PHYS 204

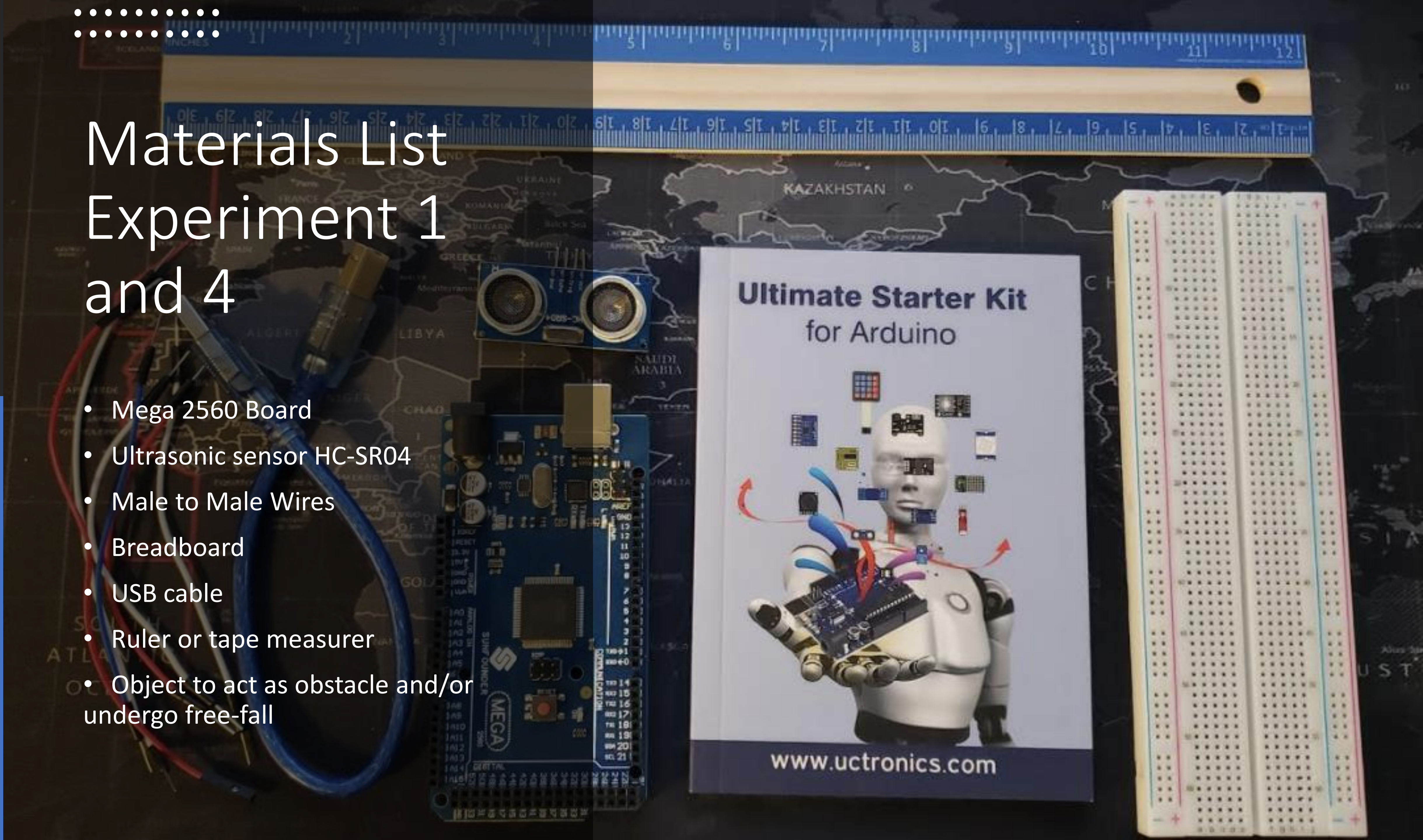
Introduction

- The best way to learn physics is to *do* physics. The fundamental principles teach you the physical principles behind the workings of the universe as well as sculps your solutions to complex logical problems.

- This project was developed using basic scientific methods and is broken down into 6 parts. Each step builds from the previous resulting in a full experiment.

