ASSIGNMENT –4(ASSEMBLY OF HARDWARE AND SOFTWARE)

Exercise 1 –To construct a circuit which will record and display the length of time taken by the throwing arm on the Mangonel to pass between the sensors mounted on the chassis. Implement the real hardware on the breadboard, at the input side of the Arduino. Demonstrate the hardware and software by attaching to a real Mangonel.

Hardware Required

- CD4027, CD4081, CD4543
- Arduino Uno
- Seven Segment Display
- Single core connecting wires
- Digital Trainer Kit

Theory

The circuit is required to:

- Accept the individual inputs from the two sensors on the Mangonel,
- Combine the two signals into one,
- Convert the two short pulses into one long pulse,
- Deliver this pulse to the relevant input on the Arduino Board,

The input signals:

The input signals received from the sensors on the Mangonel are shown in Figure 1 below:

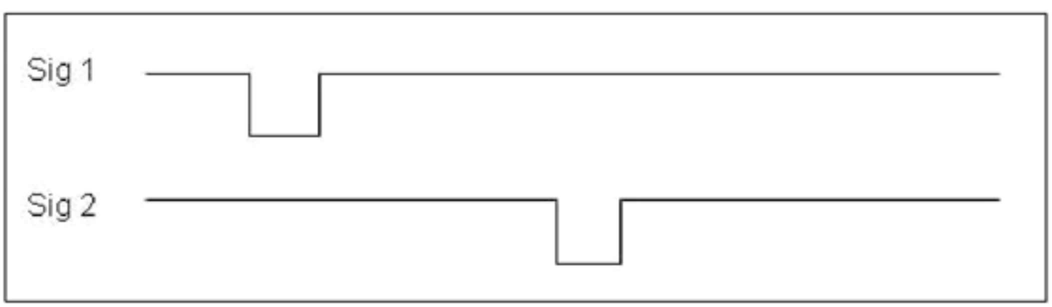


Figure 1: Mangonel sensors signals

Required signal:

The signal required for the Arduino program is shown in figure 2:

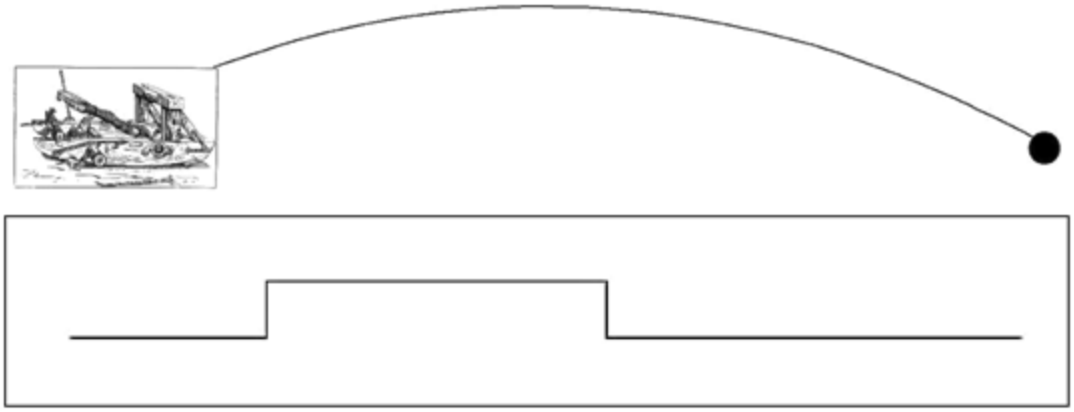


Figure 2:Required signal for Arduino.

How do we achieve the required signal?

In order to make one signal out of the two, we first need to combine the two individual signals. This is done by means of a combination logic setup. You should by now be familiar with the basic logic circuits as shown below.

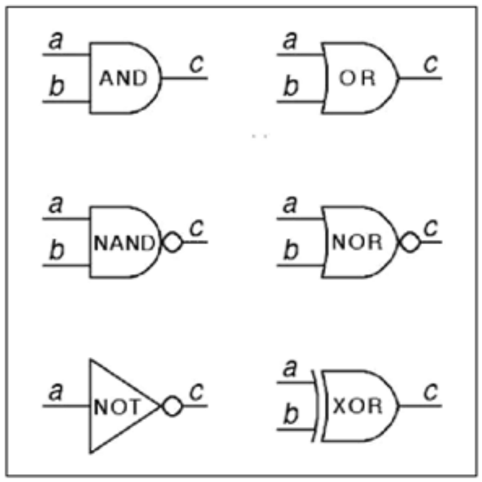


Figure 3:Basic logic gates

Of the devices above, the best choices for combining our two signals into one are the AND or NAND gates as either will preserve the signal from each individual sensor whilst also combining the two signals into one. It is important to note that NAND is simply inverted AND, similarly, if we invert NAND, we get AND again.

