Name: Sean Gordon **Lab Section: F**

CprE 288 Fall 2018 – Homework 6

Due Sunday. October 21 (on Canvas 11:59pm)

Notes:

- Homework must be typed and submitted as a PDF or Word Document (i.e. .doc or .docx) only.
- If collaborating with others, you must document who you collaborate with, and specify in what way you collaborated (see last page of homework assignment), review the homework policy section of the syllabus: http://class.ece.iastate.edu/cpre288/syllabus.asp for further details.
- Review University policy relating to the integrity of scholarship. See ("Academic Dishonesty"): http://catalog.iastate.edu/academic_conduct/#academicdishonestytext
- Late homework is accepted within two days from the due date. *Late penalty is 10% per day.* ***Except on Exam weeks****, homework only accepted 1 day late.*
- ***Note:*** *Code that will not compile is a typo. Answering a question as "will not compile"* ***will be marked incorrect****. Contact the Professor if you think you have found a typo.*
- **Note: You are not allowed to use any MACROs in your code, except for register names.**
  - Example: You will lose points for: GPIO_PORTA_DEN_R = GPIO_PORTA_DEN_R | PIN1
  - Must use: GPIO_PORTA_DEN_R = GPIO_PORTA_DEN_R | 0b0000_0010; // or 0x02

Note: Unless otherwise specified, all problems assume the TM4C123 is being used

Question 1: ADC Successive Approximation (5 pts)

A 4-bit ADC (of 16 steps in the analog range) uses the Successive Approximation implementation. The input voltage range of the ADC is 0V-32V. If the input is 20V, how does the ADC work out each bit of the digital encoding? Fill the following table to show the steps (as done in class). The first step is given.

Step	Range	DN_Mid	AV_Mid(V)	Input >= AV_Mid
0	xxxx	1000	16	1
1	1xxx	1100	24	0
2	10xx	1010	20	1
3	101x	1011	22	0
4	1010			

The digital value is _______1010______ (binary) and _______10______ (decimal).

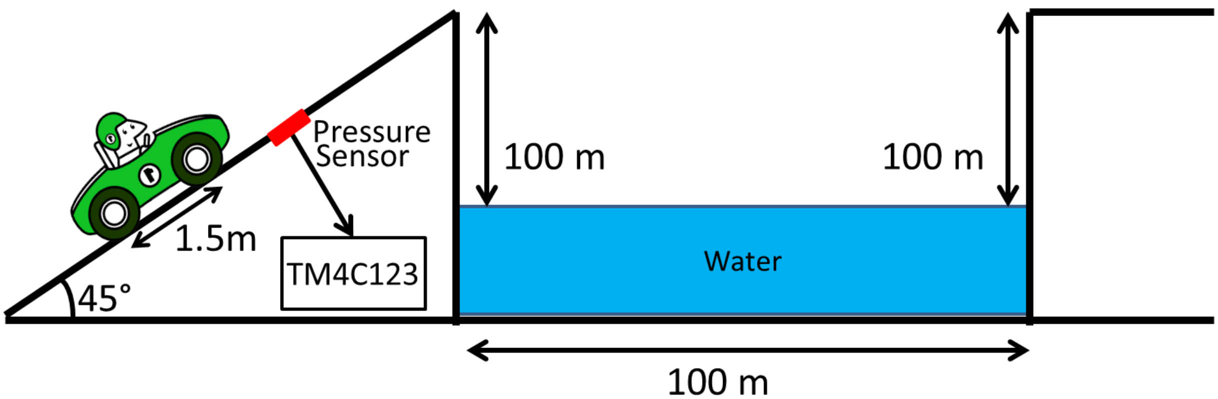
Name: Sean Gordon **Lab Section: F**

Question 2: Shutdown system (15 pts)

A pressure sensor is embedded in the ramp below. When pressure is applied to the sensor, a logic 1 is driven on the Timer 1 Input Capture input wire of the TM4C123, else a logic 0 is driven. You are to write a C-program that will apply the car's breaks to stop it if its velocity is not large enough to jump the gap shown below. This program is required to use the Timer 1 Input Capture interrupt.