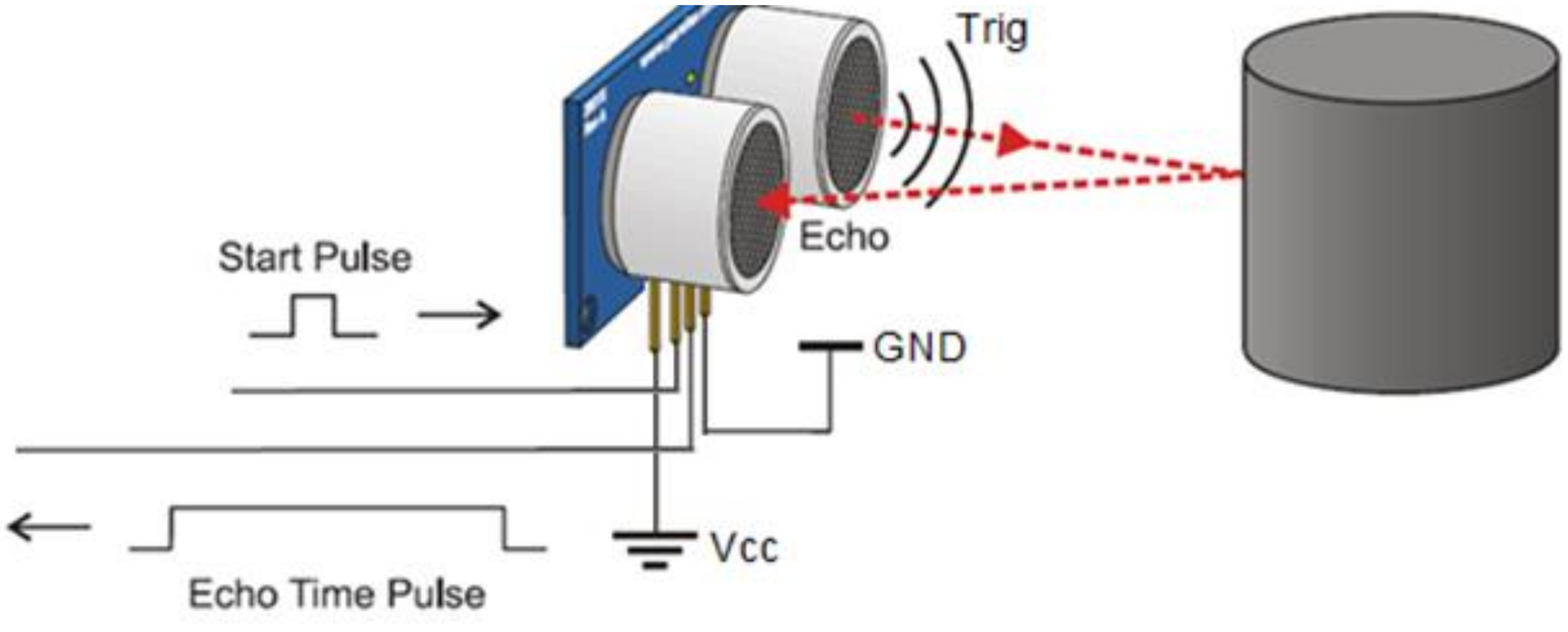
Materials List Experiment 1 and 4

- Mega 2560 Board
- Ultrasonic sensor HC-SR04
- Male to Male Wires
- Breadboard
- USB cable
- Ruler or tape measurer
- Object to act as obstacle and/or undergo free-fall



