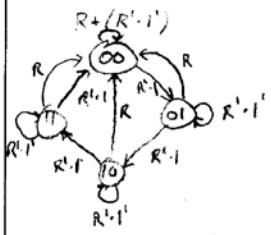
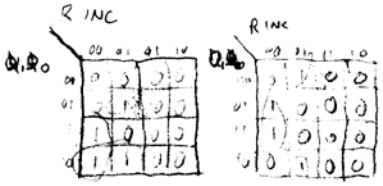
Ty Madsen
ECEN 220
8 November 2013

Lab 8 - 4 x 7 Segment Controller

LAB #8 4x7 Seg. Decoder — Ty Madsen

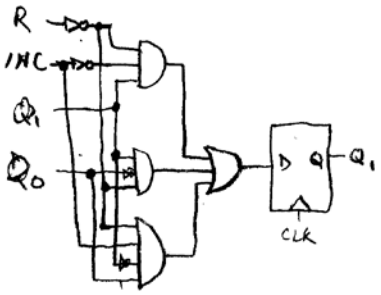


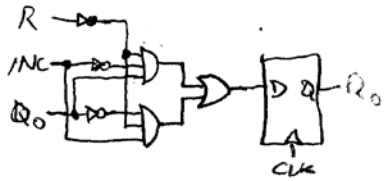
R	Inc	Q_1	Q_0	Q_1	Q_0
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	1	1
0	1	1	1	0	0
1	—	—	—	0	0



