EENG 284

–

Digital Design Lab

Lab 6

–

Basic Calculator

Lab Solutions

**Table 1:**

**A screenshot of a table

Description automatically generated**

**Table 2:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4-bit input x | *interp = 0* Unsigned | | *interp = 1* Signed | |
| msDisplay | lsDisplay | msDisplay | lsDisplay |
| 4’b0000 | blank | 0 | blank | 0 |
| 4’b0001 | blank | 1 | blank | 1 |
| 4’b0010 | blank | 2 | blank | 2 |
| 4’b0011 | blank | 3 | blank | 3 |
| 4’b0100 | blank | 4 | blank | 4 |
| 4’b0101 | blank | 5 | blank | 5 |
| 4’b0110 | blank | 6 | blank | 6 |
| 4’b0111 | blank | 7 | blank | 7 |
| 4’b1000 | blank | 8 | - | 8 |
| 4’b1001 | blank | 9 | - | 7 |
| 4’b1010 | 1 | 0 | - | 6 |
| 4’b1011 | 1 | 1 | - | 5 |
| 4’b1100 | 1 | 2 | - | 4 |
| 4’b1101 | 1 | 3 | - | 3 |
| 4’b1110 | 1 | 4 | - | 2 |
| 4’b1111 | 1 | 5 | - | 1 |

**Listing 2:**

if ( (interp == 0) && (x < 10) ) { // y0 input

msDisplay = BLANK lsDisplay = x

} else if ( (interp == 0) && (x >= 10) ) { // y1 input

msDisplay = ‘1’ lsDisplay = X-10

} else if ( (interp == 1) && (x >= 0) ) { // y0 input

msDisplay = BLANK lsDisplay = X

} else if ( (interp == 1) && (x < 0) ) { // y2 input

msDisplay = ‘-‘ lsDisplay = 0-X

}

**Figure 4:**

**A diagram of a computer

Description automatically generated**

**Table 3:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| input | y3 | y2 | y1 | y0 |
| digSel | 2’b11 | 2’b10 | 2’b01 | 2’b00 |
| msDisplay | “X” | - | 1 | blank |
| lsDisplay | “X” | 0-x | x-10 | x |

**Table 4:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ovf | interp | sign | xGE10 | digSel |
| 1 | x | x | x | 2’b11 |
| 0 | 0 | x | 0 | 2’b00 |
| 0 | 0 | x | 1 | 2’b01 |
| 0 | 1 | 0 | x | 2’b00 |
| 0 | 1 | 1 | x | 2’b10 |

**Verilog Code for sigUnsig**

module sigUnsig(x, interp, ovf, msDisplay, lsDisplay);

input wire [3:0] x;

input wire interp;

input wire ovf;

output wire [6:0] msDisplay, lsDisplay;

reg [1:0] muxSel;

wire [6:0] displayRawX, displayNegativeX, displayXminus10;

wire [3:0] xMinus10, negativeX;

wire xGE10;

localparam [6:0] displayBlank = 7'b1111111 ;

localparam [6:0] displayOne = 7'b1111001 ;

localparam [6:0] displayMinus = 7'b0111111 ;

localparam [6:0] displayX = 7'b0001001 ;

genericMux4x1 #(7) muxMsDisplay (displayX,displayMinus, displayOne, displayBlank, muxSel, msDisplay);

genericMux4x1 #(7) muxLsDisplay (displayX, displayNegativeX, displayXminus10, displayRawX, muxSel, lsDisplay);

sevenSegment xSeg (x, displayRawX);

genericAdderSubtractor #(4) xMinus10addSub (x, 4'b1010, 1'b1, xMinus10, );

sevenSegment xMinus10Seg (xMinus10, displayXminus10);

genericAdderSubtractor #(4) negativeXaddSub (4'b0000, x, 1'b1, negativeX, );

sevenSegment negativeXdisplay (negativeX, displayNegativeX);

genericComparator #(4) compareXand10 (x, 4'b1001, xGE10, , );

always @(\*)

casez ({ovf, interp, xGE10, x[3]})

4'b000?: muxSel = 2'b00;

4'b001?: muxSel = 2'b01;

4'b01?1: muxSel = 2'b10;

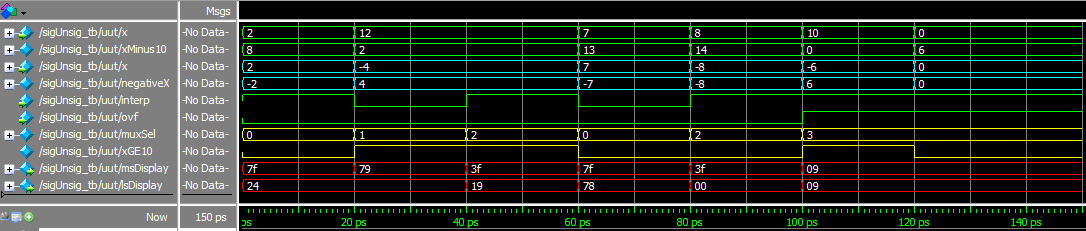
4'b01?0: muxSel = 2'b00;

default: muxSel = 2'b11;

endcase

endmodule

**Testbench:**



**Pin Assignments:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Segment | msXdisplay | lsXdisplay | msYorRESdisplay | lsYorRESdisplay |
| seg[6] | PIN\_AC22 | PIN\_W20 | PIN\_AF24 | PIN\_Y18 |
| seg[5] | PIN\_AC23 | PIN\_W21 | PIN\_AC19 | PIN\_Y19 |
| seg[4] | PIN\_AC24 | PIN\_V20 | PIN\_AE25 | PIN\_Y20 |
| seg[3] | PIN\_AA22 | PIN\_V22 | PIN\_AE26 | PIN\_W18 |
| seg[2] | PIN\_AA23 | PIN\_U20 | PIN\_AB19 | PIN\_V17 |
| seg[1] | PIN\_Y23 | PIN\_AD6 | PIN\_AD26 | PIN\_V18 |
| seg[0] | PIN\_Y24 | PIN\_AD7 | PIN\_AA18 | PIN\_V19 |

|  |  |  |
| --- | --- | --- |
|  | x | y |
| slide[3] | PIN\_AE19 | PIN\_AB10 |
| slide[2] | PIN\_Y11 | PIN\_W11 |
| slide[1] | PIN\_AC10 | PIN\_AC8 |
| slide[0] | PIN\_V10 | PIN\_AD13 |

|  |  |  |
| --- | --- | --- |
| YorRES | Key[1] | PIN\_P12 |
| interp | Key[2] | PIN\_Y15 |
| addSub | Key[3] | PIN\_Y16 |