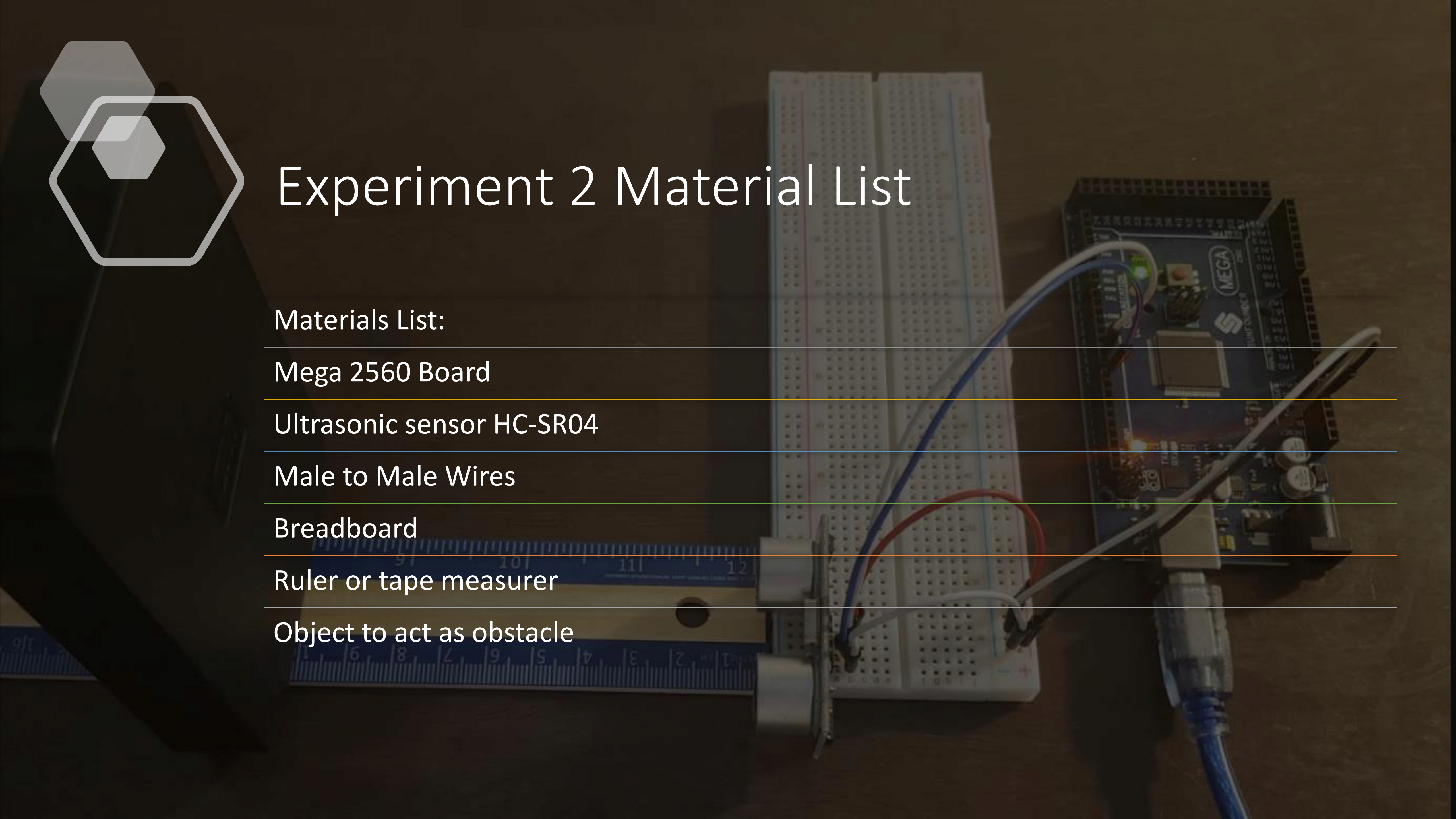
Data Collection

Trial	Ruler Distance (cm)	Total Roundtrip Distance (m)		Roundtrip time (s)	Velocity = distance/time (m/s)
1	4	0.08	189	$1.89x10^{-4}$	423.28
2	6	0.12	328	$3.28x10^{-4}$	365.85
3	8	0.16	424	$4.24x10^{-4}$	377.36
4	10	0.20	550	$5.50x10^{-4}$	363.64
5	12	0.24	636	$6.36x10^{-4}$	377.36



Precision of the Ultrasonic Sensor

Experiment 2 Material List



Materials List:

- Mega 2560 Board
- Ultrasonic sensor HC-SR04
- Male to Male Wires
- Breadboard
- Ruler or tape measurer
- Object to act as obstacle

Data Analysis

- Average velocity from table

$$v_{avg} = \frac{v_1 + v_2 + v_3 + v_4 + v_5}{5} = 381.50$$

- Percent difference where $v_{sound} = 343\ m/s$

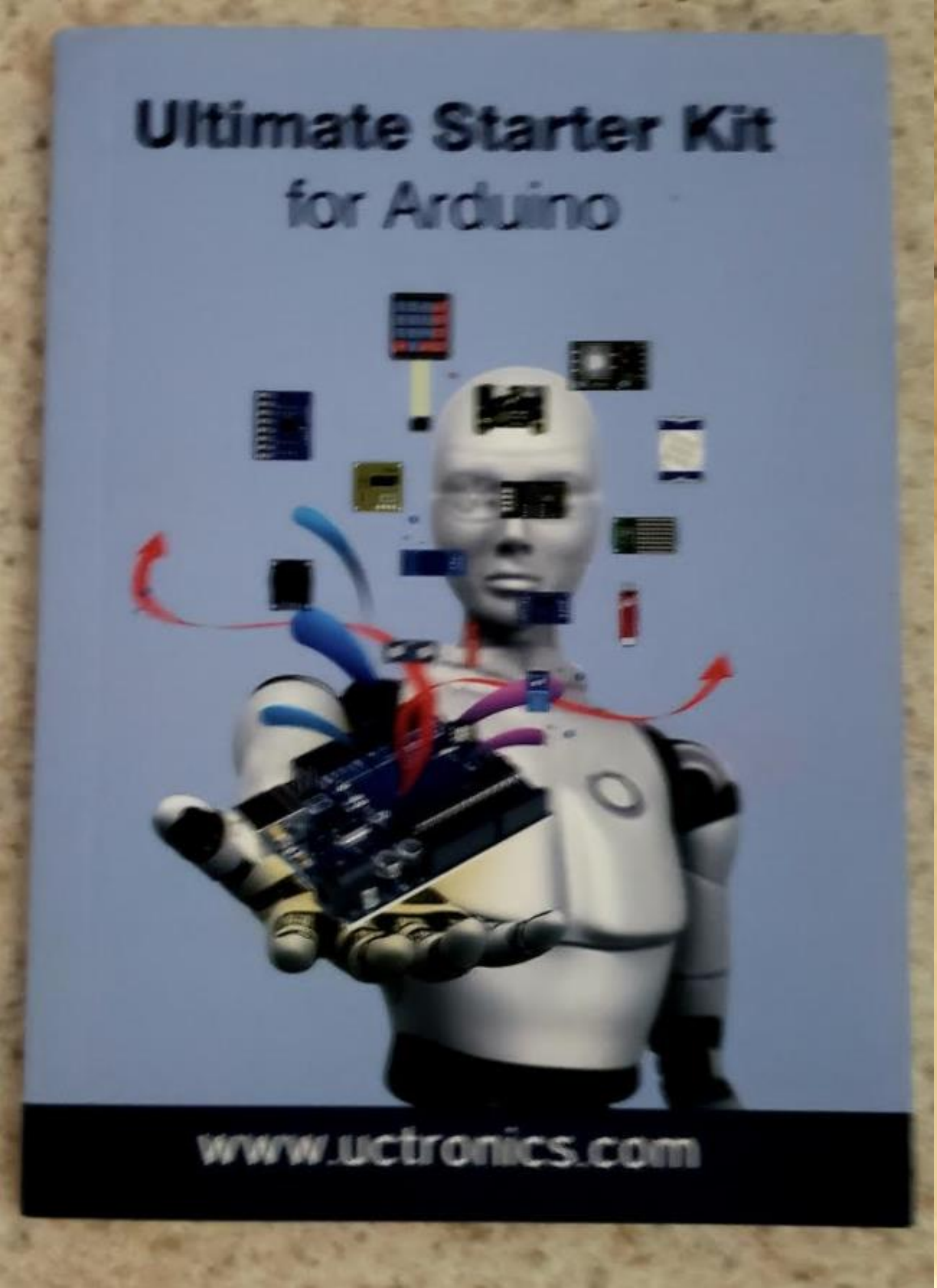
$$Percent\ difference\ = \frac{|v_{avg} - v_{sound}|}{v_{sound}} \times 100\% = 11.22$$