$$Q_1 = R'INC'Q_1 + R'Q_1Q_0' + R'INCQ_1'Q_0$$

$$Q_0 = R'INC'Q_0 + R'INCQ_0'$$





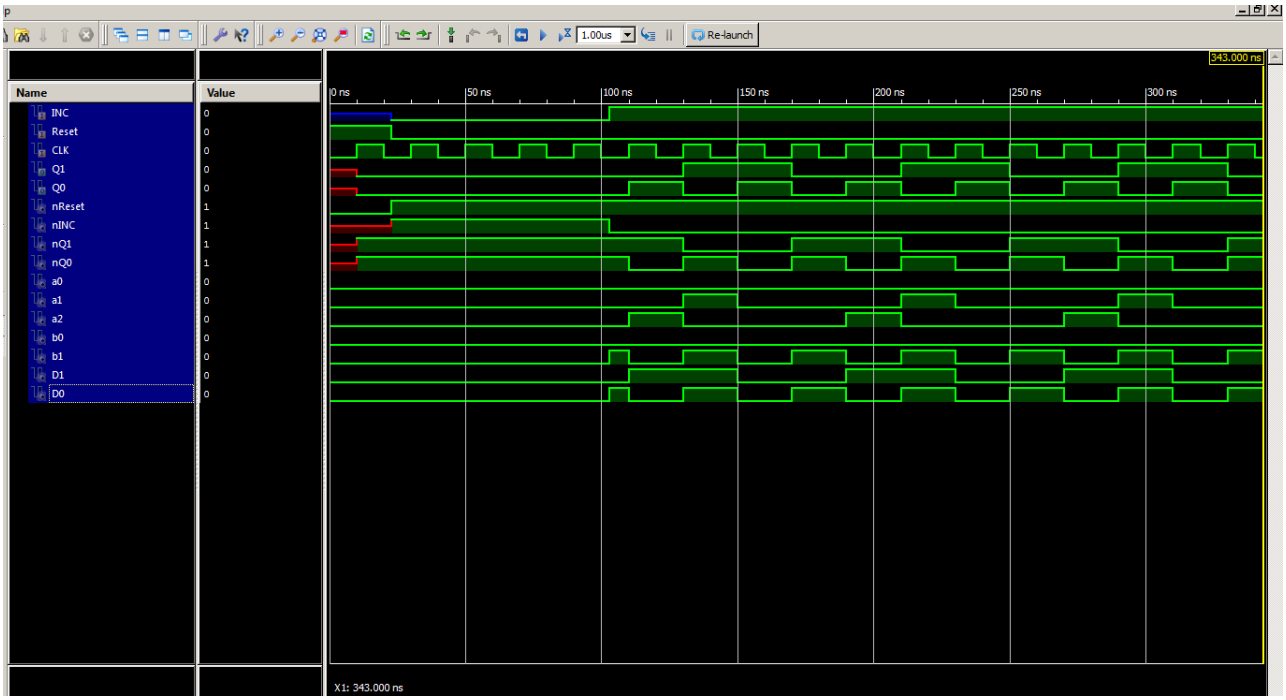
50 MHz Sys Clk in
200 Hz ← Clk

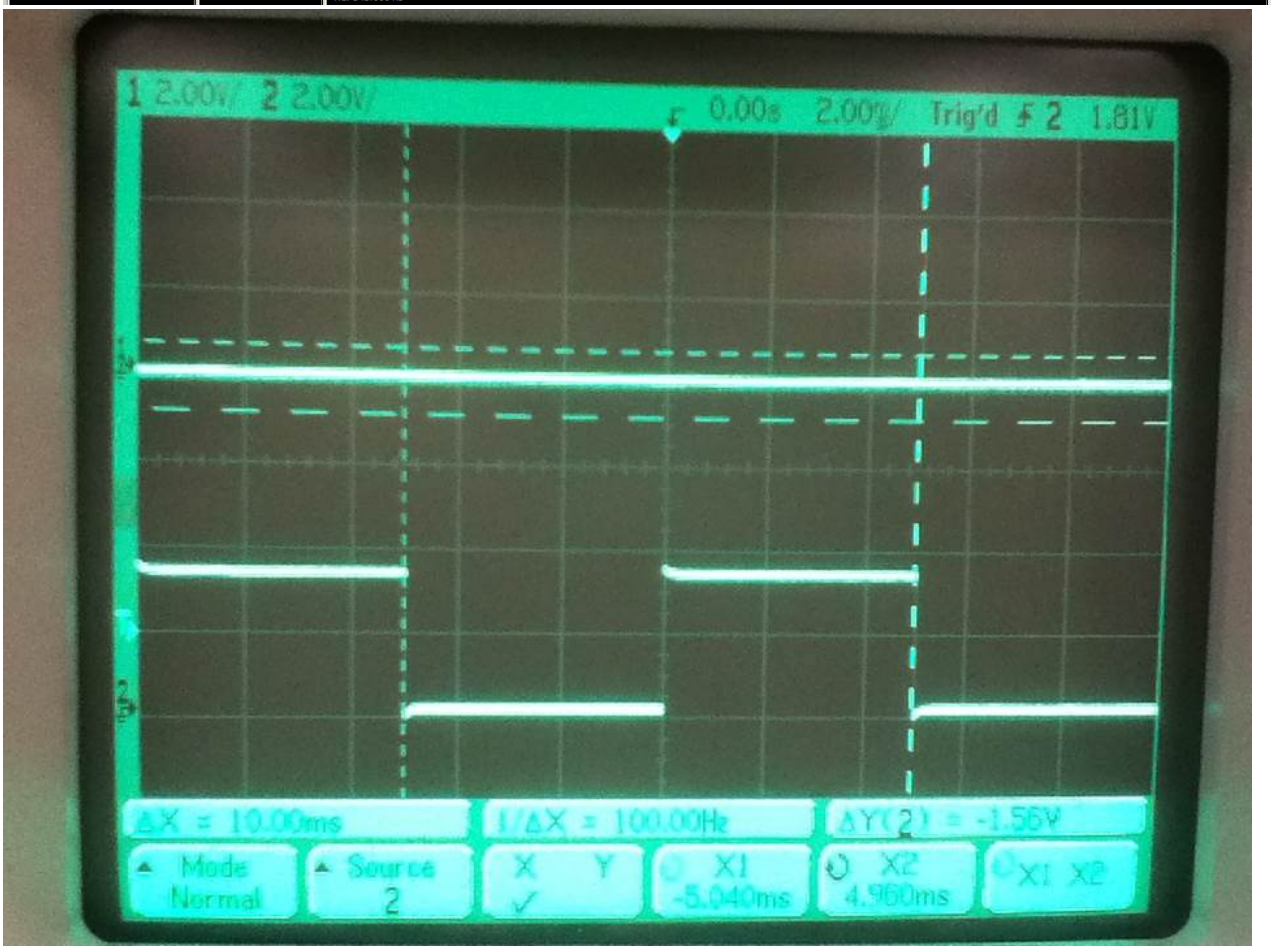
50,000,000 / sec
÷ 250,000 Counter
= 200 Hz

