```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 11/12/2020 03:57:20 PM
// Design Name:
// Module Name: Control_Logic
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////


module Control_Logic(
    input Go, Stop, Match, TwoSec, FourSec,
    input [4:0] PS,
```

```
    output [4:0] NS,
    output ShowNum, ResetTimer, RunGame,
Scored, FlashBoth, FlashAlt
    );

    wire START, NRND, Count, WIN, LOSE;
    wire Next_START, Next_NRND, Next_Count,
Next_WIN, Next_LOSE;

    // Present State
    assign START    = PS[0];
    assign NRND     = PS[1];
    assign Count    = PS[2];
    assign WIN      = PS[3];
    assign LOSE     = PS[4];

    // Next State
    assign NS[0] = Next_START;
    assign NS[1] = Next_NRND;
    assign NS[2] = Next_Count;
    assign NS[3] = Next_WIN;
    assign NS[4] = Next_LOSE;

    //Enter Logic
    assign Next_START = START&~Go |
WIN&FourSec | LOSE&FourSec;
    assign Next_NRND  = START&Go |
```

```
NRND&~TwoSec;
    assign Next_Count = NRND&TwoSec |
Count&~Stop;
    assign Next_WIN   = (Count&Stop&Match) |
(WIN&~FourSec);
    assign Next_LOSE  = (Count&Stop&~Match) |
(LOSE&~FourSec);

    // Outputs
    assign ShowNum = ~START;
    assign ResetTimer =
NS[1];//Next_NRND|Next_WIN|Next_LOSE;
    assign RunGame = Count;
    assign FlashBoth = WIN&Match&~FourSec;
    assign FlashAlt = LOSE&~Match&~FourSec;
    assign Scored = WIN&Match;

endmodule
```