CROB Logic Flow

Robert Goodwin

Fri, Sep 4, 2009

The program CROB follows state machine logic. This note describes its logic flow:

```
Parameters to CROB:
       Enable Bit#           { Usual enable bit for local appl }
       grad Chan#     0102   { GRxMID gradient reading }
       paRev Chan#    0105   { pa reverse power reading }
       crobar Bit#    0115   { crobar trip status }
       inhib Bit#     010F   { permanent inhibit status }
       HVPS Bit#      0117   { HVPS status }
       ovRdy Bit#     010B   { overload reset ready status }
       sysHV Bit#     0116   { system HV status }
       sysReset Bit#  0195   { system reset control }
       hvOn Bit#      0197   { HV ON control }

Every 15 Hz cycle,

IF trip_window > 0, trip_window--
both_set = (inhib = 1) and (crobar = 1)

{ Note: Initial state is State 0. Normal state is State 1. }

State 0:       { await calm, advance to State 1 }
       IF NOT both_set AND (sysHV = 0) AND (HVPS = 0)
              set trip_window = 0, wait 1 sec, advance to State 1

State 1:       { normal state, await possible crobar trip }
       grad = grad_reading
       IF both_set, wait 2 sec, advance to State 2

       ELSE IF (grad > 0.8 v) AND (grad < nominal*1.05)
              grad_2back = grad_1back, grad1_back = grad

State 2:       { crobar problem, start recovery }
       IF both_set,
              Adjust gradient by -800 steps
              IF trip_window = 0, trip_window = 5*60*15 {5 minutes}, trip_count = 0
              trip_count++
              wait 10 sec, advance to State 3

       ELSE go back to State 1 { crobar recovered, maybe by human }

State 3:       { reset HV if needed, continue }
       IF NOT both_set, go back to State 1 { unexpected case, await next trip }

       ELSE If ovRdy = 1
              IF HVPS = 1, go back to State 0 { HV tripped, give up }

              ELSE    IF trip_count < 4
                             sysReset pulse hi 0.5 sec
                             wait 1 sec, advance to State 4

                      ELSE go back to State 0  { give up automatic recovery }
```

```
State 4:      { if resets worked, start bringing gradient back up }
       IF HVPS = 0
              IF sysHV = 1
                     HVON pulse hi 0.5 sec, wait 0.5 sec

              ELSE
                     adj_type = 50 steps, adj_count = 16, coarse = true,
                     wait 4 sec, advance to State 5

       ELSE wait 2 sec, go back to State 0 { nothing more can be done }

State 5:      { step by step adjustments to the gradient }
       IF both_set, wait 2 sec, go back to State 2

       ELSE   IF paRev < 0.2 v
                     grad = grad_reading
                     IF (course AND (grad < 0.8 v)) OR (NOT coarse AND (grad < grad_2back))
                            adjust gradient by adj_type steps
                            wait 1 sec
                            adj_count--
                            IF adj_count = 0
                                   IF grad < 0.1 v
                                          go back to State 2 { gradient lost }

                                   ELSE adj_count = 2 { try a couple more times }

                     ELSE   IF NOT coarse
                                   go back to State 1

                            ELSE coarse = false, adj_type = 10 steps
```

In simple terms, CROB watches for crobar trips, defined as both `inhib` and `crobar` bits set. When it detects a crobar trip, it turns down the gradient, waits for the `ovRdy` status, resets the RF system, and brings the gradient back up in stages. At first, it sends bursts of 50 steps, called coarse control. Then, after reaching a gradient of 0.8 volts, it uses bursts of 10 steps, called fine control. While bringing up the gradient, it watches for the reverse power to remain below 0.2 volts, else it holds back before issuing the next burst of steps. When the gradient reaches the level that existed before the trip occurred, it returns to await the next crobar trip.