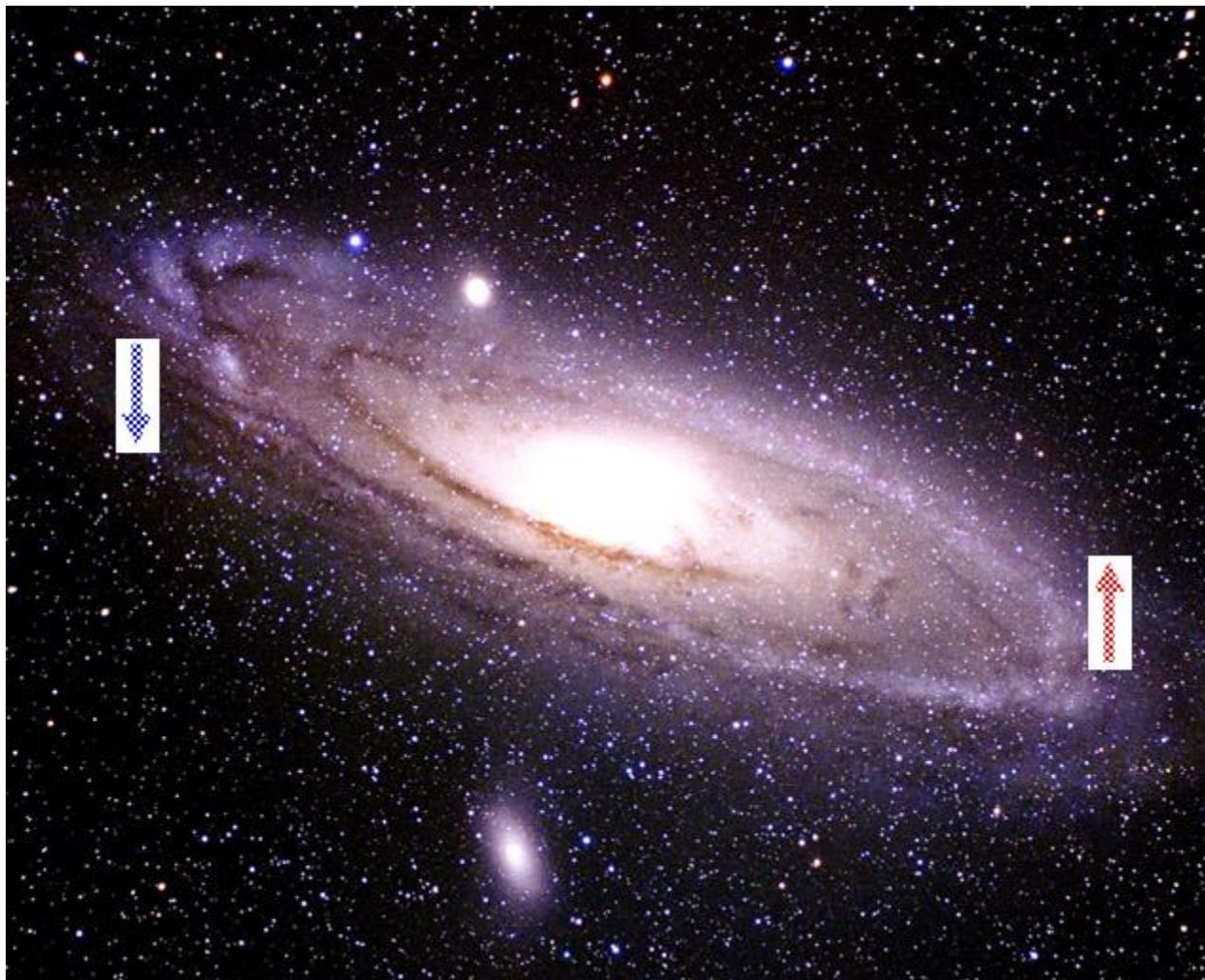


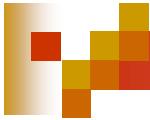
# SUMMIT 2007

Karl Jean-Francois-James  
Madison H.S.  
& Eldiquen Mangubat-IS 232



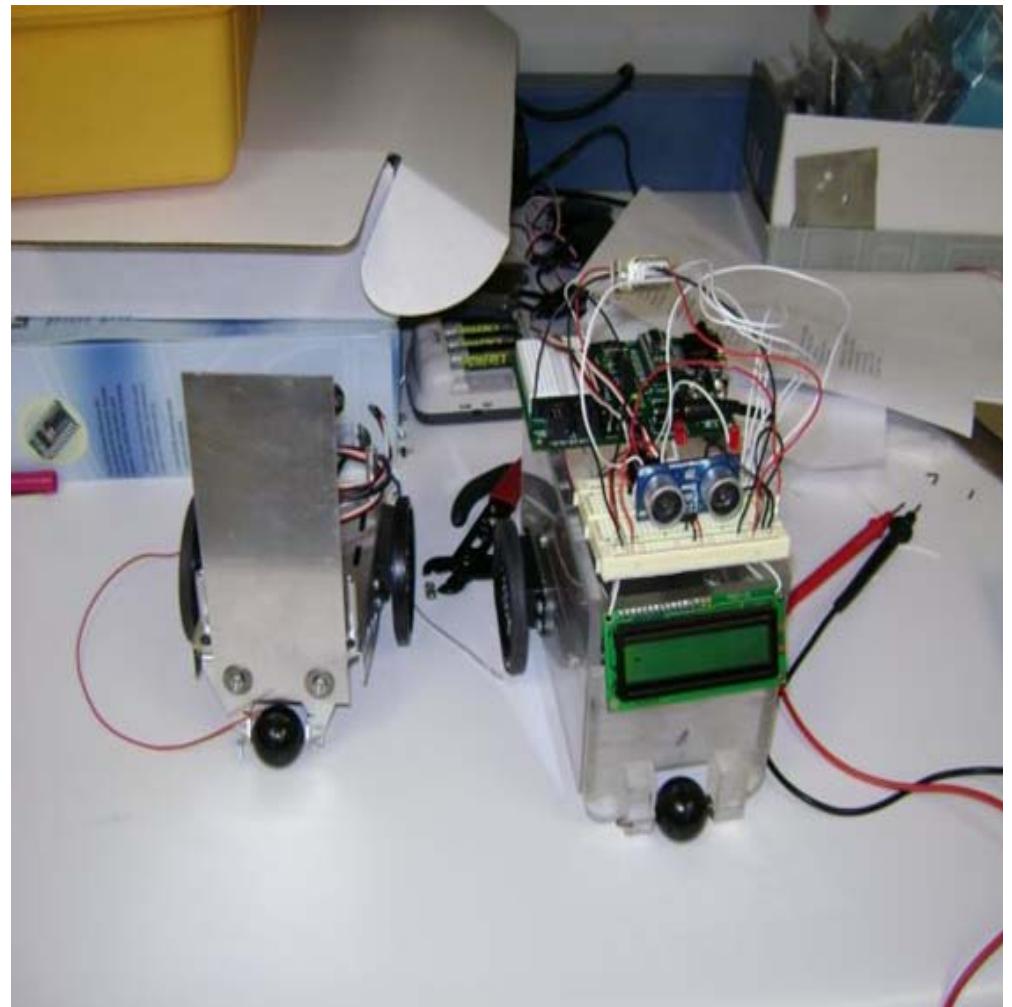
# Visualization of the Doppler Effect

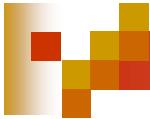




# Purpose

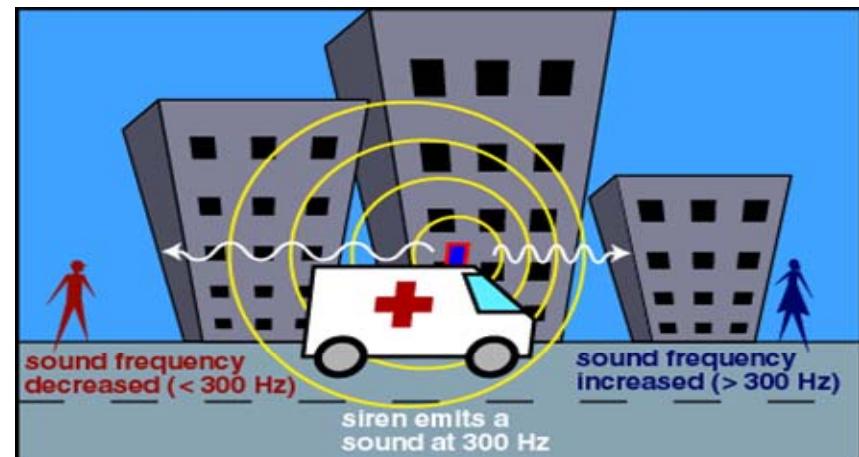