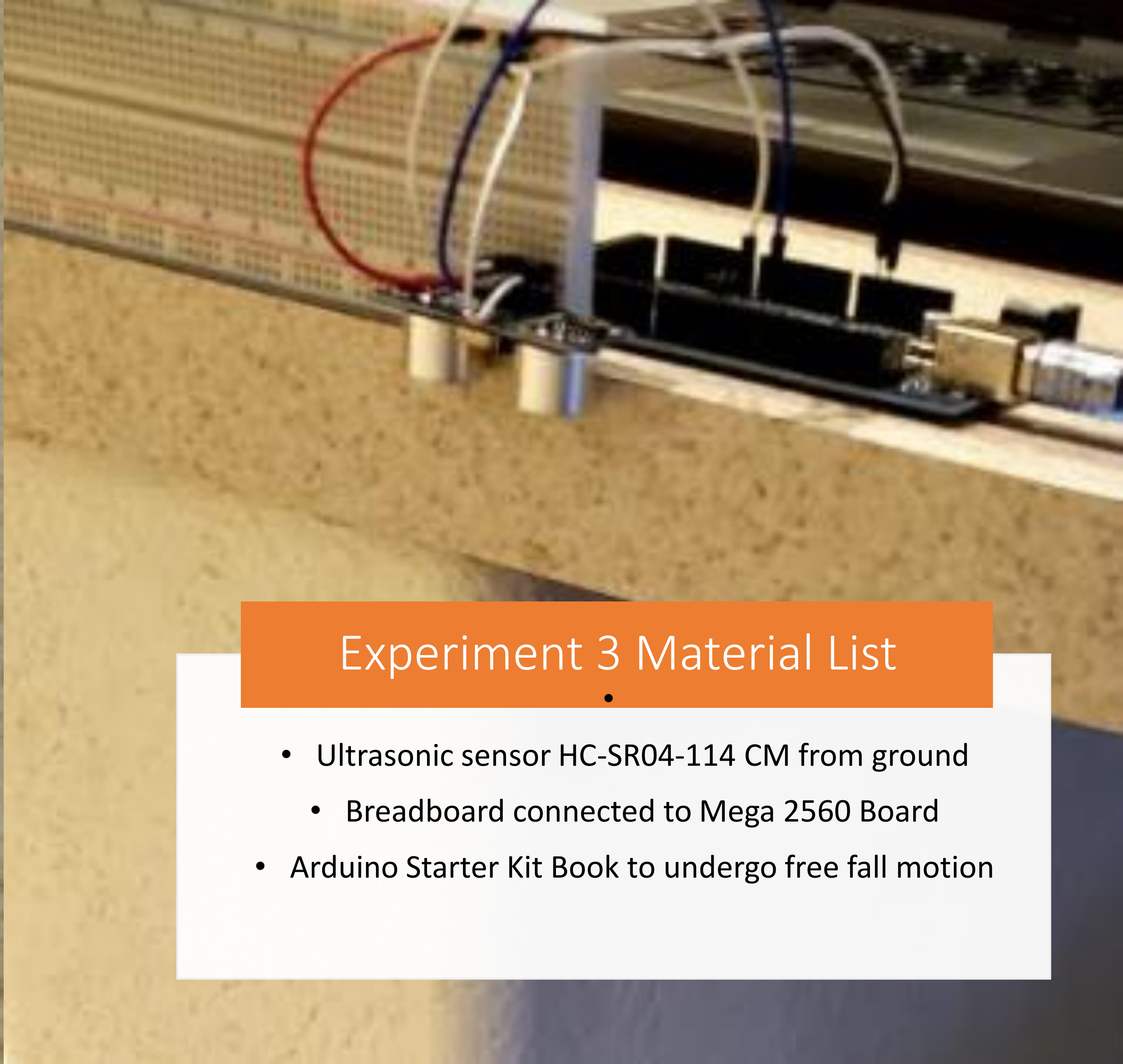
Conclusions – Experiment 2

Serial monitor was not outputting data as I did not have the 5v connected on the Megaboard.

Faulty equipment, bad data processing or human error can lead to inaccurate results.

Gravitational Acceleration





Experiment 3 Material List

- Ultrasonic sensor HC-SR04-114 CM from ground
- Breadboard connected to Mega 2560 Board
- Arduino Starter Kit Book to undergo free fall motion

