Ultrasonic Radar

Guillermo Cabrera

April 15th, 2023

Abstract

This formal report outlines the development of an ultrasonic radar using MSP430, an ultrasonic rangefinder, and a servo motor. The project aimed to design a low-cost and reliable radar system capable of detecting obstacles in a specific area. The report details the technical specifications of each component and the design process, including the software development and testing. The results show that the radar system successfully detects obstacles within the specified range and provides real-time feedback. The report concludes that the system can be used in various applications, such as autonomous vehicles and robotics, and offers several suggestions for future improvements. Overall, this project provides an effective solution for obstacle detection and offers a valuable contribution to the field of radar systems.

Introduction

The field of physics has seen tremendous growth and development over the years, leading to significant advancements in various sectors. With the increasing complexity of modern systems, it has become essential to have a deep understanding of different programming languages to operate and control these systems. In this context, the motivation of this project was to develop a system that would teach the author how to code in C while simultaneously combining different systems they had learned separately. Specifically, the author had already learned how to implement an ultrasonic sensor, a beeper, and LEDs in previous labs. Thus, for this project, they sought to combine all these elements while also learning how to operate a servo motor in a desired manner.

The ability to control multiple systems and integrate them into a functioning whole is a valuable skill in the field of physics, and this project aims to provide an opportunity for the author to develop that skill. This project also builds upon previous work in the field of physics, where researchers have developed different systems for controlling and monitoring various physical phenomena. In particular, this project is relevant to the development of systems that use ultrasonic sensors for detecting and measuring distance, and servo motors for precise control of mechanical systems.
Overall, the purpose of this project is to demonstrate the author's ability to apply their knowledge of different physics concepts and programming languages to build a functional system. The project will provide valuable insights into the practical implementation of ultrasonic sensors, beepers, LEDs, and servo motors, which are crucial components in many modern systems. Additionally, this project will contribute to the existing knowledge base in the field of physics by demonstrating the integration of different systems into a single functioning unit.

Theory

How the Ultrasonic Rangefinder works

Ultrasonic range finders are devices that use sound waves to detect and measure distance to a target object. They work on the principle of echolocation, which is similar to the way bats and dolphins navigate through their surroundings.
The ultrasonic range finder emits a short burst of high-frequency sound waves from a transmitter, which travels through the air towards the target object. The sound waves reflect off the object and return to the range finder, where they are detected by a receiver. The range finder then calculates the time it took for the sound waves to travel to the object and back, using the speed of sound in air, which is approximately 343 meters per second at room temperature.
The time-of-flight measurement is then used to calculate the distance to the target object by dividing the round-trip time by two and multiplying it by the speed of sound. This process is repeated multiple times per second, providing a real-time measurement of the distance between the range finder and the target object.
The accuracy of the ultrasonic range finder depends on several factors, including the frequency of the sound waves, the size of the transmitter and receiver, and the presence of any obstacles or reflective surfaces between the range finder and the target object. Higher frequency sound waves tend to provide more accurate measurements, but they are also more easily absorbed by air and other materials, limiting their range.
Overall, the physics behind the operation of an ultrasonic range finder relies on the properties of sound waves, and their ability to reflect off objects and travel through the air. This technology has numerous applications in various fields, including robotics, automotive, and industrial automation.