Dp - Sw (0-3)

bp - 100
zero - 101

NINJA ACEROLA





The frequency for the tp 100Hz, so for zero it will be 200 Hz

MOD4 verilog:

```
module Mod4(
  input INC,
  input Reset,
  input CLK,
  output Q1,
  output Q0
  );

        wire nReset, nINC, nQ1, nQ0, a0, a1, a2, b0, b1, D1, D0;

        not(nReset, Reset);
        not(nINC, INC);
        not(nQ1, Q1);
        not(nQ0, Q0);

        and(a0, nReset, nINC, Q1);
        and(a1, nReset, Q1, nQ0);
        and(a2, nReset, INC, nQ1, Q0);
        or(D1, a0, a1, a2);

        and(b0, nReset, nINC, Q0);
        and(b1, nReset, INC, nQ0);
        or(D0, b0, b1);

        FF_DC ff1(Q1, CLK, Reset, D1);
        FF_DC ff0(Q0, CLK, Reset, D0);

endmodule

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

isim force add CLK 0 -time 0 -value 1 -time 10ns -repeat 20ns

isim force add Reset 1 -time 0

run 23 ns
```

```
isim force add INC 0 -time 0 -value 1 -time 80ns -repeat 320ns

isim force add Reset 0 -time 0

#Nothing will change in the waveform viewer until you run the simulation for some period of time.
run 320ns
```

Timer:

```
module myTimer(clk, reset, zero, tp);
            input clk, reset;
  output zero, tp;

  wire [23:0] counter;
  prog_timer timer(clk, reset, 1'b1, 24'd250000, counter, zero, tp);

endmodule
```

```
NET     clk         LOC = "B8"; # GCLK
NET     reset           LOC = "H13"; #BTN3
NET     zero            LOC = "B4"; # IO0
NET     tp              LOC = "A4"; # IO1
```

Seven segment controller:

```
module segController47(
   input [3:0] Digit1,
   input [3:0] Digit2,
   input [3:0] Digit3,
   input [3:0] Digit4,
   input clk,
   input reset,
   input Dp0,
   input Dp1,
   input Dp2,
   input Dp3,
   output Ca,
   output Cb,
   output Cc,
   output Cd,
   output Ce,
   output Cf,
   output Cg,
   output AN0,
   output AN1,
   output AN2,
   output AN3,
   output DP,
   output [1:0] Mod,
   output tp,
   output zero
   );

         wire [23:0] counter;
         prog_timer timer(clk, reset, 1'b1, 24'd250000, counter, zero, tp);

         Mod4 mod(zero, reset, clk, Mod[1], Mod[0]);

         wire [3:0] number;
         mux16_4 mux16(number, Mod, Digit1, Digit2, Digit3, Digit4);

         SEVEN_SEG_DECODER decoder(.N3(number[3]), .N2(number[2]), .N1(number[1]),
.N0(number[0]), .A(Ca), .B(Cb), .C(Cc), .D(Cd), .E(Ce), .F(Cf), .G(Cg));

         Decoder2_4 decoder24(Mod[0], Mod[1], AN0, AN1, AN2, AN3);
```

```
        MUX41 dpmux(DP, Mod, {Dp3,Dp2,Dp1,Dp0});


endmodule
```