Note: You are not allowed to use any MACROs in your code, except for register names.
- Will lose points for: GPIO_PORTA_DEN_R = GPIO_PORTA_DEN_R | PIN1
- Must use: GPIO_PORTA_DEN_R = GPIO_PORTA_DEN_R | 0b0000_0010; // or 0x02

Assumptions:
1) The car is moving at a constant velocity while on the ramp
2) The car is treated as a "point mass" for computing the physics of the problem



a) Complete `Config_Timer1` to program the Timer 1 configuration registers as follows: (5 pts)

- Input Capture interrupt enabled
- 24-bit timer
- Detect positive edge events

```
Config_Timer1()
{
     SYSCTL_RCGCGPIO_R |= 0x2;        //Enable clock for GPIO port B

     GPIO_PORTB_AFSEL_R |= 0b0001_0000;    //Set pin 4 to AF
     GPIO_PORTB_PCTL_R |= 0x0007_0000;     //Set pin 4 to timer

     GPIO_PORTB_DEN_R |= 0b0001_0000;      //Pin 4 to digital
     GPIO_PORTB_DIR_R &= 0b1110_1111;           //Pin 4 to input


     SYSCTL_RCGCTIMER_R |= 0b0000_0010;    //Enable Timer1 clock
```

```
    TIMER1_CTL_R &= 0b1000_0000;          //Clear all Timer1A bits

    TIMER1_CFG_R = 0x4;                   //Set to 16-bit

    //Count up, Edge-Time Mode, Capture Mode
    TIMER1_TAMR_R = 0b0001_0111;


    TIMER1_TAPR_R = FF;                   //Set prescaler

    TIMER1_TAILR_R = 0xFFFF               //Set Timer1 cap


    TIMER1_ICR_R |= 0b0000_0100;          //Clear capture interrupts
    TIMER1_IMR_R |= 0b0000_0100;          //Enable capture
    interrupts

    TIMER1_CTL_R |= 0b0000_0001;          //Enable Timer1
}
```

b) Complete the ISR (i.e. handler) called `Timer_1_Handler` to store `first_wheel_hit` and `second_wheel_hit` (5 pts)

c) Complete `Stop_car()`. It should return 1 if the car is not moving fast enough to jump the gap (5 pts)

Name: Sean Gordon **Lab Section: F**

```
// Global variables (you may use additional variables)
volatile unsigned int first_wheel_hit;  // When 1st wheel hits sensor
volatile unsigned int second_wheel_hit; // When 2nd wheel hits sensor
volatile int done_flag=0; // 1 after both first and second_wheel_hit
                          // have been stored

static int state = 0;

// Store first_wheel_hit and second_wheel_hit
void Timer_1_Handler(void)
{
     if(TIMER1_RIS_R & 0b0000_0100){    //If correct interrupt
          TIMER1_ICR = 0b0000_0100;     //Clear interrupt

          if(state == 0){               //Capture 1st time
               first_wheel_hit = TIMER1_TAR_R;
               state = 1;
          }
          else{                         //Capture 2nd time
               second_wheel_hit = TIMER1_TAR_R;
               done_flag = 1;           //Set done flag
               state = 2;
          }
     }
}

// Return 1 if the car is not fast enough to jump the gap
int Stop_car(void)
{

     double velocity =   //.0625 = time between clock ticks
               (second_wheel_hit - first_wheel_hit) * .0625;
     velocity = 1.5 / (velocity / 1000000);

     if(velocity  < 31.2)     //Too slow
          return 1;
     return 0;                     //Fast enough
}
```

Name: Sean Gordon **Lab Section: F**

```
// Program to stop Car if it is not fast enough to jump
// the gap.
main()
{

  int stop = 0;

  Config_Timer1();


  while(1)
  {
    // Wait for events to be captured
    while(!done_flag)
    {
    }

    done_flag = 0;  // Clear flag

    // Check if car needs to stop
    stop = Stop_car();

    if(stop)
    {
      lprintf("Stopping Car!";
    }
    else
    {
      lprintf("Car going to Jump!!");
    }

  } // end while
}
```