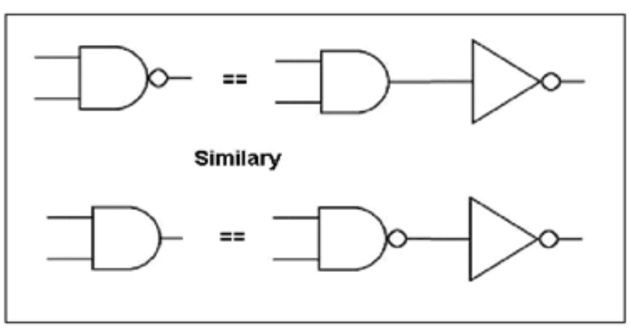


Figure 4:Logic inversion

The output from NAND and AND gates respectively to the input signal provided are shown below.

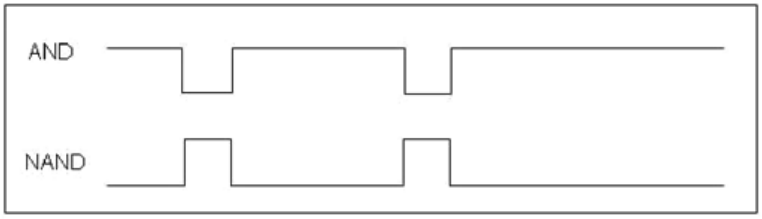


Figure 5:AND and NAND logic outputs

Figure 5 shows how the logic signals from the inputs are each maintained but combined into one signal. The next task is to convert the two small pulses into one long pulse.

Looking at the AND output signal, it could be argued that we already have a long pulse from this step alone – however, the long pulse between the two short pulses given by the AND logic is not an accurate measurement of the transit time of the arm because the pulse length is shortened by the width of the second pulse. We need to generate a single long pulse as shown in figure 6 which extends from either the rising or falling edge of the first pulse, to the corresponding edge of the second pulse.

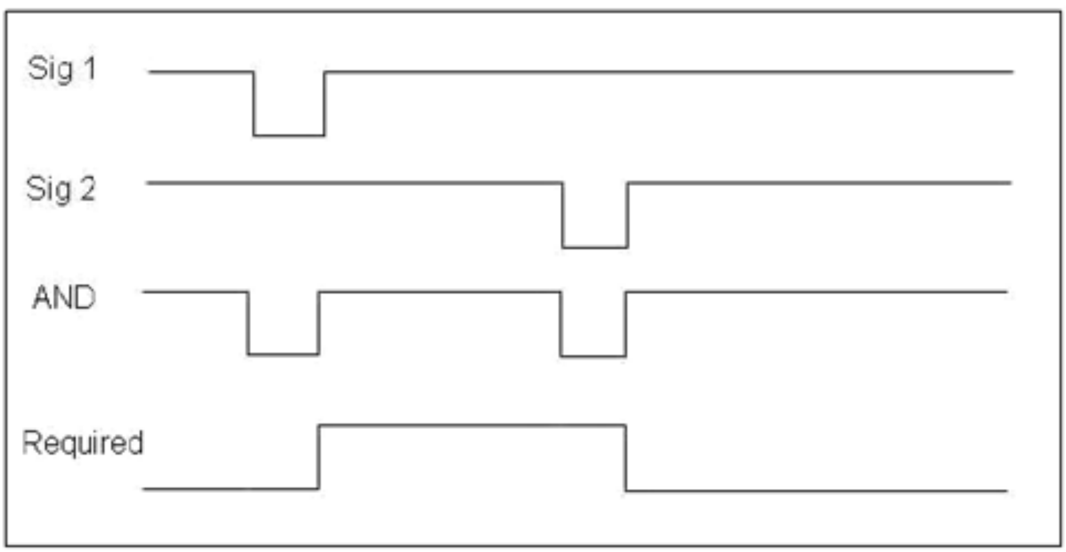
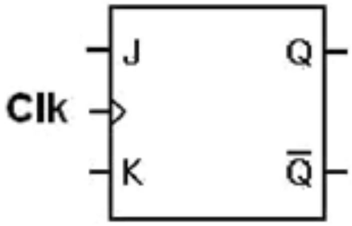


Figure 6: Logic comparison chart

This can be achieved by using the two shorter pulses in the output of the AND gate to "toggle" or "clock" the output of a latching logic device. For this purpose, we are going to use a J-K flip-flop.