Arduino Code to Output Data to the serial monitor from the sensor

sketch_sep26a | Arduino 1.8.12

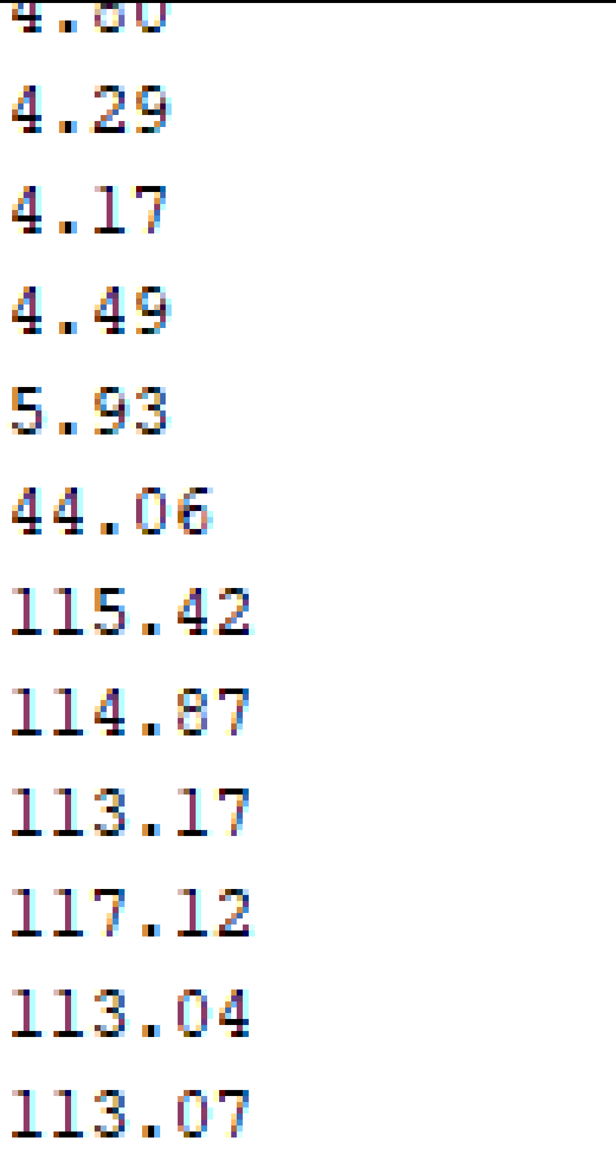
File Edit Sketch Tools Help

sketch_sep26a

```
// Define pins as constant variables that will not change
  const int Trig = 7;   // Use variable Trig (for trigger) for pin 7
  const int Echo = 8;   // Use variable Echo for pin 8

// Define variables for duration of the ping and distance traveled
  long duration;
 double distance;

void setup() {
  Serial.begin(9600);     // Set the serial interface at 9600 bits-per-second
  pinMode(Trig, OUTPUT);  // Configure the trigger (Trig) pin as output. We will use this pin to trigger the Ping sensor
  pinMode(Echo, INPUT);   // Configure the echo (Echo) pin as input. We will use this pin to read the pulse coming back from the ultrasonic sensor
}

void loop() {
  digitalWrite(Trig, LOW);  // Drive the trigger (pin 7) low
  delayMicroseconds(5);     // Keep the trigger pin low for 5 microseconds
  digitalWrite(Trig, HIGH); // Drive the trigger pin high. This will force the Ping sensor to send an ultrasonic burst
  delayMicroseconds(10);     // Wait 10 microseconds
  digitalWrite(Trig, LOW);  // Drive the trigger pin low. This stops the ultrasonic burst

  duration = pulseIn(Echo, HIGH);  // Read the length of the pulse from the ultrasonic sensor in microseconds
  distance= duration*0.0343/2.;     // Calculate one way distance traveled in cm
  Serial.println(distance);         // Display distance on the serial monitor
  delay(100);  // Pause the program for 0.1 seconds
}
```

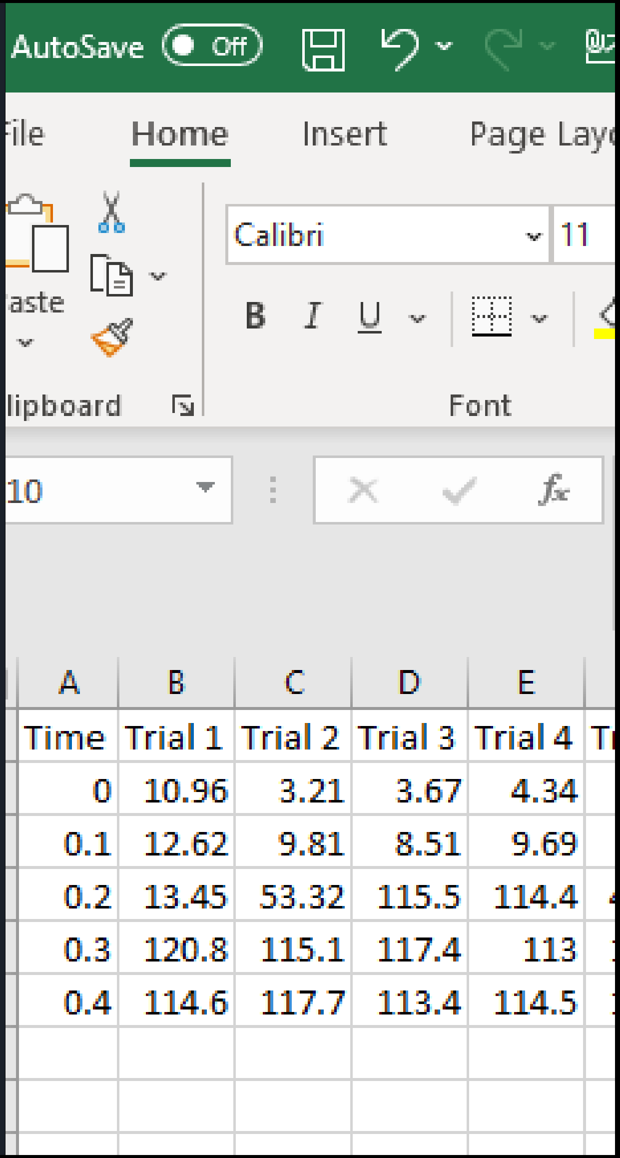
```
Sketch uses 4198 bytes (1%) of program storage space. Maximum is 253952 bytes.
Global variables use 200 bytes (2%) of dynamic memory, leaving 7992 bytes for local variables. Maximum is 8192 bytes.
```

27 Arduino Mega or Mega 2560 on C



Serial Monitor and Data in Excel

- The raw data showing the start of the fall and end of the fall of one trial.
- This data will be used for the next experiment related to the conservation of energy.