This instrument is one of the many that we will build which will help our students visualize scientific concepts that are abstract.





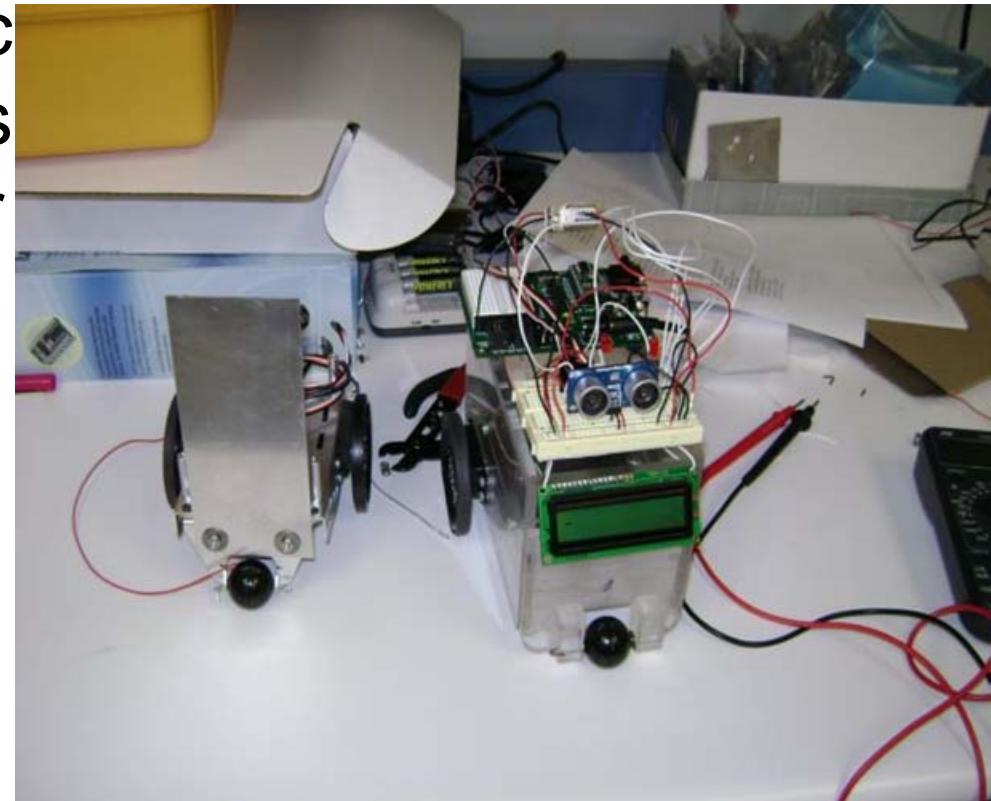
# Introduction

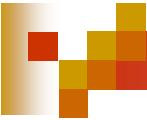
- Our project aims to make it easier for our middle school and high school students to comprehend how Doppler Effect works.
- This concept is an obscure concept to most students to visualize without the aid of machines such as the one we have just finished built.



# Introduction

Integrating Ultrasonic Sensors, servomotors and a microcontroller to demonstrate the Doppler Effect

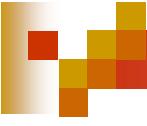




# Designing Process

Our design is a two-fold design.

Program the servomotor to drive the cart forward while transporting an ultrasonic sensor that emits and receives sound pulses.