INPUTS					OUTPUTS	
S_D	C_D	CP	J	K	Q_{n+1}	$\overline{Q}_{n+1}$
L	L	$\uparrow$	L	L	no change	
L	L	$\uparrow$	H	L	H	L
L	L	$\uparrow$	L	H	L	H
L	L	$\uparrow$	H	H	$\overline{Q}_n$	Q_n

Figure 7:J-K flipflop and function table.

The J -K flip- flop is the most versatile of the basic flip -flops. It has two inputs, traditionally labelled J and K. If J and K are different then the output Q takes the value of J at the next clock edge. If J and K are both low then no change occurs. If J and K both are high at the clock edge then the output will toggle from one state to the other. This toggle application means that we can use the two signal pulses from the output of the AND gate to cause the output of the flip-flop to change state, hence allowing us to create one long pulse between the corresponding edges of the two shorter pulses. The device also has set and reset functions which will be useful for clearing the output states when we want to start a new timing run. Figure 7 shows the function table of the J-K flip-flop.

The output of out J-K flip-flop then goes to the appropriate input of the Arduino and from there the program in the Arduino will measure the length of the pulse and output the appropriate values to the various registers on the display.Hence, our circuit should wind up looking something like this:

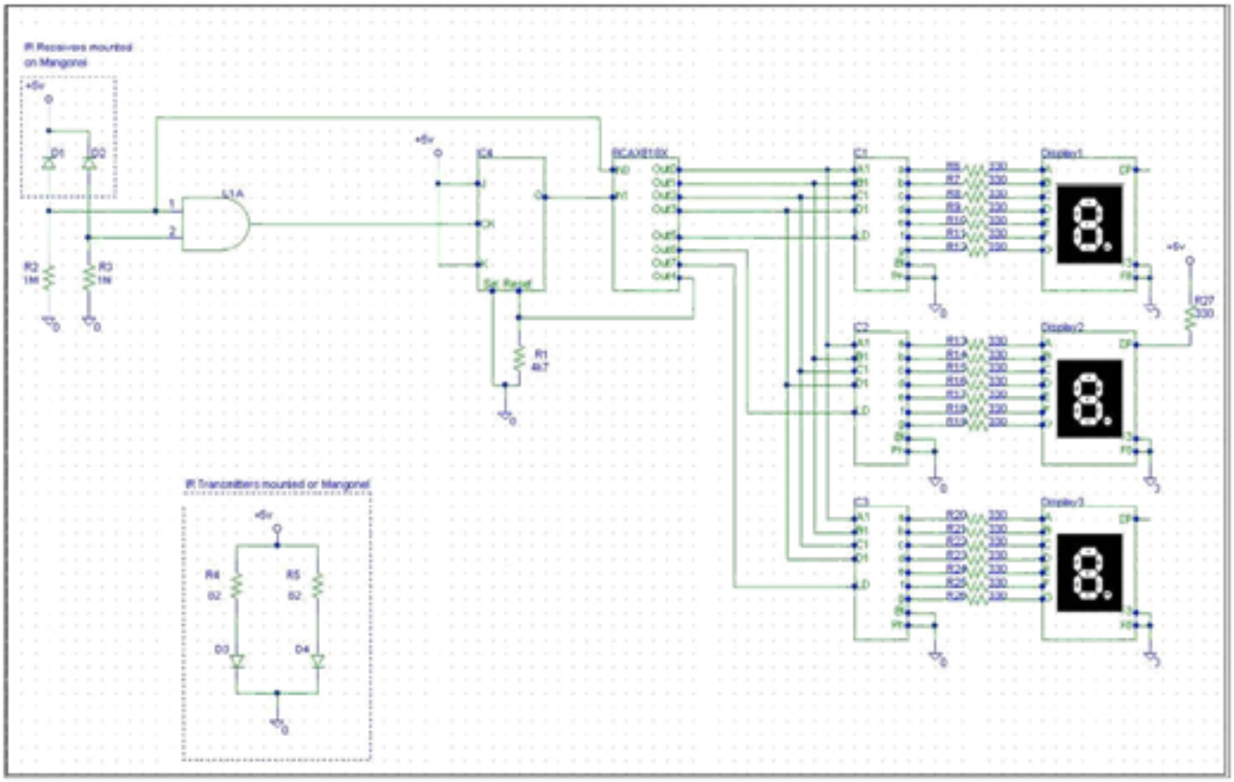
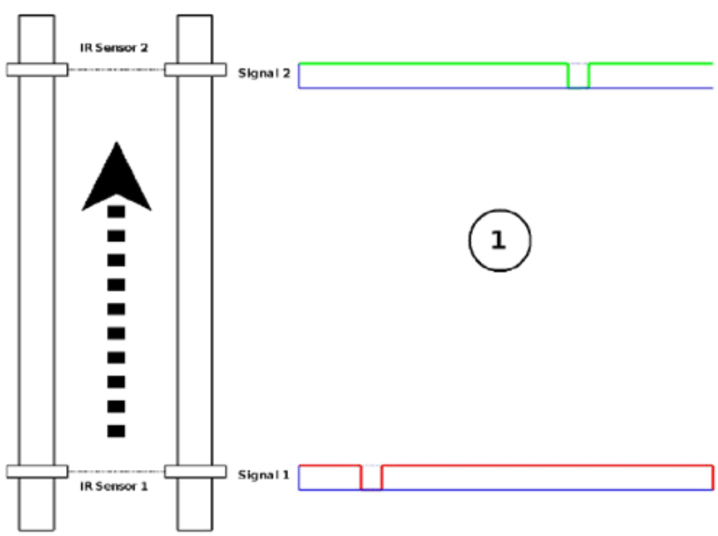