A	B	C	D	E
Time	Trial 1	Trial 2	Trial 3	Trial 4
0	10.96	3.21	3.67	4.34
0.1	12.62	9.81	8.51	9.69
0.2	13.45	53.32	115.5	114.4
0.3	120.8	115.1	117.4	113
0.4	114.6	117.7	113.4	114.5

Initial height (m) = 1.14

Data Collection

Trial	Time to hit ground (s)	Acceleration (m/s^2)
1	.4s	-8.06
2	.3s	-1.46
3	.4s	-8
4	.2s	-3.23
5	.4s	-8.06

Data Analysis

- Average acceleration from table

$$a_{avg} = \frac{a_1+a_2+a_3+a_4+a_5}{5} = -5.76$$

- Percent difference with $g = 9.8\ m/s^2$

$$Percent\ difference = \frac{|a_{avg} - g|}{g} \times 100\% = -158.77$$

Conclusions – Experiment 3

THE LINE OF CODE THAT CALCULATES THE DISTANCE IS “DISTANCE=DURATION*.0343/2”, THE LINE BELOW IT, “SERIAL.PRINTLN(DISTANCE);” OUTPUTS THE CALCULATION TO THE SERIAL MONITOR”. THE CONVERSION FROM METERS PER SECOND TO CENTIMETERS PER MICROSECOND IS REPRESENTED BY 0.0343 IN THE CODE. WE DIVIDE BY 2 AS THE ULTRASONIC SENSOR MEASURES THE DISTANCE FROM THE SENSOR TO THE OBJECT AND BACK.

THE RESULTS VARY DUE TO THE OBJECT NOT FALLING UNIFORMLY.

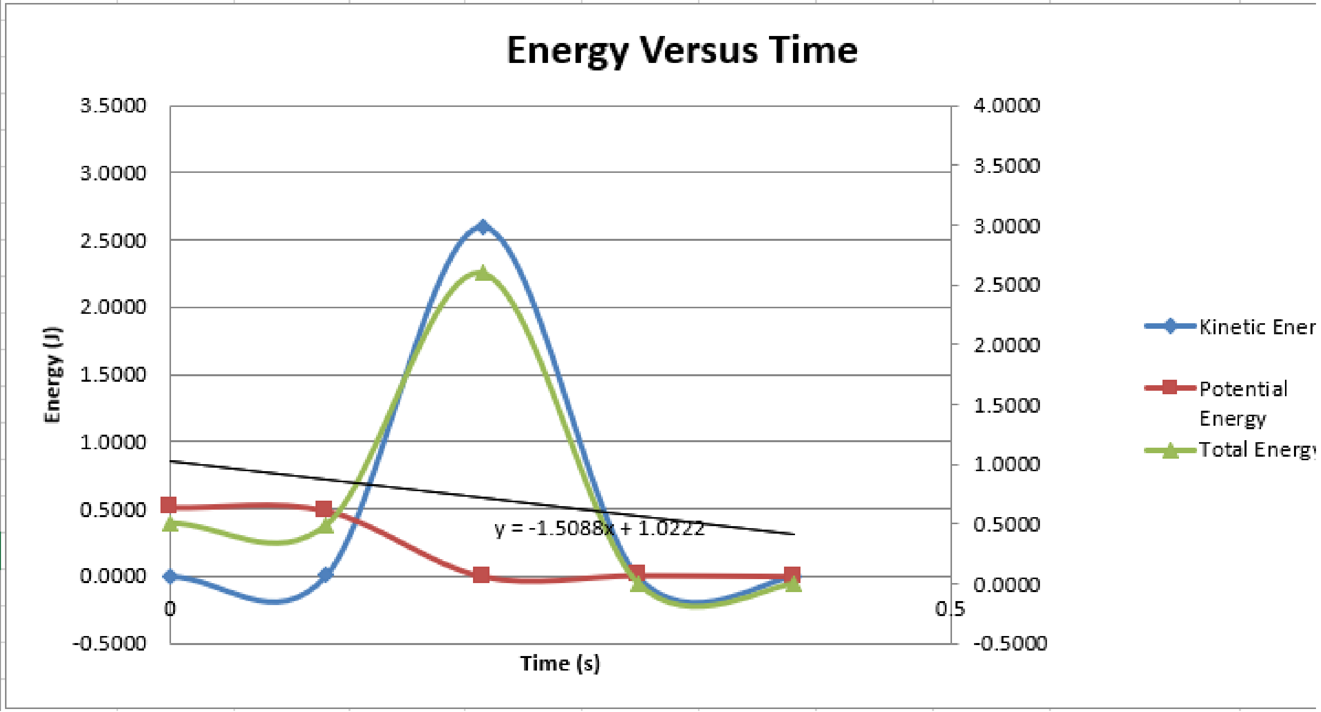
CONSERVATION OF ENERGY

Data Analysis
Trial 2 data used

- By keeping track of energy as it is transformed from one kind to another, we can learn a great deal about the universe.