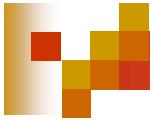
# Design

Program the servomotors to drive the cart backward



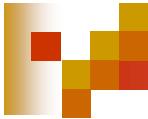
# Hypothesis

- Relationship between  $V_{source}$  and Shifted Frequency When the Source is moving away.
- Relationship between  $V_{source}$  and Shifted Frequency when the source is moving toward the stationary observer.

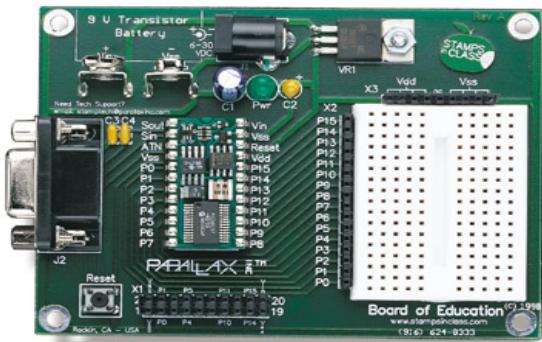
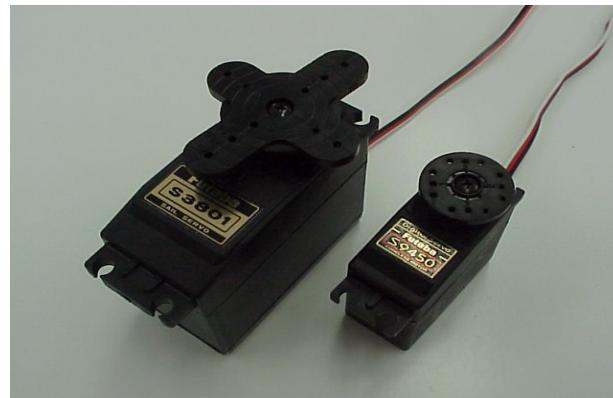


# List of Components

- Servomotors
- Boe-bot
- Basic Stamp
- Ultrasonic Sensor
- LCD

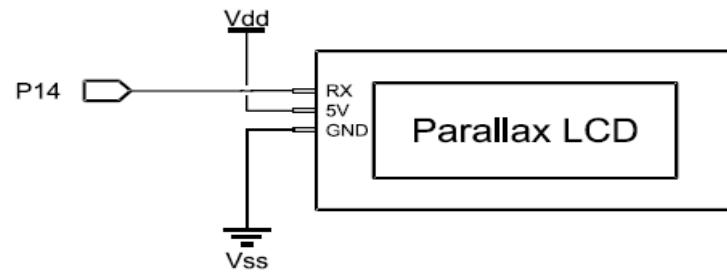


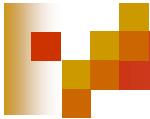
# Major Components





# LCD

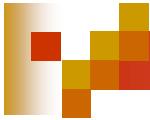




# Calibration of the Servomotors

- Insert the tip of Phillips screwdriver into potentiometer access hole.
- Continue turning the potentiometer until the motor stops turning.





# Centering the Servos

- ❖ The Basic Stamp is programmed to send to the servos a signal, instructing them to stay still.
- ❖ We used a screwdriver to adjust them so that they stay still.