Figure 8: Circuit Schematic

Explain Figures 9succinctly, in fewer than 50 words per stage, each of the five stages involved in measuring the swing time of the Mangonel arm.



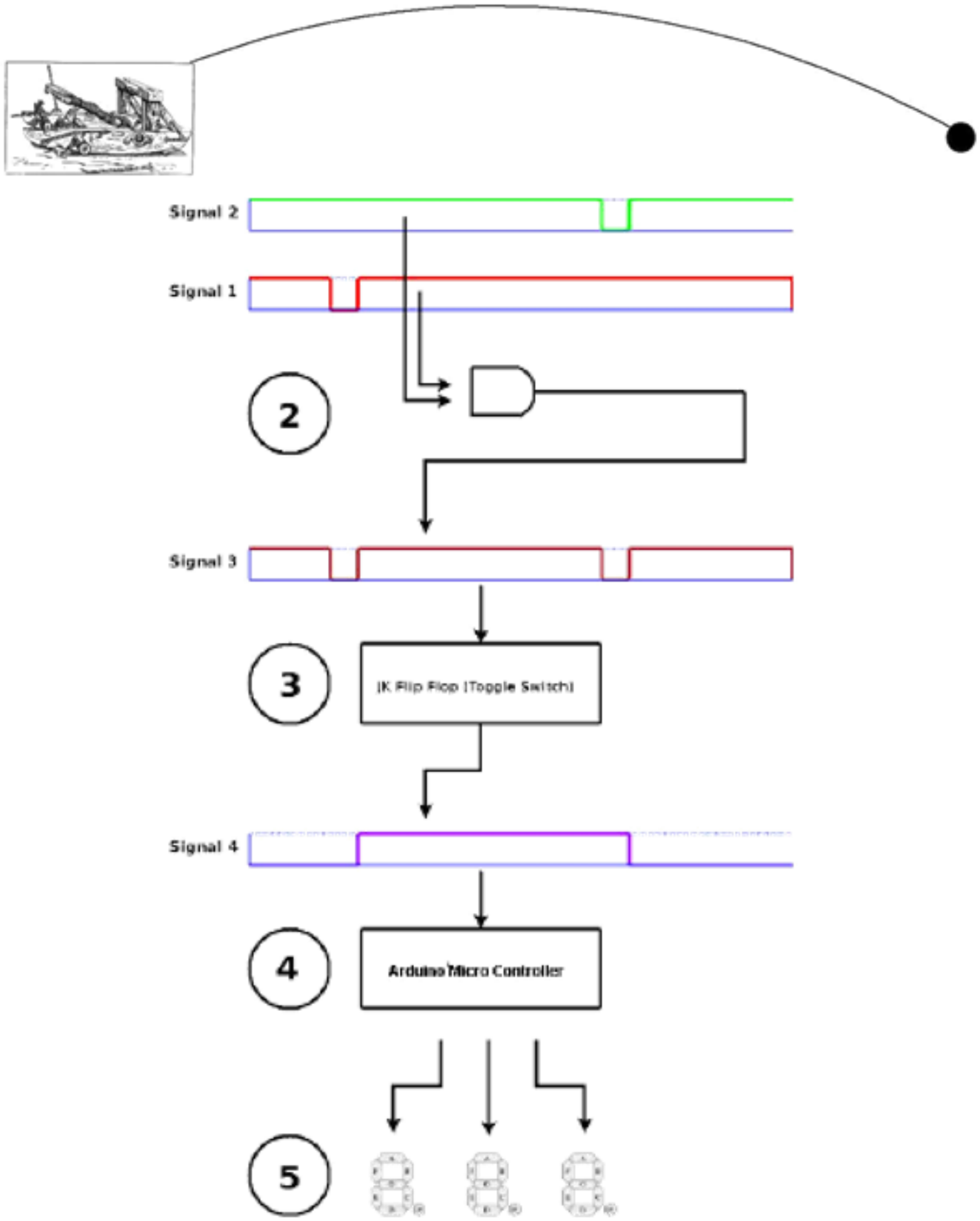


Figure 9: Stages 1 to 5

Step : 1
When mangonal arm is released from its initial position, it passes through two pairs of IR sensors when arm passes through sensor and receiver it will give an output LOW which will initially be LOW output. Hence two sensors will give LOW outputs when arm passes through them at different times respectively.

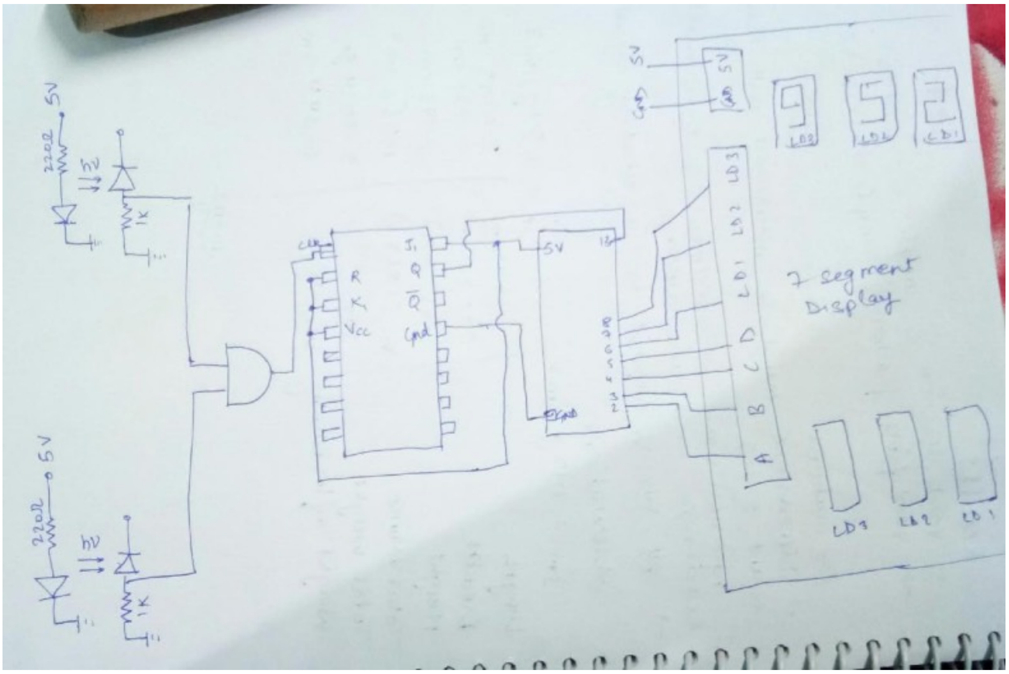
Step 2 :
Here output wave of two pairs of IR sensors is sent as input to the IC 4081. IC combines the two input into the single output waveform which is LOW at two distinct times. When the mangonel passed through sensors, these are that LOW points in the waveform of IC CD4081

Step 3:
In this stage, output waveform of CD4081 is set to clock of CD4027, which is used in toggle state. When clock goes from LOW to HIGH for the first time, the output waveform becomes HIGH and when the CLOCK goes from LOW to HIGH for second time, the output becomes LOW. Hence a single long pulse is created between corresponding two short pulses.