Test bench:

```
module testBench7segctrl(
   input clk,
   input reset,
   input Dp0,
   input Dp1,
   input Dp2,
   input Dp3,
   output Ca,
   output Cb,
   output Cc,
   output Cd,
   output Ce,
   output Cf,
   output Cg,
   output AN0,
   output AN1,
   output AN2,
   output AN3,
   output DP,
   output [1:0] Mod,
   output tp,
   output zero
   );

        segController47 controller(4'b0001, 4'b1010, 4'b1011, 4'b1000, clk, reset, ~Dp0, ~Dp1, ~Dp2,
~Dp3, Ca, Cb, Cc, Cd, Ce, Cf, Cg, AN0, AN1, AN2, AN3, DP, Mod, tp, zero);


endmodule

# This file is a general .ucf for Nexys2 rev A board
# To use it in a project:
# - remove or comment the lines corresponding to unused pins
```

```
# - rename the used signals according to the project

NET    clk         LOC = "B8"; # GCLK
NET    reset         LOC = "H13"; #BTN3
NET    zero          LOC = "B4"; # IO0
NET    tp           LOC = "A4"; # IO1

## clock pin for Nexys 2 Board
#NET "clk"   LOC = "B8"; # Bank = 0, Pin name = IP_L13P_0/GCLK8, Type = GCLK, Sch name = GCLK0
##NET "clk1" LOC = "U9"; # Bank = 2, Pin name = IO_L13P_2/D4/GCLK14, Type = DUAL/GCLK, Sch name
= GCLK1
#
## Leds
#NET "Led<0>"  LOC = "J14"; # Bank = 1, Pin name = IO_L14N_1/A3/RHCLK7, Type = RHCLK/DUAL, Sch
name = JD10/LD0
#NET "Led<1>"  LOC = "J15"; # Bank = 1, Pin name = IO_L14P_1/A4/RHCLK6, Type = RHCLK/DUAL, Sch
name = JD9/LD1
#NET "Led<2>"  LOC = "K15"; # Bank = 1, Pin name = IO_L12P_1/A8/RHCLK2, Type = RHCLK/DUAL, Sch
name = JD8/LD2
#NET "Led<3>"  LOC = "K14"; # Bank = 1, Pin name = IO_L12N_1/A7/RHCLK3/TRDY1, Type =
RHCLK/DUAL, Sch name = JD7/LD3
#NET "Led<4>"  LOC = "E17"; # Bank = 1, Pin name = IO, Type = I/O, Sch name = LD4
#NET "Led<5>"  LOC = "P15"; # Bank = 1, Pin name = IO, Type = I/O, Sch name = LD5
#NET "Led<6>"  LOC = "F4";  # Bank = 3, Pin name = IO, Type = I/O, Sch name = LD6
#NET "Led<7>"  LOC = "R4";  # Bank = 3, Pin name = IO/VREF_3, Type = VREF, Sch name = LD7

## Switches
NET Dp0 LOC = "G18"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW0
NET Dp1 LOC = "H18"; # Bank = 1, Pin name = IP/VREF_1, Type = VREF, Sch name = SW1
NET Dp2 LOC = "K18"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW2
NET Dp3 LOC = "K17"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW3
#NET "sw<4>" LOC = "L14"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW4
#NET "sw<5>" LOC = "L13"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW5
#NET "sw<6>" LOC = "N17"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW6
#NET "sw<7>" LOC = "R17"; # Bank = 1, Pin name = IP, Type = INPUT, Sch name = SW7


## 7 segment display
NET Ca LOC = "L18"; # Bank = 1, Pin name = IO_L10P_1, Type = I/O, Sch name = CA
NET Cb LOC = "F18"; # Bank = 1, Pin name = IO_L19P_1, Type = I/O, Sch name = CB
NET Cc LOC = "D17"; # Bank = 1, Pin name = IO_L23P_1/HDC, Type = DUAL, Sch name = CC
NET Cd LOC = "D16"; # Bank = 1, Pin name = IO_L23N_1/LDC0, Type = DUAL, Sch name = CD
```

NET Ce LOC = "G14"; # Bank = 1, Pin name = IO_L20P_1, Type = I/O, Sch name = CE
NET Cf LOC = "J17"; # Bank = 1, Pin name = IO_L13P_1/A6/RHCLK4/IRDY1, Type = RHCLK/DUAL, Sch name = CF
NET Cg LOC = "H14"; # Bank = 1, Pin name = IO_L17P_1, Type = I/O, Sch name = CG
NET DP LOC = "C17"; # Bank = 1, Pin name = IO_L24N_1/LDC2, Type = DUAL, Sch name = DP

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

Anomalies:
NONE ☺