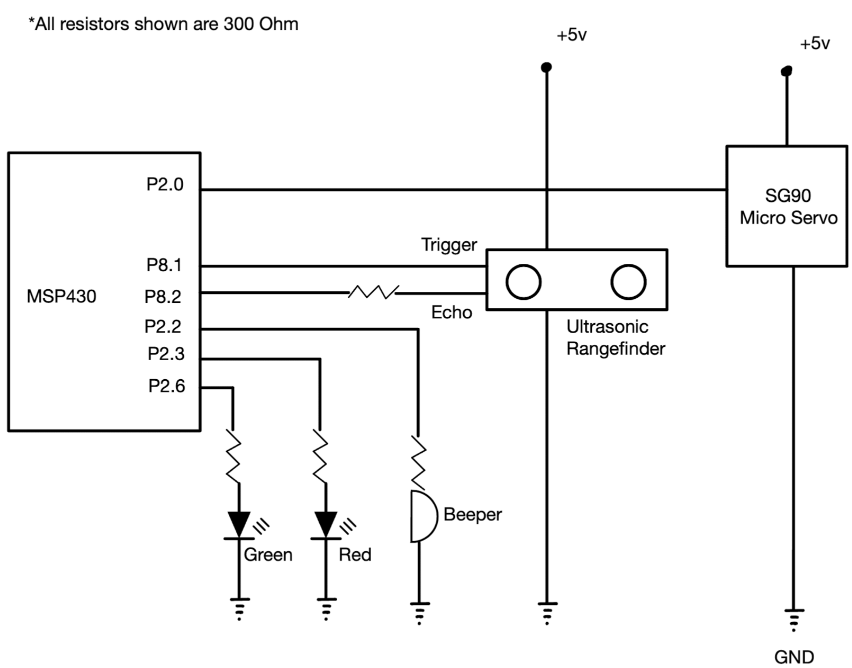
How the SG90 Servo Motor works

A SG90 micro servo motor is a type of electric motor that is designed to rotate to a specific position and hold that position. It consists of a small DC motor, a gear train, and a control circuit.
The motor inside the SG90 operates using the principles of electromagnetism. When an electric current is passed through the motor's coils, a magnetic field is generated, which causes the motor's rotor to rotate. The direction and speed of the rotation can be controlled by changing the direction and magnitude of the current passing through the coils.
The gear train inside the SG90 serves to convert the rotary motion of the motor into a more precise rotational motion of the output shaft. The gear ratio of the train determines the degree of rotation of the output shaft for a given rotation of the motor.
The control circuit inside the SG90 micro servo motor is responsible for controlling the position of the motor's output shaft. It receives a control signal in the form of a pulse-width-modulated (PWM) signal from an external source, such as a microcontroller or other electronic device. The control circuit then compares the duration of the pulse to a reference signal and adjusts the motor's position accordingly.
The reference signal determines the desired position of the motor's output shaft. The duration of the pulse determines the speed at which the motor rotates to the desired position. The control circuit continuously monitors the position of the output shaft and makes adjustments as necessary to maintain the desired position.

Overall, the physics behind the operation of a SG90 micro servo motor is based on the principles of electromagnetism and gear mechanics. These motors have numerous applications in various fields, including robotics, and automation.

Apparatus

The apparatus used in this project consisted of an MSP430 microcontroller, an ultrasonic range finder, a SG90 micro servo motor, a beeper, and two LEDs. The MSP430 microcontroller served as the central processing unit for the entire system. It was responsible for generating control signals for the ultrasonic range finder, the servo motor, the beeper, and the LEDs.
The ultrasonic range finder was used to detect the distance between the system and an object in front of it. It was mounted on a bracket and was able to rotate 180 degrees horizontally. The range finder emitted ultrasonic sound waves and received the echoes that bounced back from the object. The time taken for the sound waves to travel to the object and back was used to calculate the distance to the object.
The SG90 micro servo motor was used to rotate the ultrasonic range finder bracket to scan the surrounding area. The motor was controlled by the MSP430 microcontroller, which generated the appropriate PWM signals to move the motor to the desired position. The beeper and the three LEDs were used to provide feedback to the user. The beeper emitted an audible tone when an object was detected, while the LEDs provided a visual indication of the distance to the object.
The entire apparatus was powered by a near by outlet and my computer, which was connected to the MSP430 microcontroller and the various components. The ultrasonic range finder, the SG90 micro servo motor, the beeper, and the LEDs were all connected to the MSP430 microcontroller via appropriate input/output pins.
Overall, the apparatus used in this project was designed to detect and measure the distance to objects in the surrounding environment, and to provide feedback to the user in the form of audible and visual signals. It combined the use of an ultrasonic range finder, a servo motor, and various other components to create a functional system.



How to Operate

To operate the ultrasonic radar system, the custom program written in C is loaded onto the MSP430 microcontroller using the Code Composer Studio Integrated Development Environment (IDE). Once the program is loaded, the MSP430 triggers the SG90 micro servo motor to start rotating, allowing the ultrasonic range finder to scan the surrounding environment.

