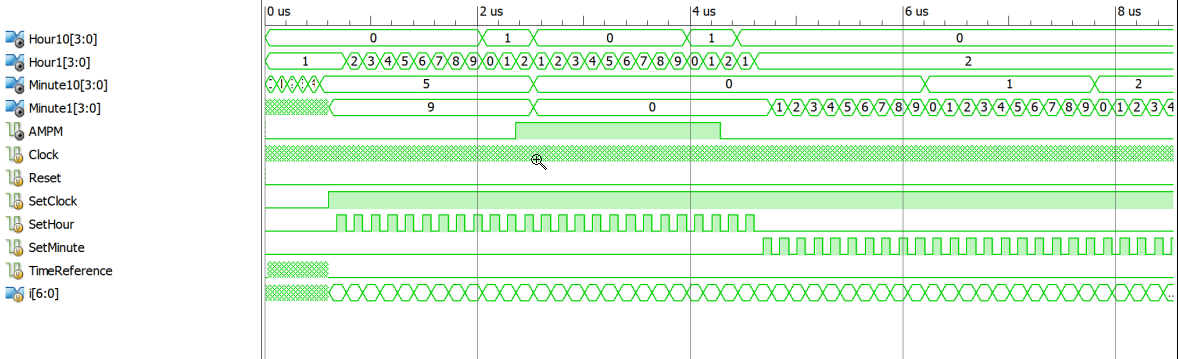
Brendan Mulholland CM 1832

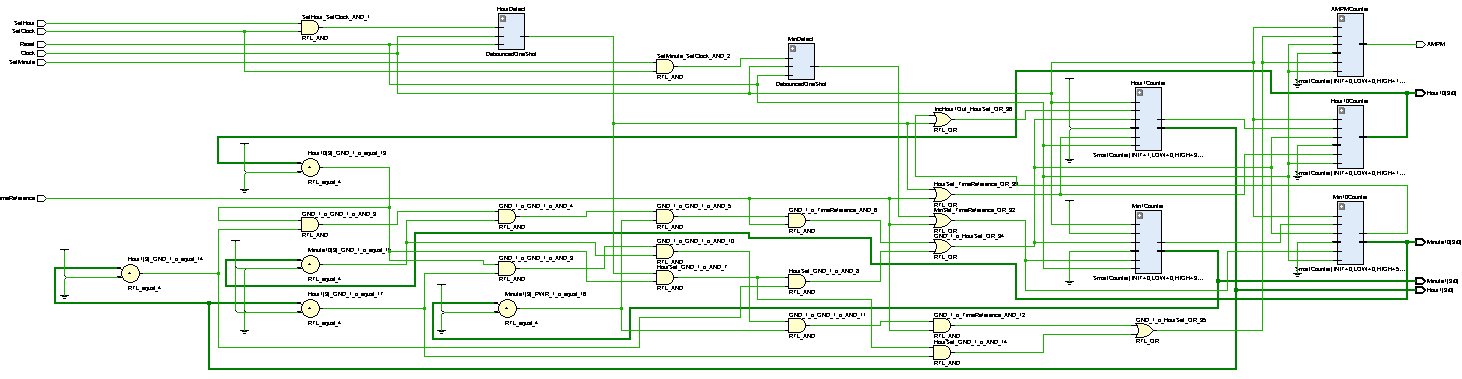
ECE 433

10/7/2019

Lab 5 Memo: Clock unit

This is my implementation of the clock unit. I utilized my Smart Counter wrappers and a combine Debouncer and One Shot modules. I made a test bench for the unit, that includes all relevant state transitions and inputs.







Clock Module

`timescale 1ns / 1ps

// File name    : ClockTB.v

// Written by   : Brendan Mulholland

// ECE433-01 Fall 2019

// Date: October 6,2019

// Purpose: The Test Bench for the Clock Module

module ClockTB;

    // Inputs

    reg Clock;

    reg Reset;

    reg SetClock;

    reg SetHour;

    reg SetMinute;

    reg TimeReference;

    // Outputs

    wire [3:0] Hour10;

    wire [3:0] Hour1;

    wire [3:0] Minute10;

    wire [3:0] Minute1;

    wire AMPM;

    reg [6:0] i;

    // Instantiate the Unit Under Test (UUT)

    ClockModule2019 uut (

        .Clock(Clock),

        .Reset(Reset),

        .SetClock(SetClock),

        .SetHour(SetHour),

        .SetMinute(SetMinute),

        .TimeReference(TimeReference),

        .Hour10(Hour10),

        .Hour1(Hour1),

        .Minute10(Minute10),

        .Minute1(Minute1),

        .AMPM(AMPM)

    );

    initial begin

        // Initialize Inputs

        Clock = 0;

        Reset = 0;

        SetClock = 0;

        SetHour = 0;

        SetMinute = 0;

        TimeReference = 0;

    end

    always #5 Clock = ~Clock;

    initial begin

    #10 Reset = 1;

    #10 Reset = 0;

    //Increment through 10 minutes

    #2 TimeReference = ~TimeReference;

    for(i=0; i < 58; i= i+1) begin

        #5 TimeReference = ~TimeReference;

        #5 TimeReference = ~TimeReference;

    end

    #5 TimeReference = 0;

    SetClock = 1;

    for(i=0; i < 25; i= i+1) begin

        #80 SetHour = 1;

        #80 SetHour = 0;

    end

    for(i=0; i < 25; i= i+1) begin

        #80 SetMinute = 1;

        #80 SetMinute = 0;

    end

    $stop;

    end

endmodule

ClockTB.v