David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

INPUT				OUTPUT						
N3	N2	N1	N0	A	B	C	D	E	F	G
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	0	1	1	0
0	1	0	0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0	1	0	0
0	1	1	0	0	1	0	0	0	0	0
0	1	1	1	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	1	0	0
1	0	1	0	0	0	0	1	0	0	0
1	0	1	1	1	1	0	0	0	0	0
1	1	0	0	0	1	1	0	0	0	1
1	1	0	1	1	0	0	0	0	1	0
1	1	1	0	0	1	1	0	0	0	0
1	1	1	1	0	1	1	1	0	0	0

0001 0000 1001 0011

Segment A:

Karnaugh Map

		N3 N2			
		00	01	11	10
N1 N0	00	0	1	0	0
	01	1	0	1	0
	11	0	0	0	1
	10	0	0	0	0

A(N3N2N1N0) = N3’N2’N1’N0 + N3’N2N1’N0’ + N3N2N1’N0 + N3N2’N1N0

David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

Segment B:

Karnaugh Map

		N3 N2			
		00	01	11	10
N1 N0	00	0	0	1	0
	01	0	1	0	0
	11	0	0	1	1
	10	0	1	1	0

B(N3N2N1N0) = N3N1N0 + N2N1N0’ + N3N2N0’ + N3’N2N1’N0

Segment C:

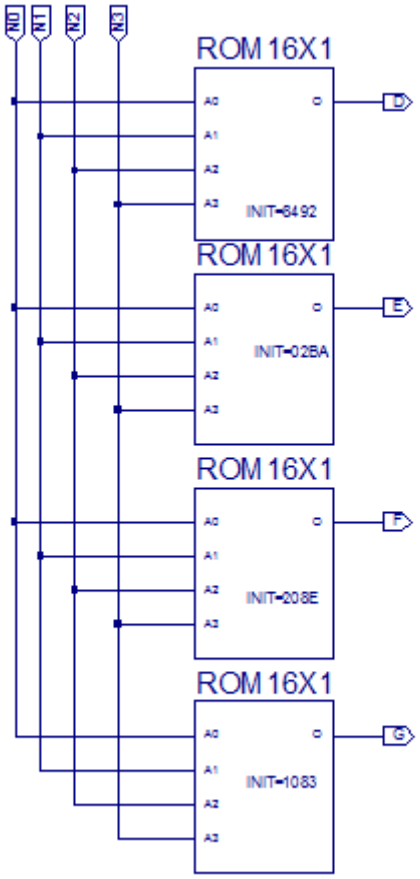
Karnaugh Map

		N3 N2			
		00	01	11	10
N1 N0	00	0	0	1	0
	01	0	0	0	0
	11	0	0	1	0
	10	1	0	1	0

C(N3N2N1N0) = N3N2N1 + N3N2N0’ + N3’N2’N1N0’

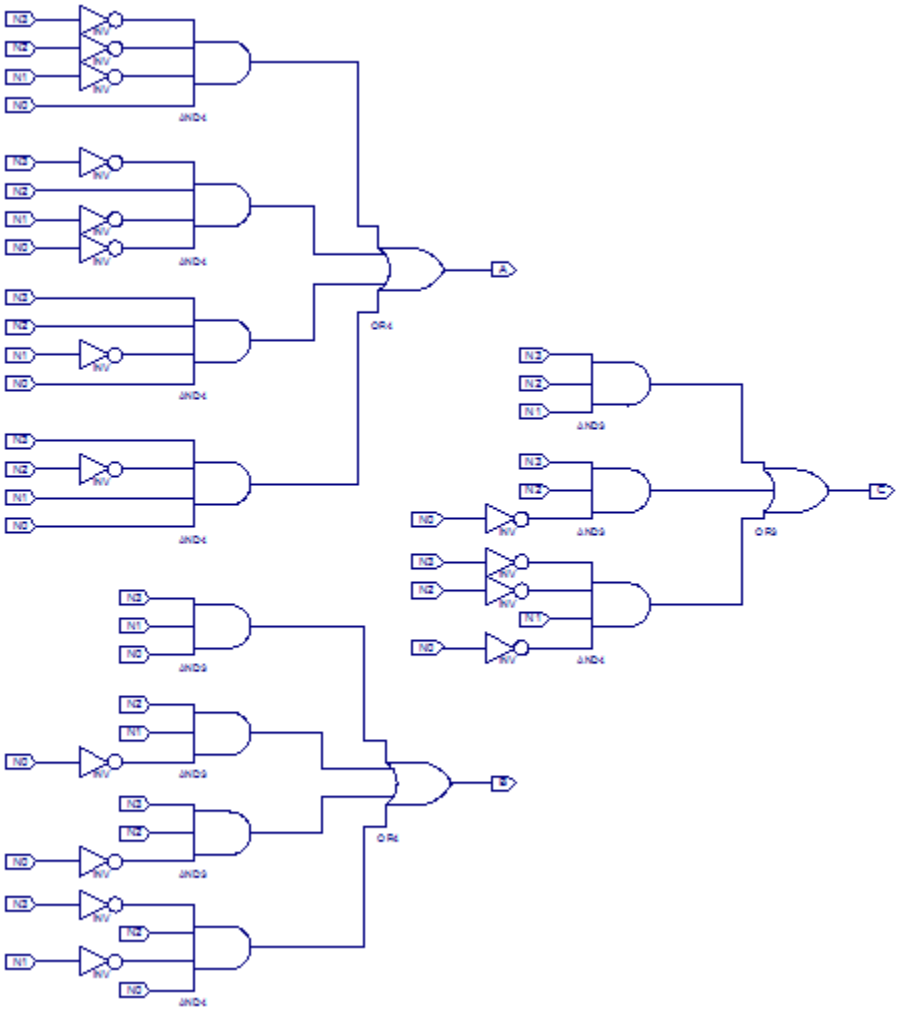
David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