# Calibration of the Servomotors

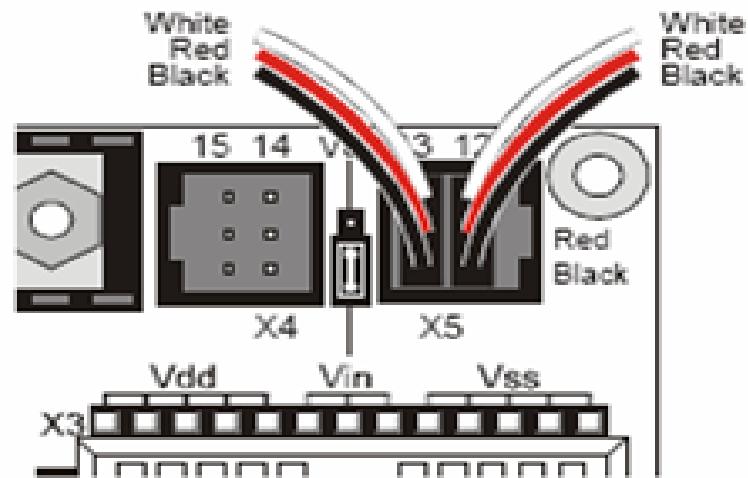
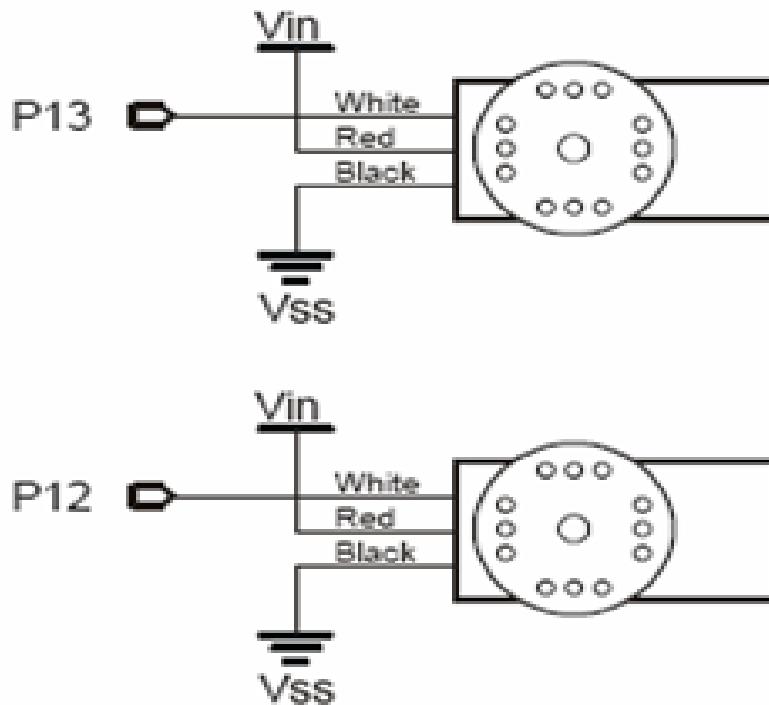
```
' Robotics with the Boe-Bot - CenterServoP13.bs2
' This program sends 1.5 ms pulses to the servo connected to
' P13 for manual centering.

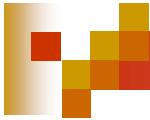
' {$STAMP BS2}
' {$PBASIC 2.5}

DEBUG "Program Running!"

DO
    PULSOUT 13, 750
    PAUSE 20
LOOP
```

# Servo Connection Schematics and Wiring Diagram





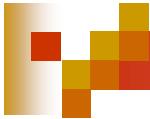
# Duration of a Pulse Width

- Pulse width of 850 corresponds to 1.7 ms
- Pulse width of 650 corresponds to 1.3 ms
- 1.7ms – Servo *connected to P1*
- 1.3 ms – Servo *connected to P0*
- 20 ms – *Pause duration*
- 1.6 ms – *Code overhead*



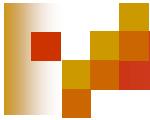
# Determination of the Duration of the length of the servo motor running

- ----- -----
- *24.6 ms – Total*
- If you want to run the servos for a certain amount of time, you can calculate it like this:
- *Number of pulses = Time s / 0.0246s = Time / 0.0246*
- Lets' say we want to run the servos for 3 seconds. That's
- *Number of pulses = 3 / 0.0246 = 122*



# How Can We Know How Long the ServoMotor is Working?

- In the program below, the ***EndValue*** of the **FOR...NEXT** loop is 122 which can be multiplied by 0.0246 to get 3 seconds
- FOR counter = 1 TO 122
- PULSOUT 13, 850
- PULSOUT 12, 650
- PAUSE 20
- NEXT