Step 4:
In this stage, long single pulse of CD4081 IC is given to the arduino micro controller under pulse in function which measures the time for which the input waveform is HIGH.

Step 5:
The time in microseconds is converted to milliseconds and displayed on 7 segment display with the help of CD4511 IC which converts BCD into its corresponding decimal level.

Schematic Diagram



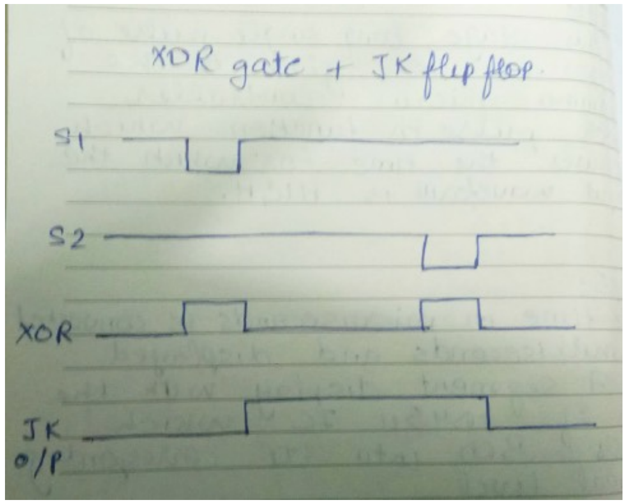
Reflection:

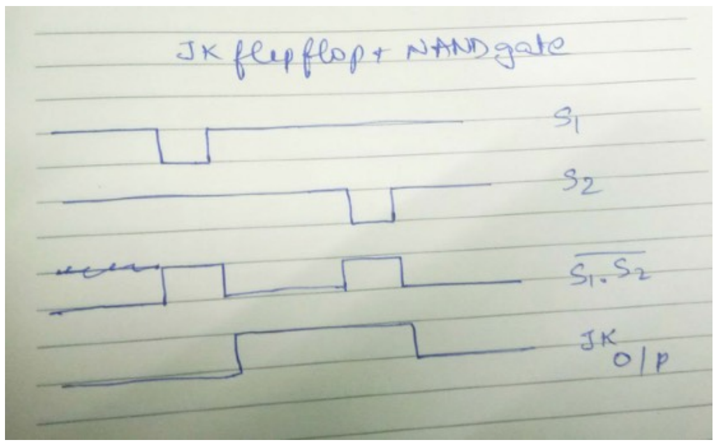
→ Through this experiment, I learnt how to use different logic gates to get the desired signal.

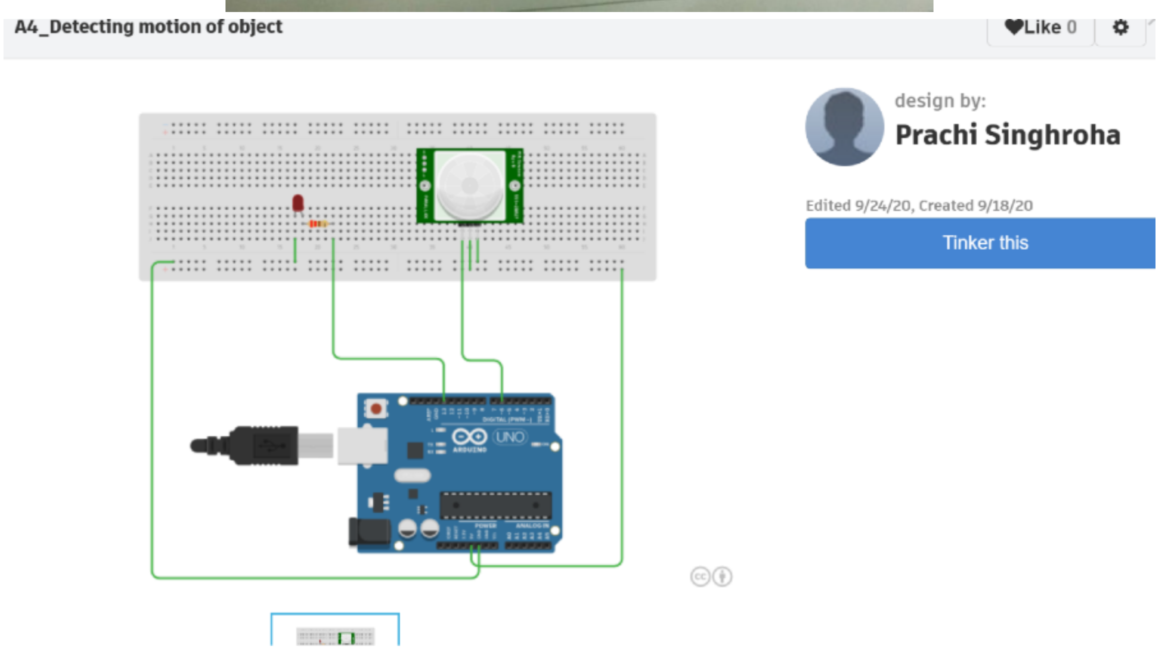
→ I was able to measure the time taken by arm to go from one pair of IR sensor to another and hence was able to calculate angular velocity.