The ultrasonic range finder sends out a series of high-frequency sound waves and measures the time it takes for the sound waves to bounce back from any nearby objects. This data is transmitted to the MSP430 microcontroller, which uses it to calculate the distance to each object.

As the ultrasonic range finder scans the environment, the system's LED lights and beeper are activated based on the data received from the range finder. The LED lights provide a visual indication of the distance to nearby objects, while the beeper emits an audible tone that increases in frequency as the distance between the system and an object decreases.

To monitor the operation of the system in real-time, the custom program is run on a computer. The program provides a graphical interface that displays the current position of the ultrasonic range finder and the distance to nearby objects. This interface allows the operator to easily visualize the system's operation and make any necessary adjustments.

Results

The ultrasonic radar successfully performed the intended function of mapping out distances with reasonable accuracy, within a certain distance range, and providing an indication of object proximity using the LED and beeper. The radar exceeded initial expectations in terms of accuracy and effectiveness.

Problems Encountered:

During the development process, several issues were encountered and resolved. The first problem was that the servo motor and sensor needed to work simultaneously. Initially, both codes were placed in the same program while loop, resulting in the motor rotating completely to the right and left before the sensor could send out an echo signal. To solve this issue, the sensor code was placed inside the servo's while loop, enabling the sensor to send out a pulse with every angle increment. However, this resulted in another problem where the motor was very slow when an object was far away, and the speed increased significantly when an object was closer. The reason for this was that the time taken for the pulse to travel to and from the object was longer for objects at greater distances, causing a coupling effect that affected the motor's speed. To resolve this issue, all the delay cycles except for the 10 microsecond one used for the sensor were removed, which improved the motor's speed to a reasonable level.

The second problem was encountered when trying to use the beeper. Initially, it was assumed that PWM was required, necessitating the use of another timer. Timer A0 and A1 were already in use for the servo and sensor, respectively. Therefore, timer B was utilized. However, it was later discovered that PWM was not required, and the beeper could be turned on and off inside the motor's while loop. This created a frequency that was combined with an if statement to ensure that the beeper only turned on when an object was detected within a specific distance range.

As the beeper code was also included in the sensor while loop, the beeper was coupled to the sensor, causing the while loops to run faster as objects moved closer to the sensor. This resulted in a higher frequency of the beeper, providing an audible indication of an object's proximity.

I demonstrate this in the video show in this link: https://youtube.com/shorts/M4vXiS4kbfU

In summary, the issues encountered during the development process were successfully resolved, resulting in an ultrasonic radar that exceeded initial expectations in terms of accuracy and effectiveness.

Discussion

The results of this project show that the implemented radar system was successful in achieving the goals that were set out. The system was able to map out distances with reasonable accuracy within a certain distance range and the addition of the LED and beeper provided a convenient and intuitive way to indicate the proximity of the detected object.
Overall, the radar performed very well and exceeded expectations. One of the key successes was the integration of the servo motor and the ultrasonic sensor to work simultaneously. By putting the sensor code inside the servo's while loop, the radar was able to send out a pulse in every angle increment, resulting in more precise distance measurements.
One area that could have been improved is the speed of rotation of the servo motor. It was observed that when the detected object was far away, the motor would rotate very slowly and would increase speed significantly when an object was detected close to it. This was due to the coupling effect of the time it takes for the pulse to travel to and from the object, which affected the motor speed. However, by removing all the delay cycles except for the 10-microsecond one used for the sensor, the speed of the motor was improved.
Another area that could have been improved is the accuracy of the distance measurements. While the radar was able to map out distances within a certain range with reasonable accuracy, the measurements were not precise enough to be used as a measuring device. To improve the accuracy, it would be necessary to use a higher resolution ultrasonic sensor, or implement a more sophisticated algorithm to process the ultrasonic signals.
In conclusion, the implemented radar system was successful in achieving the desired goals, but there is still room for improvement. With further improvements, the system could be used in a wide range of applications, such as obstacle detection, autonomous navigation, and robotics.

