```
SUBDESIGN dx35
(
    clk, reset, operate     : INPUT;
    beam_0_inN              : INPUT;
    beam_0_inNNEG           : INPUT;
    beam_0_inNPOS           : INPUT;
    word_0_inN              : INPUT;
    word_0_inNNEG           : INPUT;
    word_0_inNPOS           : INPUT;
    er_flag_inN             : INPUT;
    er_flag_inNEG           : INPUT;
    er_flag_inPOS           : INPUT;
    nabr_pos_wire_in[10..0] : INPUT;
    nabr_neg_wire_in[10..0] : INPUT;
    core_wire_in[23..0]     : INPUT;
    wire_dead[69..0]        : INPUT;
    beam_0_out              : OUTPUT;
    word_0_out              : OUTPUT;
    error_flag              : OUTPUT;
    W0_error_flag           : output;
    B0_error_flag           : output;
    w_out[139..0]           : OUTPUT;
    time_0                  : OUTPUT;
    OUT_ENA                 : OUTPUT;

)
VARIABLE
    demux1_4: MACHINE WITH STATES (S0,S1,S2,S3);
    w_out[139..0]   :dffe;
    w_temp[104..0]  :dffe;
    out_ena         :dff;
    W0_error_flag   :dffe;
    B0_error_flag   :dffe;
    temp_error_flag :dff;
    beam_zero[2..0] :dffe;
    error_flag      :dffe;
    beam_0_in       :node;
    beam_0_inEXC_OR :node;
    beam_0_out      :dffe;
    word_0_out      :dff;
    time_0          :dff;
    error_out_enable :dffe;
    error_flag_in   :node;

BEGIN
    time_0.clk          = !clk;
    time_0.clrn         = !reset;
    out_ena.clk         = clk;
    out_ena.clrn        = !reset;
    beam_zero[].clk     = clk;
    beam_zero[].clrn    = !reset;
    word_0_out.clk      = clk;
    word_0_out.clrn     = !reset;
    W0_error_flag.clk   = clk;
    W0_error_flag.clrn  = !reset;
    B0_error_flag.clk   = clk;
    B0_error_flag.clrn  = !reset;
    error_flag_in       = er_flag_inN # er_flag_inNEG # er_flag_inPOS;
    error_flag.clk      = clk;
    error_flag.clrn     = !reset;
    temp_error_flag.clk = clk;
    temp_error_flag.clrn = !reset;
    w_out[].clk         = clk;
    w_out[].clrn        = !reset;
```

```
    w_temp[].clk        = clk;
    w_temp[].clrn       = !reset;
    demux1_4.clk        = clk;
    demux1_4.reset      = reset;
    w_out[104..0].d     = w_temp[104..0].q;
    beam_0_in           = beam_0_inN & beam_0_inNNEG & beam_0_inNPOS;
    beam_0_inEXC_OR     = (beam_0_inN & beam_0_inNNEG & beam_0_inNPOS) # (!beam_0_inN &
!beam_0_inNNEG & !beam_0_inNPOS);
    beam_0_out.clk      = clk;
    beam_0_out.clrn     = !reset;
    error_out_enable.d  = operate & word_0_inN;
    error_out_enable.ena = operate & word_0_inN;
    error_out_enable.clk = clk;
    error_out_enable.clrn = !reset;
    W0_error_flag.ena   = error_out_enable.q;
    B0_error_flag.ena   = error_out_enable.q;

    CASE demux1_4 IS
        WHEN S0 =>
                time_0.d = gnd;
                beam_zero[0].ena = vcc;
                beam_zero[2..1].ena = gnd;
                beam_zero[0].d = beam_0_in;
                beam_0_out.ena = gnd;

                word_0_out.d = gnd;

                W0_error_flag.d = !(word_0_inNNEG & word_0_inN & word_0_inNPOS);
                B0_error_flag.d = !beam_0_inEXC_OR;
                temp_error_flag.d = error_flag_in;
                error_flag.ena = gnd;

                OUT_ENA.d = gnd;
                w_out[].ena = gnd;
                w_temp[34..0].ena = vcc;
                w_temp[104..35].ena = gnd;
                w_temp[23..0].d = core_wire_in[23..0]  # wire_dead[23..0];
                w_temp[34..24].d = nabr_pos_wire_in[10..0] # wire_dead[58..48];

            if operate & word_0_inN then
                demux1_4 = s1;
            else
                demux1_4 = s0;
            end if;

        WHEN S1 =>
            time_0.d = gnd;
            beam_zero[0].ena = gnd;
            beam_zero[1].ena = vcc;
            beam_zero[2].ena = gnd;
            beam_zero[1].d = beam_0_in;
            beam_0_out.ena = gnd;

            word_0_out.d = gnd;

            W0_error_flag.d = (word_0_inNNEG # word_0_inN # word_0_inNPOS);
            B0_error_flag.d = !beam_0_inEXC_OR;
            temp_error_flag.d = error_flag_in # temp_error_flag.q # B0_error_flag.q #
W0_error_flag.q;
            error_flag.ena = gnd;

            w_out[].ena = gnd;
            OUT_ENA.d = gnd;
            w_temp[69..35].ena = vcc;
```

```
            w_temp[104..70].ena = gnd;
            w_temp[34..0].ena = gnd;
            w_temp[69..46].d = core_wire_in[23..0]  # wire_dead[47..24];
            w_temp[45..35].d = nabr_neg_wire_in[10..0] # wire_dead[69..59];
            demux1_4 = s2;

        WHEN S2 =>
            time_0.d = vcc;
            beam_zero[1..0].ena = gnd;
            beam_zero[2].ena = vcc;
            beam_zero[2].d = beam_0_in;
            beam_0_out.ena = gnd;

            word_0_out.d = gnd;

            W0_error_flag.d = (word_0_inNNEG # word_0_inN # word_0_inNPOS) ;
            B0_error_flag.d = !beam_0_inEXC_OR;
            temp_error_flag.d = error_flag_in # temp_error_flag.q # B0_error_flag.q #
W0_error_flag.q;
            error_flag.ena = gnd;

            OUT_ENA.d = gnd;
            w_out[].ena = gnd;
            w_temp[104..70].ena = vcc;
            w_temp[69..0].ena = gnd;
            w_temp[93..70].d = core_wire_in[23..0]  # wire_dead[23..0];
            w_temp[104..94].d = nabr_pos_wire_in[10..0] # wire_dead[58..48];
            demux1_4 = s3;

        WHEN S3 =>
            time_0.d = gnd;
            beam_zero[2..0].ena = gnd;
            beam_0_out.ena = vcc;
            beam_0_out.d = beam_zero[0] & beam_zero[1] & beam_zero[2] & beam_0_in;

            word_0_out.d = vcc;

            W0_error_flag.d = word_0_inNNEG # word_0_inN # word_0_inNPOS;
            B0_error_flag.d = !beam_0_inEXC_OR;
            temp_error_flag.d = gnd;
            error_flag.ena = vcc;
            error_flag.d = error_flag_in # temp_error_flag.q # B0_error_flag.q #
W0_error_flag.q ;

            OUT_ENA.d = vcc;
            w_out[].ena = vcc;
            w_temp[104..0].ena = gnd;
            w_out[139..116].d = core_wire_in[23..0] # wire_dead[47..24];
            w_out[115..105].d = nabr_neg_wire_in[10..0] # wire_dead[69..59];
            demux1_4 = s0;

    END CASE;
END;
```