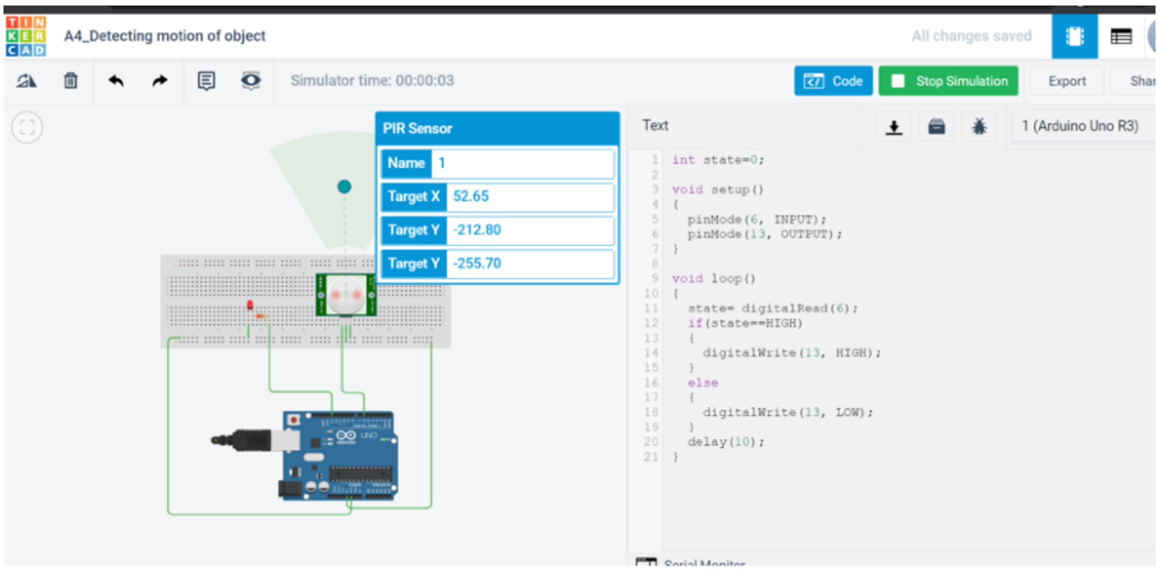
Assignment Tasks:

1. Obtain the required signal for Arduino shown in figure 2 using at least two different logic gates (explain using waveforms).