Conclusion

In conclusion, the construction of the radar device was certainly worth the effort and time put into it. Despite encountering a few technical issues along the way, the final product exceeded my initial expectations and proved to be an effective and efficient means of measuring distances and detecting objects.
Throughout this project, I gained a deeper understanding of the physics behind ultrasonic range finders and servo motors, and I also learned how to work with microcontrollers and programming languages such as C. Additionally, I developed skills in problem-solving and troubleshooting, as I encountered various obstacles and was able to find solutions to them through research and experimentation.
In retrospect, there are a few things that could have been done differently in the construction of the radar device. One improvement that could be made is to add a more precise and accurate means of measuring distances, such as using a laser rangefinder instead of the ultrasonic range finder. Another possible improvement is to add more sophisticated software and programming to allow for more advanced features, such as automatic tracking and mapping of moving objects.
Overall, I am satisfied with the outcome of this project and the knowledge and skills that I have gained through it. It was certainly a valuable learning experience and has opened up opportunities for future projects in the field of robotics and automation.

References

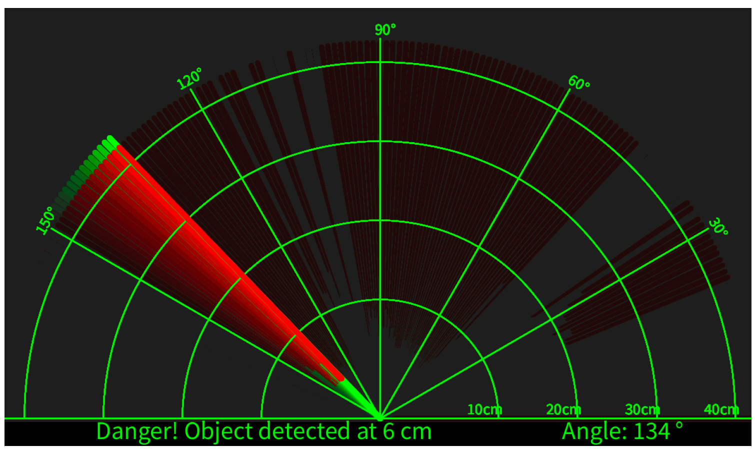
How to get a motor to rotate 180 degrees:
https://www.instructables.com/MSP430-Servo-Motor/

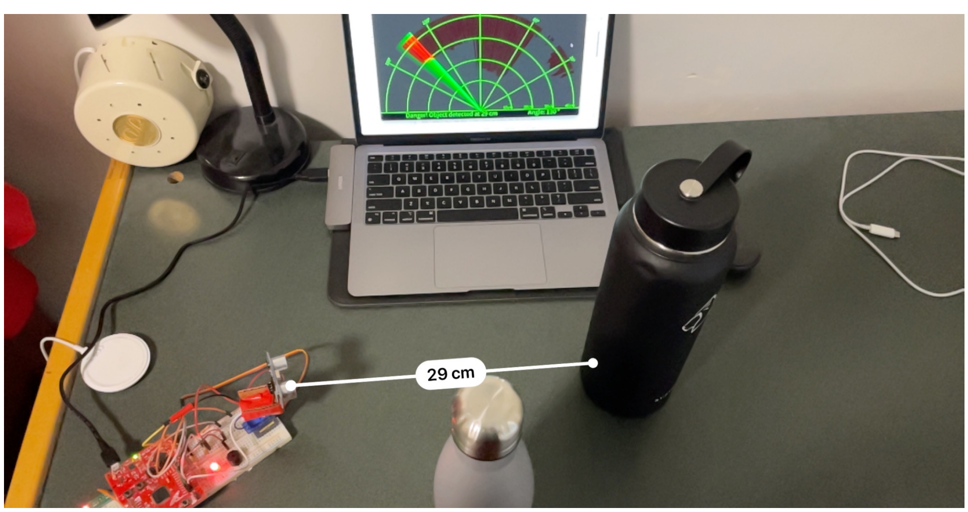
Basic idea on how to go about constructing the radar, as well as the code used for the graphic which I did not write, I only changed a few lines so that that the data from the MSP would be transmitted and create the radar map:
Video:
https://youtu.be/HNbfzXp5eHo

Code for graphic:
https://drive.google.com/file/d/1yrkBJvTP8ARQyV_4R82REC0QVLySU2Nd/view

The rest of the code on C was written on my own, and solved though trial and error and by looking at the users guide for my apparatus, as well as asking the web for general specifications.