# Program that Controls the Motion of the Servo Motors

```
' Robotics with the Boe-Bot - ServosP13CcwP12Cw.bs2
' Run the servo connected to P13 at full speed counterclockwise
' and the servo connected to P12 at full speed clockwise.
' {$STAMP BS2}
' {$PBASIC 2.5}
DEBUG "Program Running!"
DO
PULSOUT 13, 850
PULSOUT 12, 650
PAUSE 20
LOOP
```

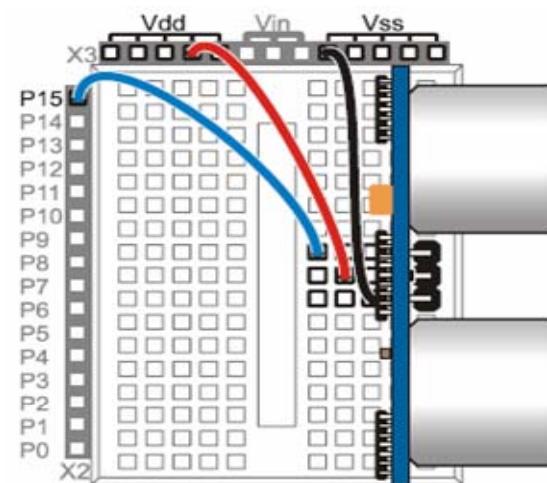
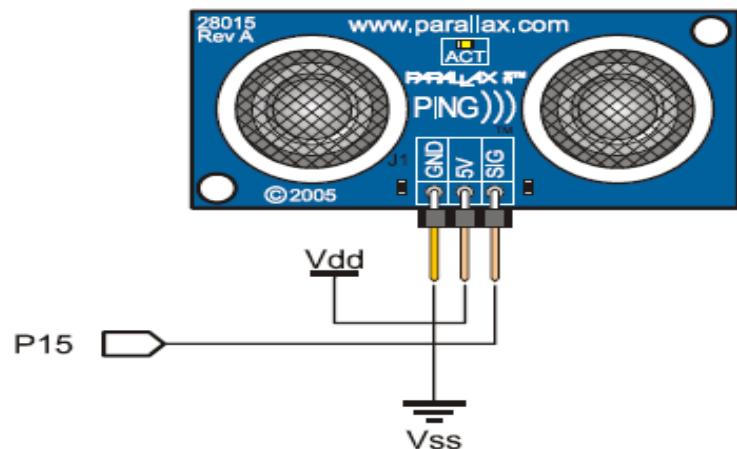


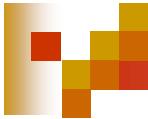
# The Ping Ultrasonic Sensor





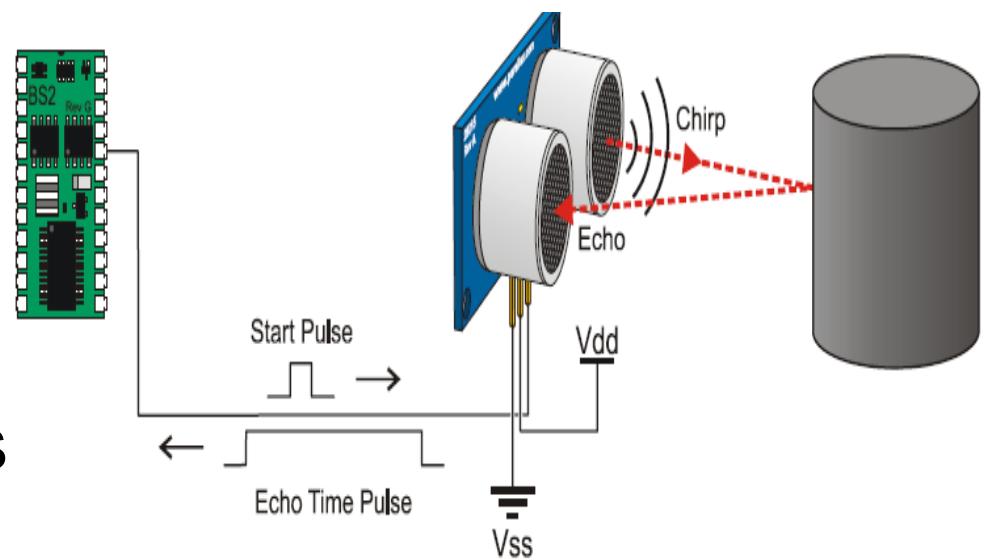
# The Schematic Diagram of the Ping

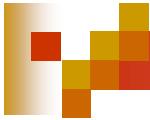




# How Does the Ping Sensor Work?

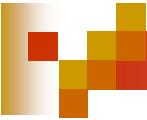
The Ping))) sensor sends a brief chirp with its ultrasonic speaker and measures the echo's return time to its ultrasonic microphone.