t (s)	y (cm)	y (m)	h (m)	v (m/s)	K (J)	U (J)	E (J)
0	4.34	0.04	1.10	0.00	0.0000	0.5094	0.5094
0.1	9.69	0.10	1.04	0.53	0.0068	0.4845	0.4913
0.2	114.44	1.14	0.00	10.48	2.6005	-0.0020	2.5985
0.3	113	1.13	0.01	-0.14	0.0005	0.0046	0.0051
0.4	114.53	1.15	-0.01	0.15	0.0006	-0.0025	-0.0019
		0.00	1.14	-11.45	3.1088	0.5296	3.6383
		0.00	1.14	0.00	0.0000	0.5296	0.5296
		0.00	1.14	0.00	0.0000	0.5296	0.5296
		0.00	1.14	0.00	0.0000	0.5296	0.5296
		0.00	1.14	0.00	0.0000	0.5296	0.5296
		0.00	1.14	0.00	0.0000	0.5296	0.5296

initial height (m)	1
mass (kg)	0.0



Data Validation

- Final velocity from experimental data

$v_{experimental}$ = 6.18 m/s

- Theoretical value of final velocity

$v_f = \sqrt{2gh}$ = 4.69 m/s

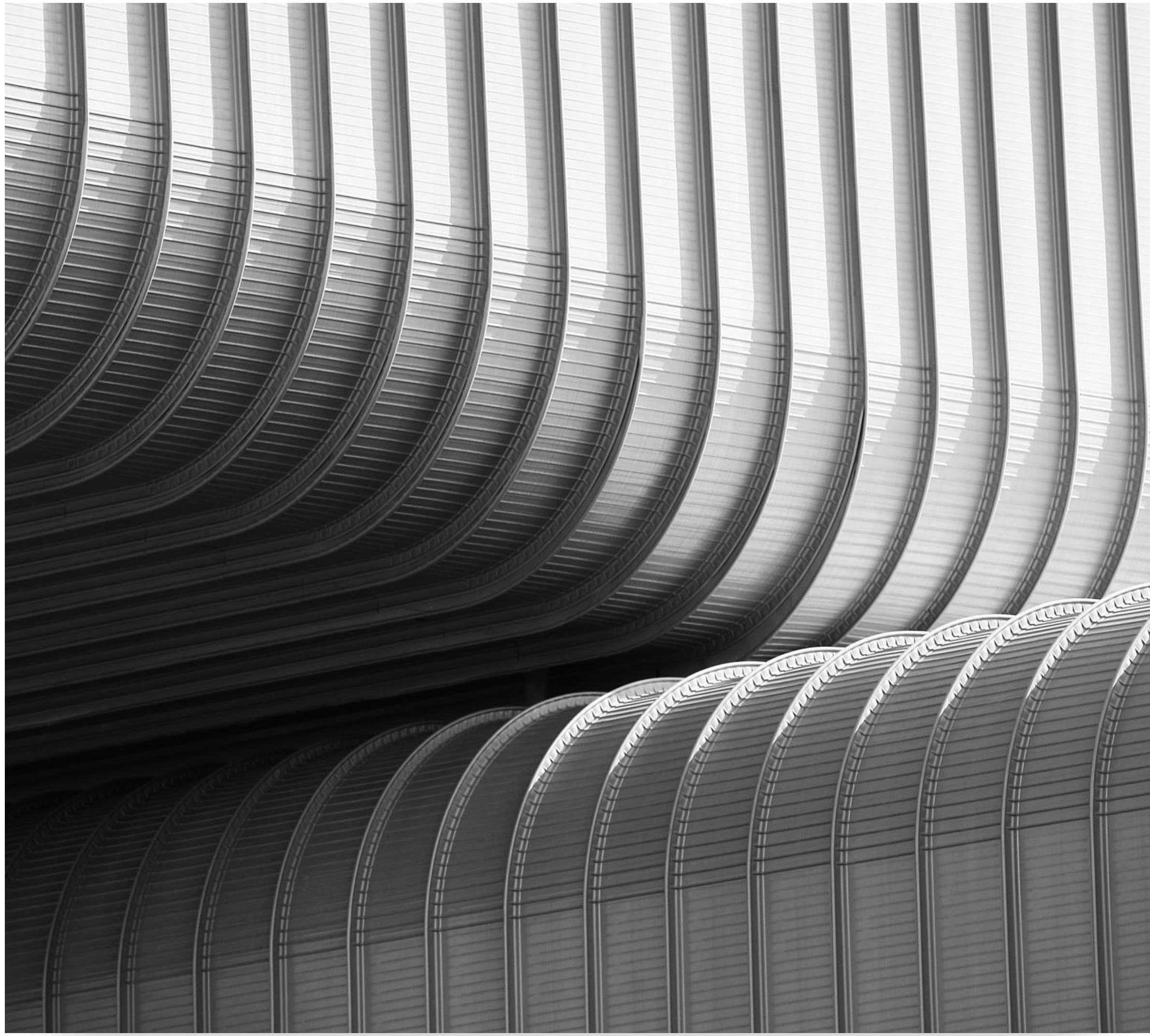
- Percent difference

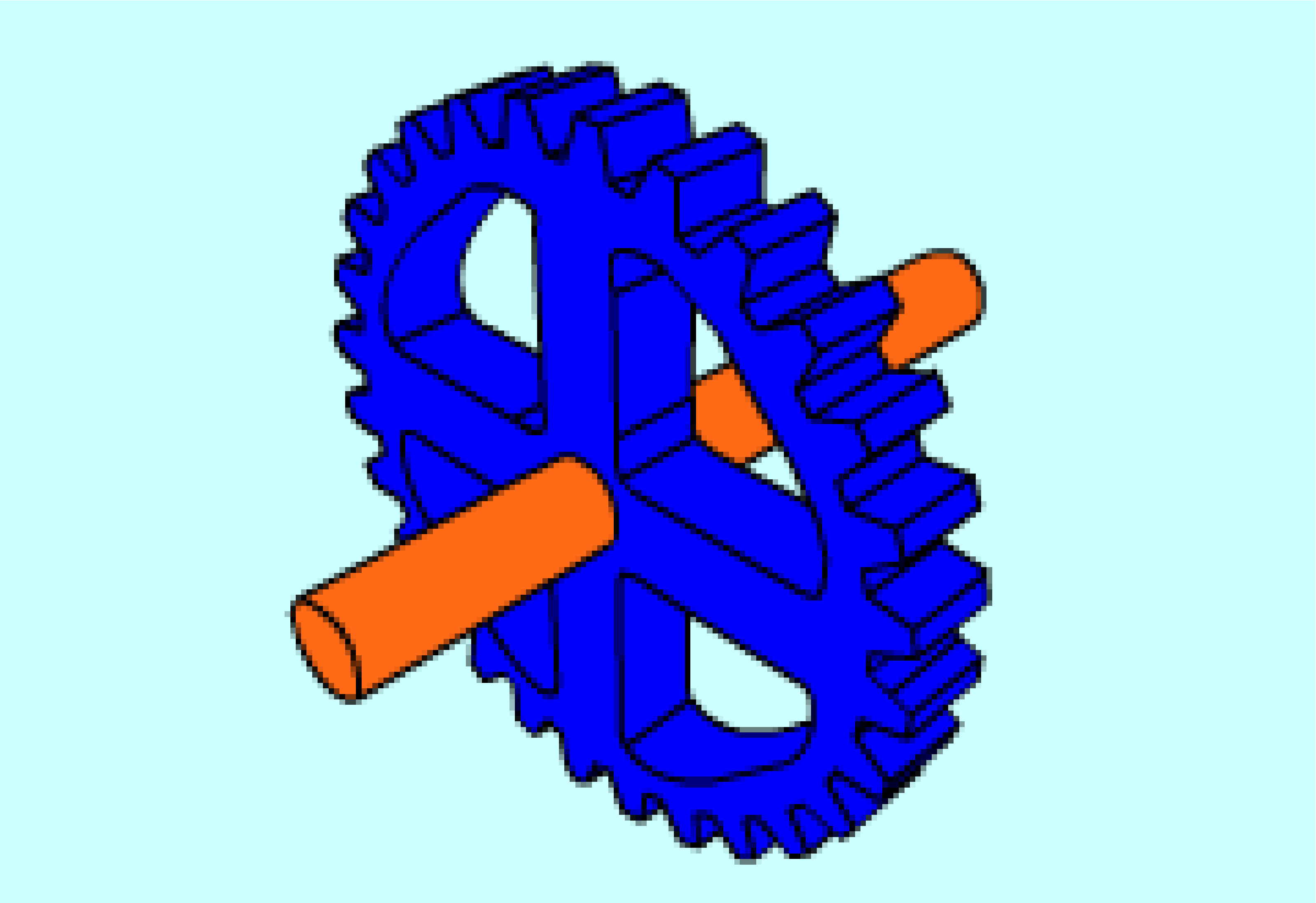
$$Percent\ difference\ = \frac{|v_f - v_{experimental}|}{v_f} \times 100\% = 31.77\%$$

Conclusions – Experiment 4

Mechanical energy is conserved at all time as there are no external forces breaking the conservation.

The kinetic energy starts at 0 and increases as the object free falls while the potential energy goes down as the object falls.





Rotational Motion

In this experiment we apply what we have learned about circular motion to convert radial displacement to linear displacement using the rotary encoder.

Experiment 5 Material List

- Materials List:
  - Mega 2560 Board
  - Rotary Encoder Module KY-040
  - Male to Female Wires
  - Object to attach onto rotary encoder
  - Ruler

Data Collection and Analysis

Object radius, *r* = 2 CM

Trial	Number of pulses N	Encoder distance $r \cdot 0.157 \cdot N$	Measured distance with ruler	Percent difference %