Appendix





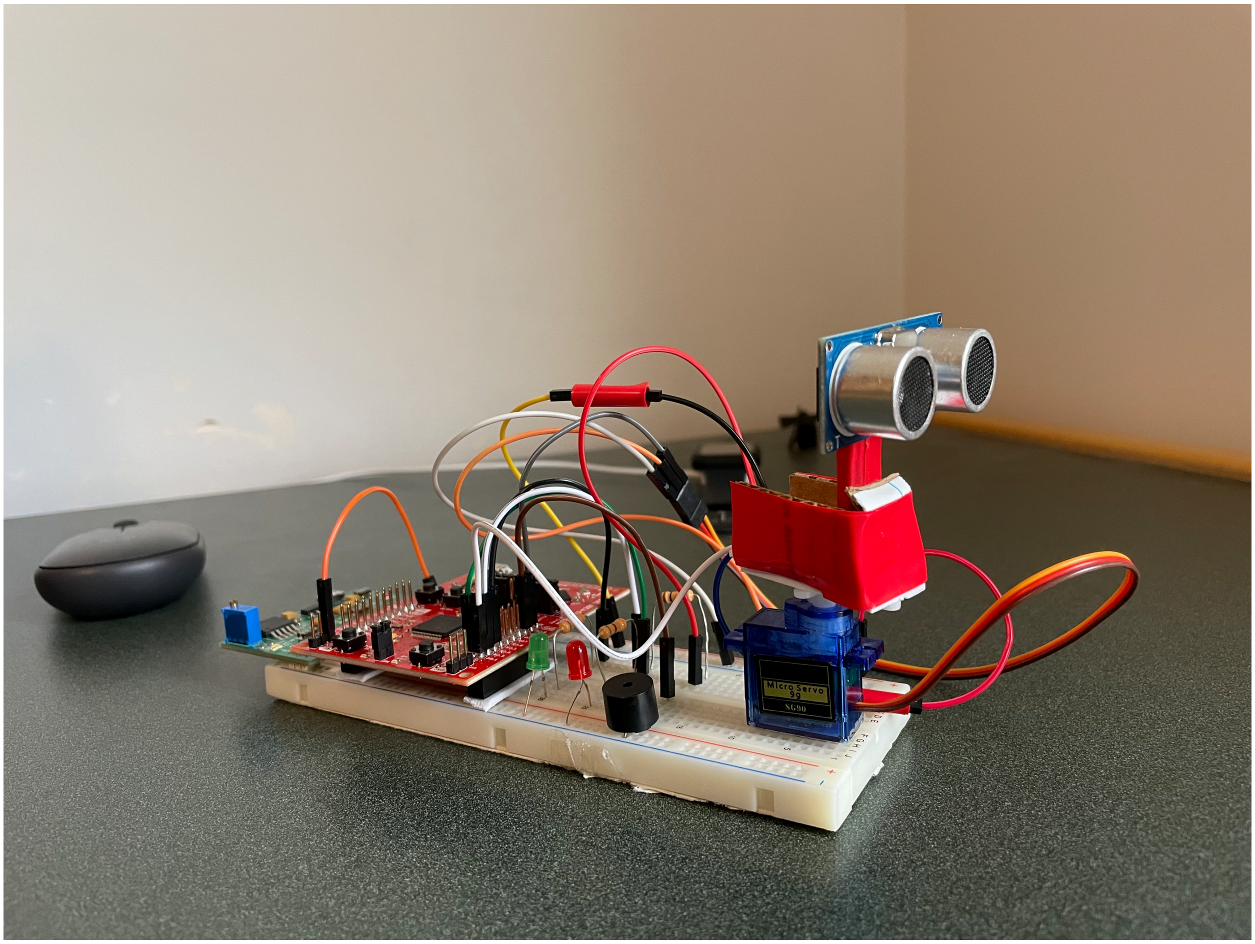
CODE ON CODE COMPOSER

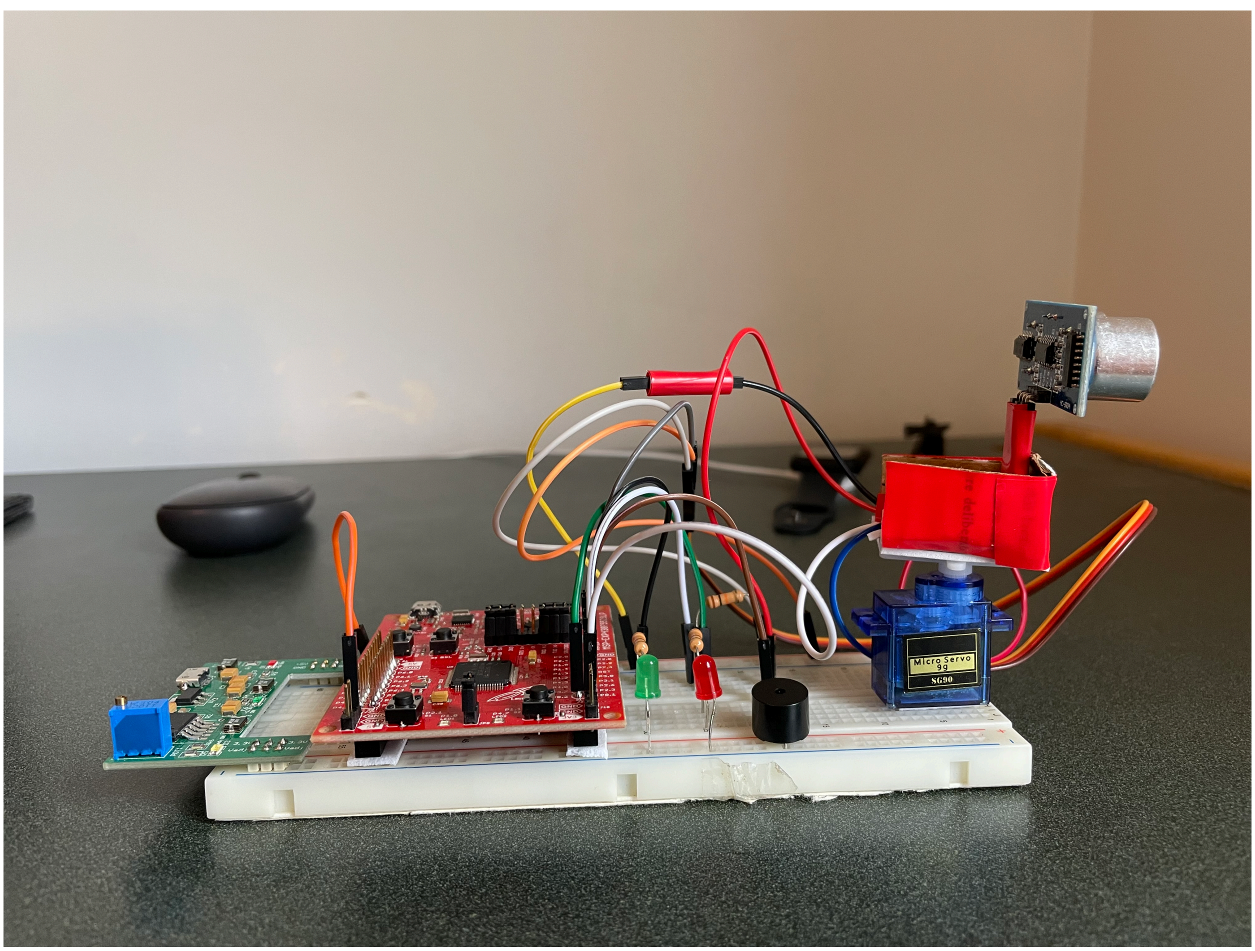
```c
#include <msp430f5529.h>
#include <stdio.h>

#define SERVO BIT0
#define TRIG BIT1
#define ECHO BIT2
#define RED BIT3
#define GREEN BIT6
#define BEEPER BIT2

unsigned int TXByte;
unsigned int time;
long distance;
int angle;

#define SERVO_PERIOD 20000 // servo period
#define min_angle 1000
//#define delaytime 50000
#define target_angle 2500 // minimum pulse width/duty cycle for servo
int current_angle = 1000; // minimum pulse width/duty cycle for servo



long getDistance(void);
void UART_send(int angle, int distance);


void main(void)
{
    WDTCTL = WDTPW + WDTHOLD; // Stop WDT



    // PWM setup
    P2DIR |= SERVO; // output servo pin on 2.0
    P2SEL |= SERVO; // pin 2.0 selected as PWM

    // ultrasonic sensor and UART setup

    P8DIR = TRIG;
    P8REN = ECHO;
    P8OUT = 0;              // Set Trig pin to be low
    UCA1CTL1 = UCSWRST;     // Reset the UART module
    P4DIR = BIT4;
    P4SEL |= BIT4 + BIT5; // Configure P4.4 and P4.5 for UART operation
    UCA1CTL1 |= UCSSEL_2; // Select SMCLK as the clock source for the UART
    UCA1BR0 = 109;          // Set the baud rate to 9600 (based on 1.048 MHz clock)
    UCA1BR1 = 0;
    UCA1MCTL = UCBRS_2;     // Set the modulation UCBRSx to 2
    UCA1CTL1 &= ~UCSWRST; // Enable the UART module
    TA0CTL = TASSEL_2 + MC_2 + ID_0 + TACLR; // SMCLK, continuous mode, no division, clear counter


    TA1CCR0 = SERVO_PERIOD - 1; // PWM period
    TA1CCTL1 = OUTMOD_7; // TA0CCR1 reset/set-high voltage // below count, low voltage when past
    TA1CTL = TASSEL_2 + MC_1 + TAIE +ID_0;  // Timer A control set to SMCLK, 1MHz, and count up mode MC_1


    P2DIR |= RED; // set red pin as output
    P2OUT &= ~RED; // turn off RED
```

