1.a)

| A | Aen | B | Ben | C | Cen | Estimado | Obtenido |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | Hi-Z |
| 0 | 1 | 1 | 1 | 0 | 0 | Error | Error |
| 1 | 0 | 1 | 0 | 1 | 0 | Hi-Z | Hi-Z |
| 1 | 1 | 0 | 0 | 0 | 1 | Error | Error |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | Error | Error |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |

1.b)

Gris: Cable no conectado al circuito

Verde claro: 1

Verde oscuro: 0

Azul: Hi-Z

Rojo: Error

1.c) Si hay más de un bit de control prendido, sus datos deben ser iguales (no pueden mezclarse 0s y 1s)

1.d) No tiene sentido que haya más de un enable y dato prendido porque no se sabe cuál es el que está tomando para el resultado final.

2.a)

Entradas:

Force\_input

en\_Force\_input control

w (R0, R1 y R2) control

en\_out (R0, R1 y R2) control

clk control

Salidas:

R0 y debug

R1 y debug

R2 y debug

2.b)

Force\_input <- 1

en\_Force\_input <- 1

w (R1) <- 1

clk <- 1

clk <- 0

w (R1) <- 0

en\_Force\_input <- 0

2.c)

Force\_input <- X

en\_Force\_input <- 1

w (R0) <- 1

clk <- 1

clk <- 0

w (R0) <- 0

en\_Force\_input <- 0 R0 <- X

en\_out (R0) <- 1

w (R1) <- 1

clk <- 1

clk <- 0

w (R1) <- 0

en\_out (R0) <- 0 R1 <- R0

en\_out (R2) <- 1

w (R0) <- 1

clk <- 1

clk <- 0

w (R0) <- 0

en\_out (R2) <- 0 R0 <- R2

en\_out (R1) <- 1

w (R2) <- 1

clk <- 1

clk <- 0

w (R2) <- 0

en\_out (R1) <- 0 R2 <- R1

3.a)

Entradas:

Force\_input

en\_Force\_input control

Reg0\_Write control

Reg0\_enableOut control

Reg1\_Write control

Reg1\_enableOut control

Reg2\_Write control

Reg2\_enableOut control

Reg3\_Write control

Reg3\_enableOut control

ALU\_A\_Write control

ALU\_B\_Write control

OP

Salidas:

N

Z

V

C

3.b) Muestran los valores hexadecimales del contenido de los registros R0 a R3, los operandos A y B, el resultado S y el bus que sale de la ALU y entra a los registros.

3.c)

Force\_input <- 0100

en\_Force\_input <- 1

Reg2\_Write <- 1

clk <- 1

clk <- 0

Reg2\_Write <- 0

en\_Force\_input <- 0 R2 <- 410

Force\_input <- 1101

en\_Force\_input <- 1

Reg3\_Write <- 1

clk <- 1

clk <- 0

Reg3\_Write <- 0

en\_Force\_input <- 0 R3 <- -310

3.d.1)

Force\_input <- 0100

en\_Force\_input <- 1

Reg0\_Write <- 1

clk <- 1

clk <- 0

Reg0\_Write <- 0

en\_Force\_input <- 0 R0 <- 410

Force\_input <- 0000

en\_Force\_input <- 1

Reg1\_Write <- 1

clk <- 1

clk <- 0

Reg1\_Write <- 0

en\_Force\_input <- 0 R1 <- 010

OP <- 11

R0\_enableOut <- 1

ALU\_A\_Write <- 1

clk <- 1

clk <- 0

ALU\_A\_Write <- 0

R0\_enableOut <- 0 ALU\_A <- R0

R1\_enableOut <- 1

ALU\_B\_Write <- 1

clk <- 1

clk <- 0

ALU\_B\_Write <- 0

R1\_enableOut <- 0 ALU\_B <- R1

ALU\_enableOut <- 1

R2\_Write <- 1

clk <- 1

clk <- 0

R2\_Write <- 0

ALU\_enableOut <- 0 R2 <- R0 or R1

OP <- 01

ALU\_enableOut <- 1

R3\_Write <- 1

clk <- 1

clk <- 0

R3\_Write <- 0

ALU\_enableOut <- 0 R3 <- R0 - R1