Raul Aguilar

CS 220

September 20, 2019

Homework 4 Sequential Logic

CHIP Bit {

IN in, load;

OUT out;

PARTS:

// Put your code here:

Mux(a=bit, b=in, sel=load, out=t1);

DFF(in=t1, out=bit, out=out);

}

A screenshot of a computer

Description automatically generated

CHIP Register {

IN in[16], load;

OUT out[16];

PARTS:

// Put your code here:

Bit(in=in[0], load=load, out=out[0]);

Bit(in=in[1], load=load, out=out[1]);

Bit(in=in[2], load=load, out=out[2]);

Bit(in=in[3], load=load, out=out[3]);

Bit(in=in[4], load=load, out=out[4]);

Bit(in=in[5], load=load, out=out[5]);

Bit(in=in[6], load=load, out=out[6]);

Bit(in=in[7], load=load, out=out[7]);

Bit(in=in[8], load=load, out=out[8]);

Bit(in=in[9], load=load, out=out[9]);

Bit(in=in[10], load=load, out=out[10]);

Bit(in=in[11], load=load, out=out[11]);

Bit(in=in[12], load=load, out=out[12]);

Bit(in=in[13], load=load, out=out[13]);

Bit(in=in[14], load=load, out=out[14]);

Bit(in=in[15], load=load, out=out[15]);

}

A screenshot of a computer

Description automatically generated

CHIP RAM8 {

IN in[16], load, address[3];

OUT out[16];

PARTS:

// Put your code here:

DMux8Way(in=load, sel=address, a=reg0, b=reg1, c=reg2, d=reg3, e=reg4, f=reg5, g=reg6, h=reg7);

Register(in=in, load=reg0, out=reg0out );

Register(in=in, load=reg1, out=reg1out );

Register(in=in, load=reg2, out=reg2out );

Register(in=in, load=reg3, out=reg3out );

Register(in=in, load=reg4, out=reg4out );

Register(in=in, load=reg5, out=reg5out );

Register(in=in, load=reg6, out=reg6out );

Register(in=in, load=reg7, out=reg7out );

Mux8Way16(a=reg0out, b=reg1out, c=reg2out, d=reg3out, e=reg4out, f=reg5out, g=reg6out, h=reg7out, sel=address, out=out);

}

A screenshot of a social media post

Description automatically generated

CHIP RAM64 {

IN in[16], load, address[6];

OUT out[16];

PARTS:

// Put your code here:

DMux8Way(in=load, sel=address[3..5], a=ram8a, b=ram8b, c=ram8c, d=ram8d, e=ram8e, f=ram8f, g=ram8g, h=ram8h);

RAM8(in=in, load=ram8a, address=address[0..2], out=ramaout);

RAM8(in=in, load=ram8b, address=address[0..2], out=rambout);

RAM8(in=in, load=ram8c, address=address[0..2], out=ramcout);

RAM8(in=in, load=ram8d, address=address[0..2], out=ramdout);

RAM8(in=in, load=ram8e, address=address[0..2], out=rameout);

RAM8(in=in, load=ram8f, address=address[0..2], out=ramfout);

RAM8(in=in, load=ram8g, address=address[0..2], out=ramgout);

RAM8(in=in, load=ram8h, address=address[0..2], out=ramhout);

Mux8Way16(a=ramaout, b=rambout, c=ramcout, d=ramdout, e=rameout, f=ramfout, g=ramgout, h=ramhout, sel=address[3..5], out=out);

}

A screenshot of a social media post

Description automatically generated

CHIP RAM512 {

IN in[16], load, address[9];

OUT out[16];

PARTS:

// Put your code here:

DMux8Way(in=load, sel=address[6..8], a=ram64a, b=ram64b, c=ram64c, d=ram64d, e=ram64e, f=ram64f, g=ram64g, h=ram64h);

RAM64(in=in, load=ram64a, address=address[0..5], out=ram64aout);

RAM64(in=in, load=ram64b, address=address[0..5], out=ram64bout);

RAM64(in=in, load=ram64c, address=address[0..5], out=ram64cout);

RAM64(in=in, load=ram64d, address=address[0..5], out=ram64dout);

RAM64(in=in, load=ram64e, address=address[0..5], out=ram64eout);

RAM64(in=in, load=ram64f, address=address[0..5], out=ram64fout);

RAM64(in=in, load=ram64g, address=address[0..5], out=ram64gout);

RAM64(in=in, load=ram64h, address=address[0..5], out=ram64hout);

Mux8Way16(a=ram64aout, b=ram64bout, c=ram64cout, d=ram64dout, e=ram64eout, f=ram64fout, g=ram64gout, h=ram64hout, sel=address[6..8], out=out);

}

A screenshot of a computer

Description automatically generated

CHIP RAM4K {

IN in[16], load, address[12];

OUT out[16];

PARTS:

// Put your code here:

DMux8Way(in=load, sel=address[9..11], a=rama, b=ramb, c=ramc, d=ramd, e=rame, f=ramf, g=ramg, h=ramh);

RAM512(in=in, load=rama, address=address[0..8], out=ramaout);

RAM512(in=in, load=ramb, address=address[0..8], out=rambout);

RAM512(in=in, load=ramc, address=address[0..8], out=ramcout);

RAM512(in=in, load=ramd, address=address[0..8], out=ramdout);

RAM512(in=in, load=rame, address=address[0..8], out=rameout);

RAM512(in=in, load=ramf, address=address[0..8], out=ramfout);

RAM512(in=in, load=ramg, address=address[0..8], out=ramgout);

RAM512(in=in, load=ramh, address=address[0..8], out=ramhout);

Mux8Way16(a=ramaout, b=rambout, c=ramcout, d=ramdout, e=rameout, f=ramfout, g=ramgout, h=ramhout, sel=address[9..11], out=out);

}

A screenshot of a social media post

Description automatically generated

CHIP RAM16K {

IN in[16], load, address[14];

OUT out[16];

PARTS:

// Put your code here:

DMux4Way(in=load, sel=address[12..13], a=rama, b=ramb, c=ramc, d=ramd);

RAM4K(in=in, load=rama, address=address[0..11], out=ramaout);

RAM4K(in=in, load=ramb, address=address[0..11], out=rambout);

RAM4K(in=in, load=ramc, address=address[0..11], out=ramcout);

RAM4K(in=in, load=ramd, address=address[0..11], out=ramdout);

Mux4Way16(a=ramaout, b=rambout, c=ramcout, d=ramdout, sel=address[12..13], out=out);

}

A screenshot of a social media post

Description automatically generated

CHIP PC {

IN in[16],load,inc,reset;

OUT out[16];

PARTS:

// Put your code here:

Inc16(in=count, out=countinc);

Mux16(a=count, b=countinc, sel=inc, out=incout);

Mux16(a=incout, b=in, sel=load, out=loadout);

Mux16(a=loadout, b=false, sel=reset, out=resetout);

Register(in=resetout, load=true, out=count, out=out);

}

A screenshot of a social media post

Description automatically generated