```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 04/13/2022 07:54:21 PM
// Design Name:
// Module Name: Top_level
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module Top_level(
    input [7:0] sw,
    input btnL,
    input btnC,
    input btnR,
    input clkin,

    output [6:0] seg,
    output  dp,
    output [3:0] an

    );
    wire [7:0] w;
    wire [7:0] seg1;
    wire [7:0] seg2;
    wire dig_sel;
    wire [7:0] foo;

    //passing buttons through the incrementer
    Incrementer i1 ( .a(sw[7:0]), .b({btnL, btnC}), .s(w[7:0]));
    //assigning the outputs of the incrementer to the seven seg displays
    //wire [3:0] test;
  //  wire [3:0] test1;
   // assign test = 4'b0010 ;
```

```
//  assign test1 = 4'b1010 ;
  SevSeg s1 ( .n(w[3:0]), .seg(seg1[6:0]));
  SevSeg s2 ( .n(w[7:4]), .seg(seg2[6:0]));

  //part 4

  m2_1x8 m1 ( .in0(seg1[7:0]), .in1(seg2[7:0]), .sel(dig_sel), .o(foo[7:0]));
  //making the output of our m2 mux into a 7 bit from an 8 bit
  assign seg[6:0] = foo[6:0];
  lab3_digsel d1 ( .clkin(clkin), .greset(btnR), .digsel(dig_sel));

  assign an[1] = ~dig_sel;
  assign an[0] = dig_sel;
  assign an[2] = 1'b1;
  assign an[3] = 1'b1;
  assign dp = 1'b1;



endmodule
```