Name: Sean Gordon **Lab Section: F**

Question 3: Software implemented Input Capture (10 pts)

a) Assume the TM4C123 does not have Input Capture hardware or Interrupts. Write a C program to save TIMER1's count value (i.e. TCNT1) when a positive edge event occurs on PortD, pin4. Note: you may want to do 3c first. (4 pts)

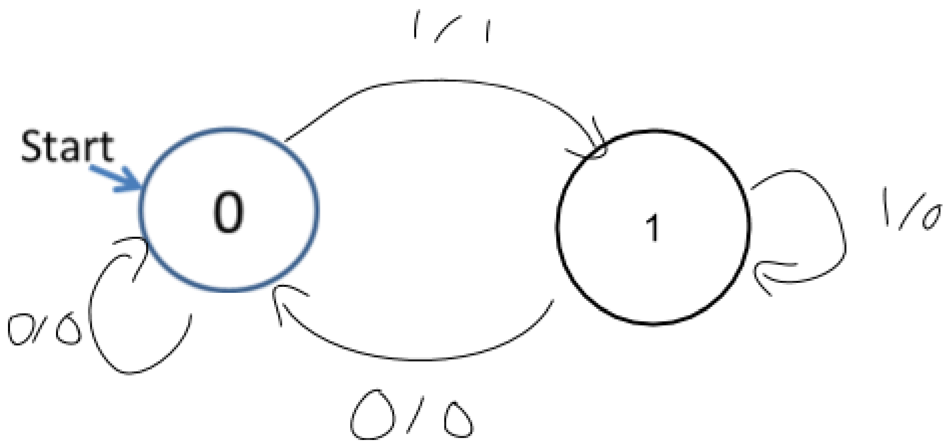
```
int main(void)
{
	unsigned int rising_edge_time;

	while((TIMER1_MIS_R & TIMER_MIS_CAEMIS) == 0){
	}
	rising_edge_time = TIMER1_TAR_R;   //Save value of timer
	TIMER1_ICR_R |= TIMER_ICR_CAECINT; //Clear interrupt

	return rising_edge_time;
}
```

b) Describe two disadvantages of your software-implemented input capture program, as compared to using Input Capture hardware with Interrupts. (3 pts)

- Polling stops the cpu from doing anything else during that time
- Chances of data loss are much greater with polling

c) Complete the bubble diagram below to implement a Mealy State Machine that implements a positive edge detector (i.e. the state machine outputs 1 each time a positive edge occurs on the input). Assume at Start the input to the state machine is initially 0 (3 pts)

