant

dormant $\xrightarrow{\text{kill}}$ exit

dormant $\xrightarrow{\text{start / activateDrivers}}$ init

init $\xrightarrow{\text{init\_ok [driversLoaded]}}$ idle

init $\xrightarrow{\text{init\_crash / init\_err\_msg,trigger\_shutdown}}$ error_diagnosis

error_diagnosis $\xrightarrow{\text{retry\_init [retry < 3] / incr\_retry}}$ init

idle $\xrightarrow{\text{begin\_monitoring}}$ monitoring

idle $\xrightarrow{\text{idle\_crash / idle\_err}}$ error_diagnosis

error_diagnosis $\xrightarrow{\text{idle\_rescue}}$ idle

monitoring $\xrightarrow{\text{monitor\_crash / moni\_err\_msg}}$ error_diagnosis

error_diagnosis $\xrightarrow{\text{moni\_rescue}}$ monitoring

error_diagnosis $\xrightarrow{\text{shutdown [retry > 3]}}$ safe_shutdown

safe_shutdown $\xrightarrow{\text{sleep}}$ dormant

}

***Part 2: Refine init***

Q = {boot_hw, senchk, tchk, psichk, ready}
$q_o$ : boot_hw
$\Sigma_1$ = {hw_ok, senok, t_ok, psi_ok}
$\Sigma_2$ = {check_sensors, check_t, check_psi}
V = {}
Λ = {

→ boot_hw

boot_hw $\xrightarrow{\text{hw\_ok / check\_sensors}}$ senchk

senchk $\xrightarrow{\text{senok / check\_t}}$ tchk

tchk $\xrightarrow{\text{t\_ok / check\_psi}}$ psichk

psichk $\xrightarrow{\text{psi\_ok}}$ ready

}

***Part 3: Refine monitoring***

Q = {monidle, regulate_environment, lockdown}
$q_o$ : monidle
$\Sigma_1$ = {no_contagion, after_100ms, contagion_alert, purge_succ}
$\Sigma_2$ = {FACILITY_CRIT_MESG, set_inlockdown}
V = {inlockdown : Bool = False}
Λ = {

→ monidle

monidle $\xrightarrow{\text{no\_contagion}}$ regulate_environment

monidle $\xrightarrow{\text{contagion\_alert / FACILITY\_CRIT\_MESG,set\_inlockdown}}$ lockdown

regulate_environment $\xrightarrow{\text{after\_100ms}}$ monidle

lockdown $\xrightarrow{\text{purge\_succ / unset\_inlockdown}}$ monidle

monidle $\xrightarrow{[\text{inlockdown == false}]}$ exit

}

***Part 4: Refine lockdown***

Q = {prep_vpurge, alt_temp, alt_psi, risk_assess, safe_status}
$q_o$ : prep_vpurge
$\Sigma_1$ = {initiate_purge, tcyc_comp, psicyc_comp, risk_checked}
$\Sigma_2$ = {lock_doors, unlock_doors}
V = {risk : $\mathbb{R}$}
Λ = {

→ prep_vpurge

prep_vpurge $\xrightarrow{initiate\_purge \,/\, lock\_doors}$ alt_temp

prep_vpurge $\xrightarrow{initiate\_purge \,/\, lock\_doors}$ alt_psi

alt_temp $\xrightarrow{\text{tcyc\_comp}}$ risk_assess

alt_psi $\xrightarrow{\text{psicyc\_comp}}$ risk_assess

risk_assess $\xrightarrow{\text{risk\_checked } [\text{risk} >= 0.01]}$ prep_vpurge

risk_assess $\xrightarrow{\text{risk\_checked } [\text{risk} < 0.01] \,/\, \text{unlock\_doors}}$ safe_status

}

***Part 5: Refine error_diagnosis***

Q = {error_rcv, applicable_rescue, reset_module_data, final_state}
$q_o$ : error_rcv
$\Sigma_1$ = {apply_protocol_rescues, reset_to_stable}
$\Sigma_2$ = {}
V = {err_protocol_def : Bool}
Λ = {

→ error_rcv

error_rcv $\xrightarrow{[\text{err\_protocol\_def} == \text{True}]}$ applicable_rescue

applicable_rescue $\xrightarrow{\text{apply\_protocol\_rescues}}$ final_state

error_rcv $\xrightarrow{[\text{err\_protocol\_def} == \text{False}]}$ reset_module_data

reset_module_data $\xrightarrow{\text{reset\_to\_stable}}$ final_state

}