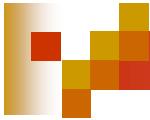
# Determination of Distance Using the PING)))™ Ultrasonic Sensor

- Provides distance measurements between moving or stationary objects.
- The time it takes for the echo to return from the target calculates the distance measurement.



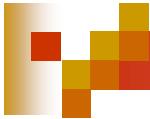
# Standards

- Standard 5 – Technology
- Engineering design is an iterative process involving modeling and optimization used to develop technological solutions to problems.



# Standards

- Technological tools, materials, and other resources should be selected on the basis of safety, cost, availability, appropriateness, and environmental impact.
- Technological processes change energy, information, and material resources into more useful forms.



# New York State Standards

- Mathematical analysis, scientific inquiry and engineering design, as appropriate to, pose questions seek answers and develop solutions.
- Students will develop an understanding of computer programming and attain some facility in writing computer programs.



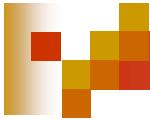
# Programming

## Variables

- D1 VAR Word ' D1 = Initial distance (cm)
- D2 VAR Word ' D2 = Final distance (cm)
- v VAR Word ' v = Velocity (cm/s)
- t VAR Word ' t = Time (s)
- fprime VAR Word ' fprime = Frequency of Observer (Hz)
- counter VAR Word ' Counter for the servo

# Constants

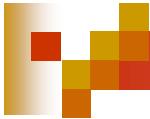
- v0 CON 34000 ‘v0 = Velocity of Ultrasonic (cm/s)
- f CON 40000 ‘f = Frequency of Ultrasonic (Hz)
- CmConstant CON 2260 ‘CmConstant = Distance Constant



# Programming

## Ultrasonic Sensor Taking Initial Distance Reading

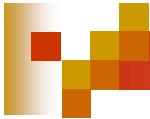
- PULSOUT 15, 5
- PULSIN 15, 1, t
- PAUSE 2000
- D1 = CmConstant \*\* t
- DEBUG HOME, "Initial Distance Measured: ", DEC3 D1, " cm", CR



# Programming

## Ultrasonic Sensor Taking Final Distance Reading

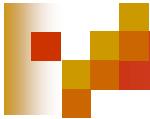
- PULSOUT 15, 5
- PULSIN 15, 1, t
- PAUSE 100
- D2 = CmConstant \*\* t
- DEBUG "Final Distance Measured: ",DEC3 D2, " cm ", CR



# Programming

## Calculations for Velocity & fprime

- $v = (D2-D1)/3$
- DEBUG "Velocity: ", DEC3 v, " cm/s ", CR
- $fprime = f^*(v0/(v0-v))$
- DEBUG "f": ", DEC5 fprime, " hz "



# Programming

## LCD Display

- SEROUT 3, 84, [22, 12]
- SEROUT 3, 84, [128, "Initial Distance", DEC3 D1, " cm " ]
- PAUSE 2000
- SEROUT 3, 84, [128, "Final Distance", DEC3 D2, " cm " ]
- PAUSE 2000
- SEROUT 3, 84, [128, "Velocity", DEC3 v, "cm/s" ]
- PAUSE 2000
- SEROUT 3, 84, [128, "Frequency", DEC5 fprime, "hz"]
- PAUSE 2000



# Thanks to

- Dr Kapila
- Dr Noel



# Gratitude

- Padmini
- Aschuman
- Daniel
- Valentin



# Thanks To

- Billy
- Jared
- Nathan



# Thanks

Special Thanks to Shing Lik Wong, our mentor.