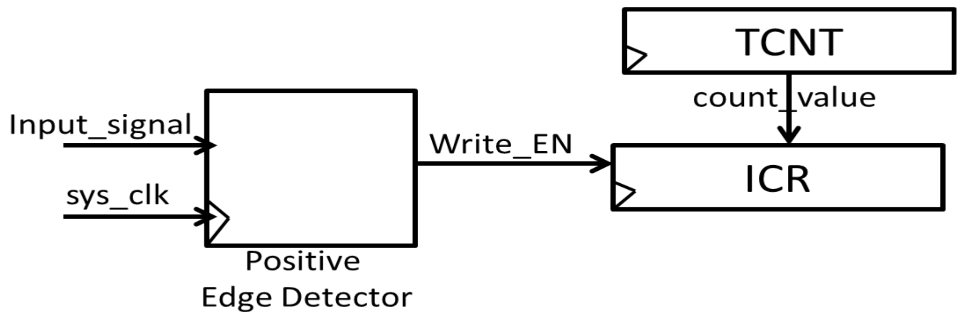
Name: Sean Gordon **Lab Section: F**



Name: Sean Gordon **Lab Section: F**

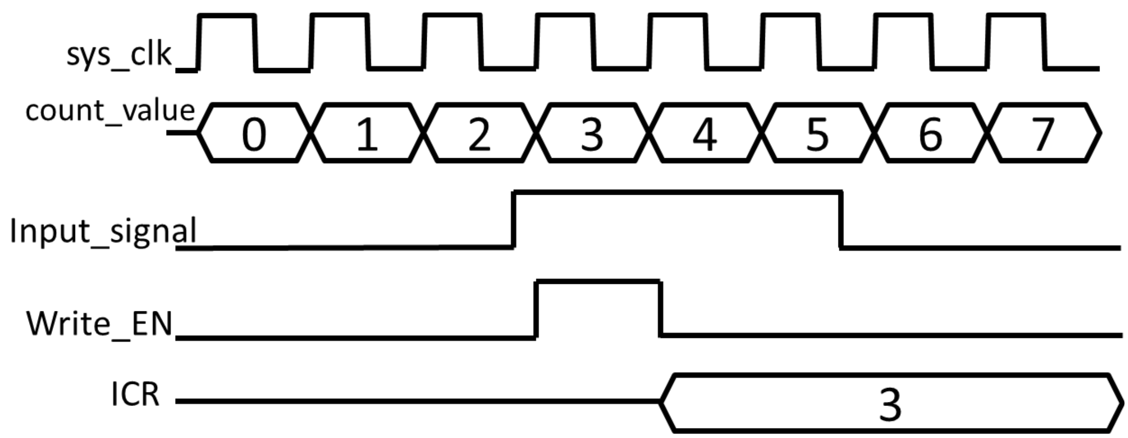
Question 4: Edge Detection (10 pts)

The following simple block diagram illustrates how one may implement an Input Capture circuit.



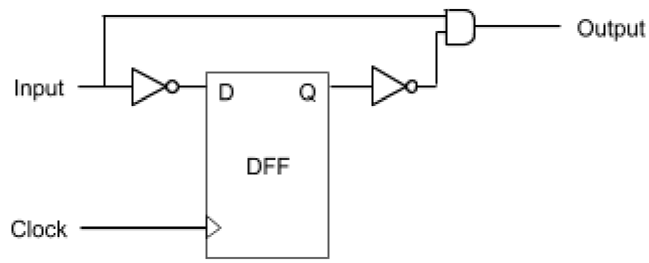
An example timing diagram for capturing the rising edge of `Input_signal` is given below. In summary, on the occurrence of a positive edge on `Input_signal`: 1) `Write_EN` pulses '1' for one `sys_clk` cycle, and 2) the value in the Timer/Counter Register (TCNT) is loaded into the Input Capture Register (ICR).

Assume: Any time `Input_signal` transitions is will hold its value for at least 1 `sys_clk` cycle.

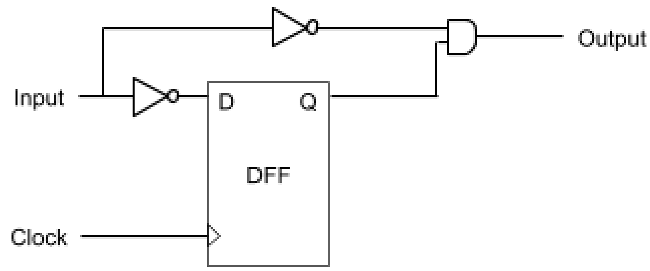


Name: Sean Gordon **Lab Section: F**

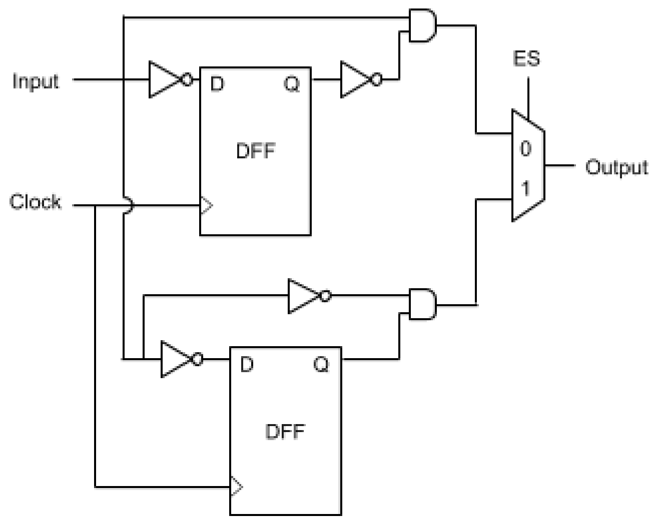
a) Draw out a digital circuit to implement an Edge Detector that creates a 1-sys_clk-wide pulse on Write_EN to load the TCNT register into the ICR register when a rising edge occurs on Input_signal. The only components you can use are D-Flip Flops, and AND, OR, NOT gates (5 pts)



b) Repeat a) for an edge detector that creates a Write_EN pulse for detecting the falling edge of Input_signal. (3 pts)