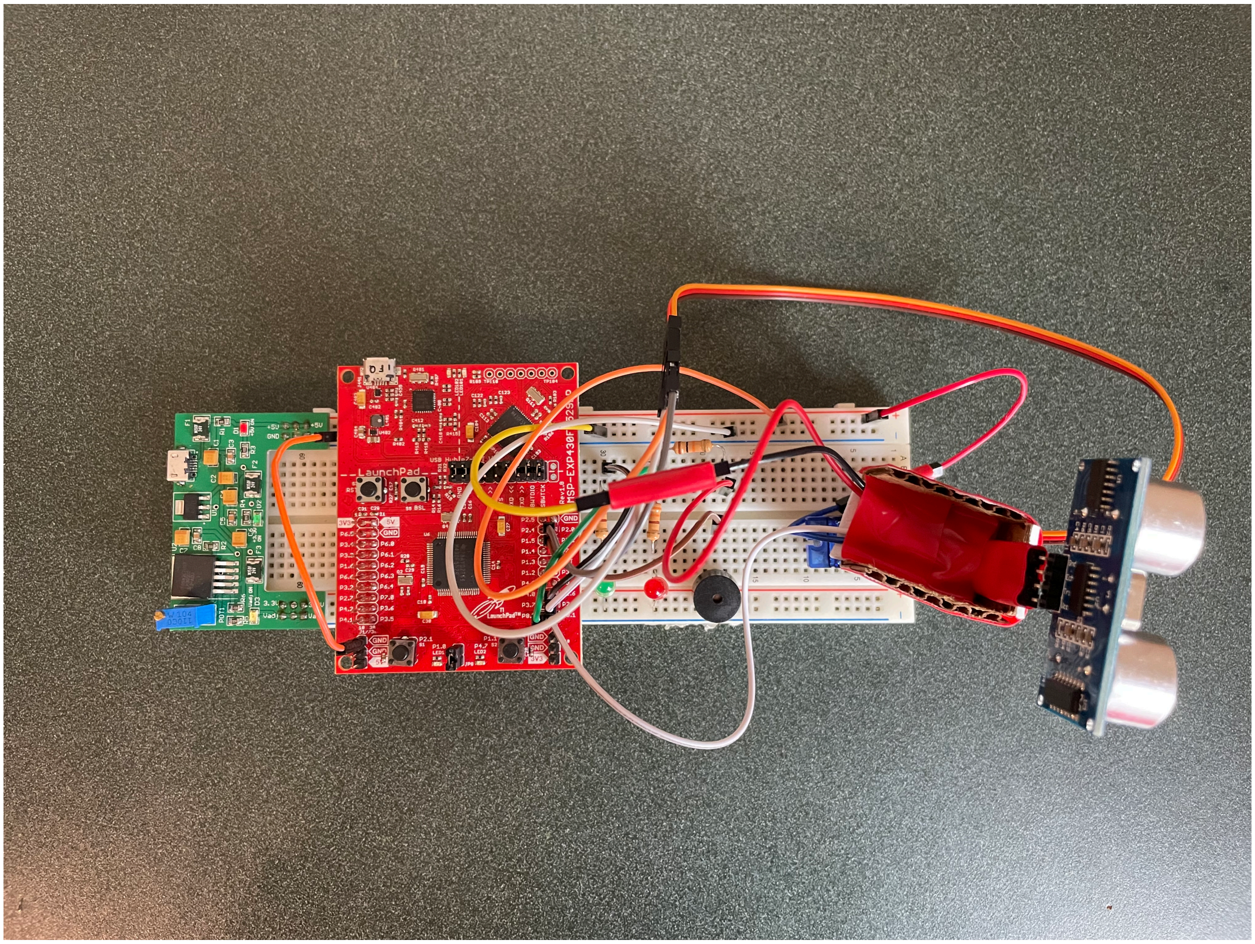
```
P2DIR |= GREEN; // set green pin as output
P2OUT &= GREEN; // turn off green

P2DIR |= BEEPER; // set beeper pin as output
P2OUT &= BEEPER; // turn off beeper



// SERVO action code

while(1)
{

while(current_angle > min_angle)
{
    current_angle -= 1;
    TA1CCR1 = current_angle;
    angle = (current_angle - 750) * 0.09;
    distance = getDistance();

    UART_send(angle, distance);
}

while(current_angle < target_angle)
{
    current_angle += 1;
    TA1CCR1 = current_angle;
    angle = (current_angle - 750) * 0.09;
    distance = getDistance();

    UART_send(angle, distance);

}

}

}
```

```
ong getDistance(void)

            if (time > 1764)
            {
                P2OUT &= ~GREEN;
                P2OUT |= RED;
/                 P2OUT |= BEEPER;
/                 P2OUT &= ~BEEPER;
/                 __delay_cycles(1000);
/                 P2OUT |= BEEPER;
```

```
            }
            else
            {
               P2OUT &= ~RED;
               P2OUT &= ~BEEPER;
               P2OUT |= GREEN;
               P2OUT &= ~BEEPER;
               __delay_cycles(1000);
               P2OUT |= BEEPER;

            }


            P8OUT |= TRIG;
            __delay_cycles(10);
            P8OUT &= ~TRIG;

           while (!(P8IN & ECHO));

            TA0CTL |= MC_2; // Start timer
            TA0R = 0;       // Clear timer

            /* next  line to use internal calibrated 1.048MHz clock: */

            while ((P8IN & ECHO));
            // read the counter and find the time in us
            time = TA0R;
            //TA2CCR1 = TA0R;

            long distance = time * 0.034 / 2;


           return distance;

}

void UART_send(int angle, int distance)
{
    char data[16];
    sprintf(data, "%d,%d,", angle, distance);
    int i = 0;
    while(data[i])
    {
      while (!(UCA1IFG & UCTXIFG));
      UCA1TXBUF = data[i]; // Transmit TXByte;
      i++;
    }
}
```


Micro Servo
9g
SG90


Micro Servo
9g
SG90


LaunchPad