ROM Schematic:



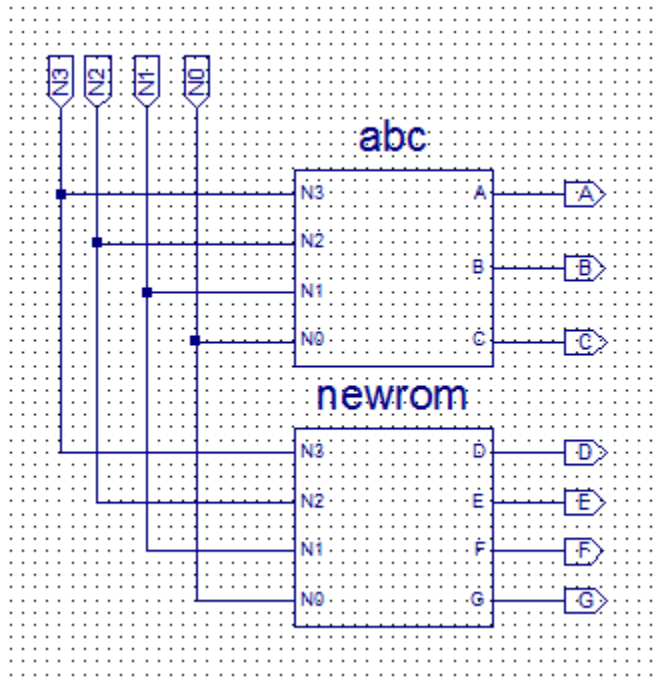
David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

Combinational Logic Schematic:



David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

Seven Segment Decoder Schematic:

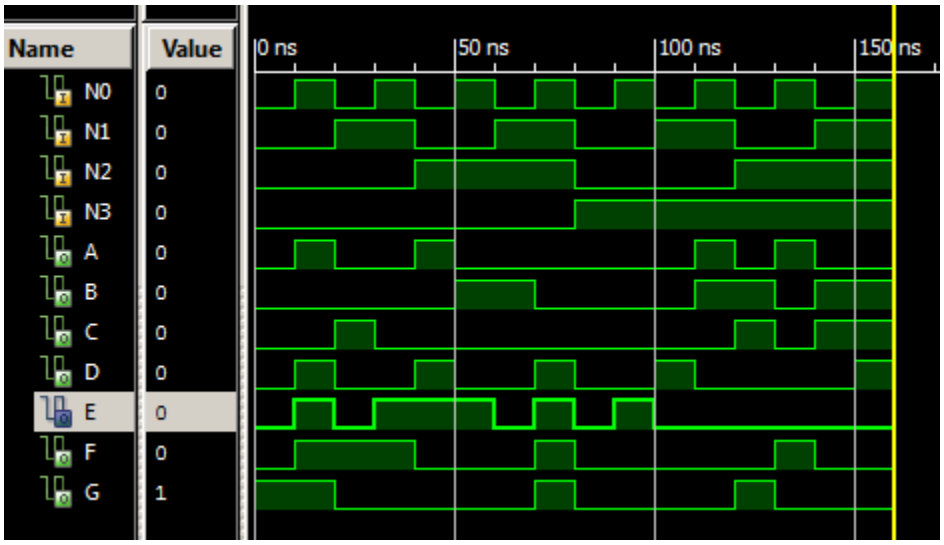


TCL:

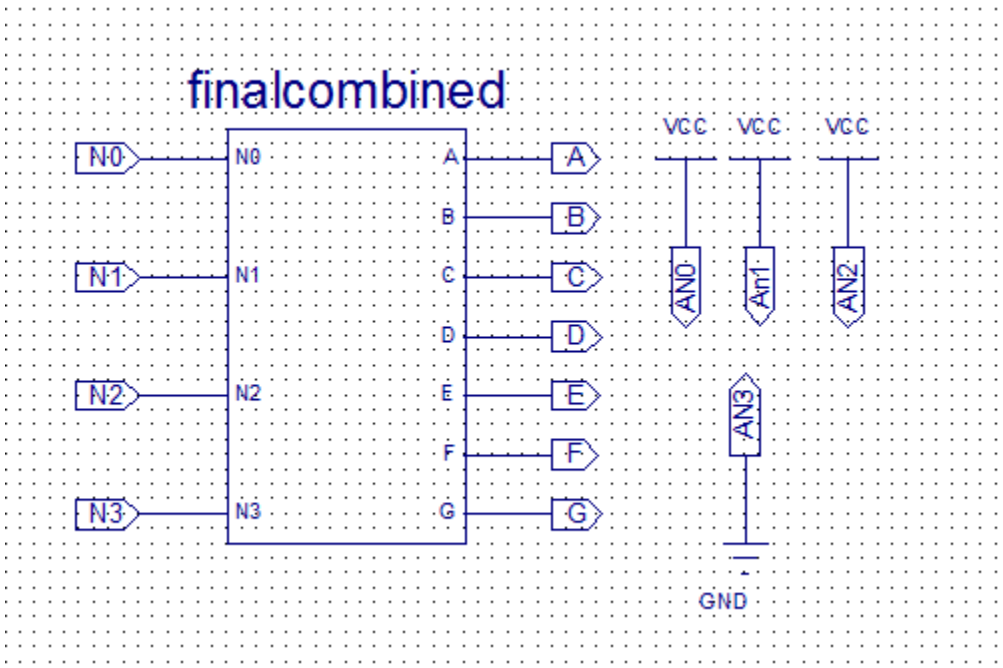
```
#add all signals to the waveform viewer
wave add / -radix hex



#define how the data input signals will behave when you run the simulation
#the command "isim force add (signal) 0 -time 0 -value 1 -time 20ns -repeat 40ns"
#will have a value of 0 from time 0, then change to 1 at time 20ns, and then repe
isim force add N0 0 -time 0 -value 1 -time 10ns -repeat 20ns
isim force add N1 0 -time 0 -value 1 -time 20ns -repeat 40ns
isim force add N2 0 -time 0 -value 1 -time 40ns -repeat 80ns
isim force add N3 0 -time 0 -value 1 -time 80ns -repeat 160ns



#Nothing will change in the waveform viewer until you run the simulation for some
run 160ns
```

David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

Simulation Waveform:



Test Bench Schematic:



David DeGraw
Lab #4 Seven Segment Decoder
2/3/2015

Test Bench UCF:

```
NET "N0" LOC = "L14"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW4
NET "N1" LOC = "L13"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW5
NET "N2" LOC = "N17"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW6
NET "N3" LOC = "R17"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW7

## Buttons
#NET "btn<0>" LOC = "B18"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name =
#NET "btn<1>" LOC = "D18"; # Bank = 1, Pin name = IP/VREF_1, Type = VREF, Sch n
#NET "btn<2>" LOC = "E18"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name =
#NET "btn<3>" LOC = "H13"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name =

## 7 segment display
NET "A" LOC = "L18"; # Bank = 1, Pin name = IO_L10P_1, Type = I/O, Sch name = C
NET "B" LOC = "F18"; # Bank = 1, Pin name = IO_L19P_1, Type = I/O, Sch name = C
NET "C" LOC = "D17"; # Bank = 1, Pin name = IO_L23P_1/HDC, Type = DUAL, Sch nam
NET "D" LOC = "D16"; # Bank = 1, Pin name = IO_L23N_1/LDC0, Type = DUAL, Sch na
NET "E" LOC = "G14"; # Bank = 1, Pin name = IO_L20P_1, Type = I/O, Sch name = C
NET "F" LOC = "J17"; # Bank = 1, Pin name = IO_L13P_1/A6/RHCLK4/IRDY1, Type = R
NET "G" LOC = "H14"; # Bank = 1, Pin name = IO_L17P_1, Type = I/O, Sch name = C
#NET "dp"     LOC = "C17"; # Bank = 1, Pin name = IO_L24N_1/LDC2, Type = DUAL,

NET "AN0" LOC = "F17"; # Bank = 1, Pin name = IO_L19N_1, Type = I/O, Sch name =
NET "AN1" LOC = "H17"; # Bank = 1, Pin name = IO_L16N_1/A0, Type = DUAL, Sch na
NET "AN2" LOC = "C18"; # Bank = 1, Pin name = IO_L24P_1/LDC1, Type = DUAL, Sch
NET "AN3" LOC = "F15"; # Bank = 1, Pin name = IO_L21P_1, Type = I/O, Sch name =
```

Anomalies:

Even though this week's lab was probably the most intense so far, it was pretty straight forward. I had a bug in my rom schematic, and regenerating all of the symbols took forever and was basically the worst thing ever.