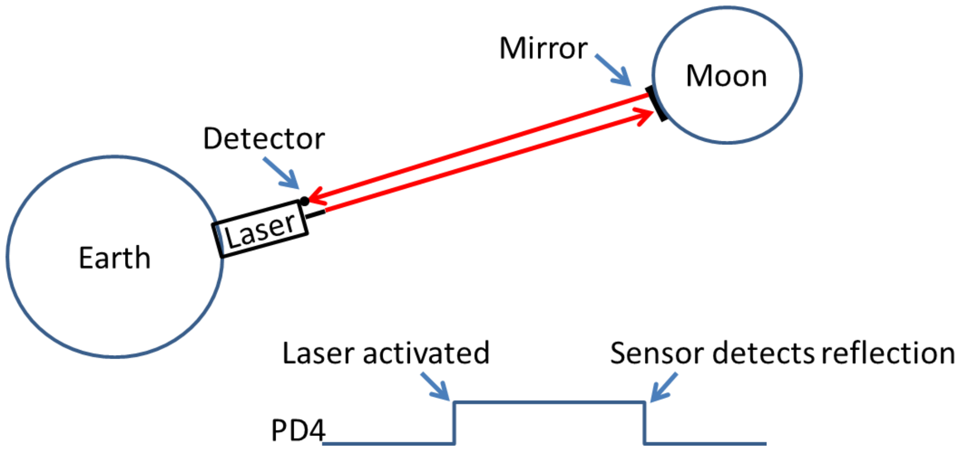
c) Let us assume we can configure the hardware for detecting either the positive edge or negative edge of an input by configuring the "Edge Select" bit of a configuration register (call it the Edge Select bit: ES). Draw the digital circuit to allow the Edge Detector to detect a positive edge when ES=1, and a negative edge when ES=0. You may now use multiplexers in addition to D-Flip Flops, and AND, OR, NOT gates (2 pts)



Name: Sean Gordon **Lab Section: F**

Question 5: Timer Accuracy (10 pts)

A laser that works similar to your Lab 7 Ping sensor is used to measure the distance to the Moon from Earth. The laser is fired at a mirror placed on the Moon, and a sensor attached to the laser detects when the reflection arrives back to the laser. As shown below. This is similar to an actual method used for measuring the Earth-Moon distance, see:
http://en.wikipedia.org/wiki/Lunar_Laser_Ranging_experiment



When the laser is activated, TIMER1's Input Capture pin is set to a 1. When the sensor detects the reflection, this pin is set to 0. A program has been written that uses Input Capture to compute the distance between the Earth and the Moon.bb

Given that Input Capture measures time in Timer ticks (i.e. clock cycles), how different can the programs calculation of the Earth-Moon distance be from the actual distance?

a) Explain what causes the error in the measured distance (5 pts)

As the speed of light is very fast, the clock speed has to be very fast to precisely record when the laser returns.

b) Compute the maximum error in distance for each the following speeds of the system clock used by TIMER1: 16MHz, 8MHz, 1MHz, 1KHz (5pts)

c = 299,792,458 m/s, 1 MHz = 1000000 Hz

16MHz: 299,792,458 * 1/16000000 Hz = 18.737m

8MHz: 299,792,458 * 1/8000000 Hz = 37.474m

1MHz: 299,792,458 * 1/1000000 Hz = 299.792m

1KHz: 299,792,458 * 1/1000 Hz = 299,792.458m

Name: Sean Gordon **Lab Section: F**

Collaboration Documentation

List the people (First and Last name) you collaborated with: __________________________.

For each collaborator, describe the manner in which you collaborated:

1)

2)