XOR gate + JK flip flop
S1
S2
XOR
JK o/p


JK flip flop + NAND gate
S1
S2
$\overline{S_1 . S_2}$
JK o/p

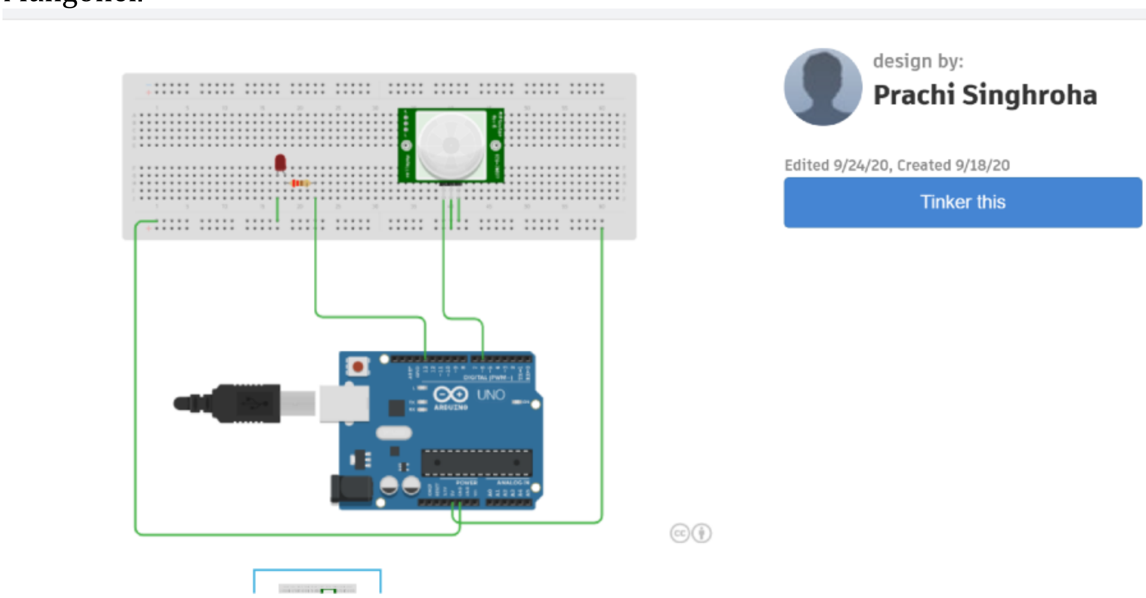

A4_Detecting motion of object
Like 0
design by:
Prachi Singhroha
Edited 9/24/20, Created 9/18/20
Tinker this
UNO

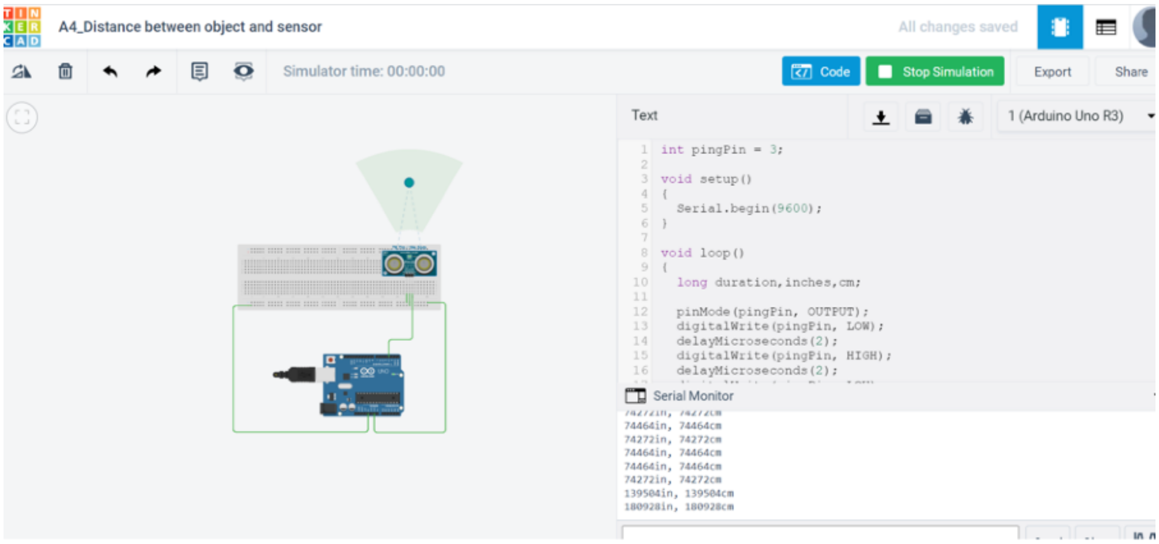


2. Using Tinkercad, design the followingsensor based micro-projects to:

   a. Detect the motion of an object, and
   b. Measure distance between an object and the sensor itself.

   Also mention the application of these project in terms of designing a Precision Mangonel.





CODE

```
int pingPin = 3;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  long duration,inches,cm;

  pinMode(pingPin, OUTPUT);
  digitalWrite(pingPin, LOW);
  delayMicroseconds(2);
  digitalWrite(pingPin, HIGH);
  delayMicroseconds(2);
  digitalWrite(pingPin, LOW);
  delayMicroseconds(2);

  pinMode(pingPin, INPUT);
  duration = pulseIn(pingPin, HIGH);

  inches= microsecondsToClockCycles(duration);
  cm= microsecondsToClockCycles(duration);

  Serial.print(inches);
  Serial.print("in, ");
```

```
 Serial.print(cm);
 Serial.print("cm");
 Serial.println();
 delay(100);
}
```

Applications:

→ We can use it to measure the distance travelled by projectile

→ We can use it to measure the time taken by mangonel arm to cross a 45° angle by noting time pulse as the arm moves from one position to another.