1	39	12.25	10	23
2	58	18.21	15	21
3	78	24.49	20	23
4	112	35.17	25	43
5	81	25.43	20	27

Code in Arduino IDE

linear | Arduino 1.8.12

File Edit Sketch Tools Help

linear

```
const int outputA = 8;
const int outputB = 7;
 int counter = 0;
 int aState;
 int aLastState;
 void setup() {
   pinMode (outputA,INPUT);
   pinMode (outputB,INPUT);

   Serial.begin (9600);
   // Reads the initial state of the outputA
   aLastState = digitalRead(outputA);
 }
 void loop() {
   aState = digitalRead(outputA); // Reads the "current" state of the outputA
   // If the previous and the current state of the outputA are different, that means a Pulse has occured
   if (aState != aLastState){
     // If the outputB state is different to the outputA state, that means the encoder is rotating clockwise
     if (digitalRead(outputB) != aState) {
       counter ++;
     } else {
       counter --;
     }
     Serial.print("Position: ");
     Serial.println(counter);
   }
   aLastState = aState; // Updates the previous state of the outputA with the current state
 }
```

```
Sketch uses 2758 bytes (1%) of program storage space. Maximum is 253952 bytes.
Global variables use 204 bytes (2%) of dynamic memory, leaving 7988 bytes for local variables. Maximum is 8192
```

1 Arduino Mega or Mega 2560 on COM4

Conclusions – Experiment 5

THE DISTANCE PLAYED THE BIGGEST FACTOR IN THE VARIATION OF THE RESULTS. THE FATHER THE DISTANCE THE HIGHER THE POTENTIAL FOR ERROR AROSE. ONE POSSIBLE REASON IS THE OBJECT SLID A BIT THE FATHER IT TRAVELED.

MEASURING THE SPEED OF A CAR ACCELERATING IS A GREAT EXAMPLE OF THIS SCENARIO. DIFFERENT TIRES SIZES CALL FOR DIFFERENCES IN THEIR CALIBRATION.

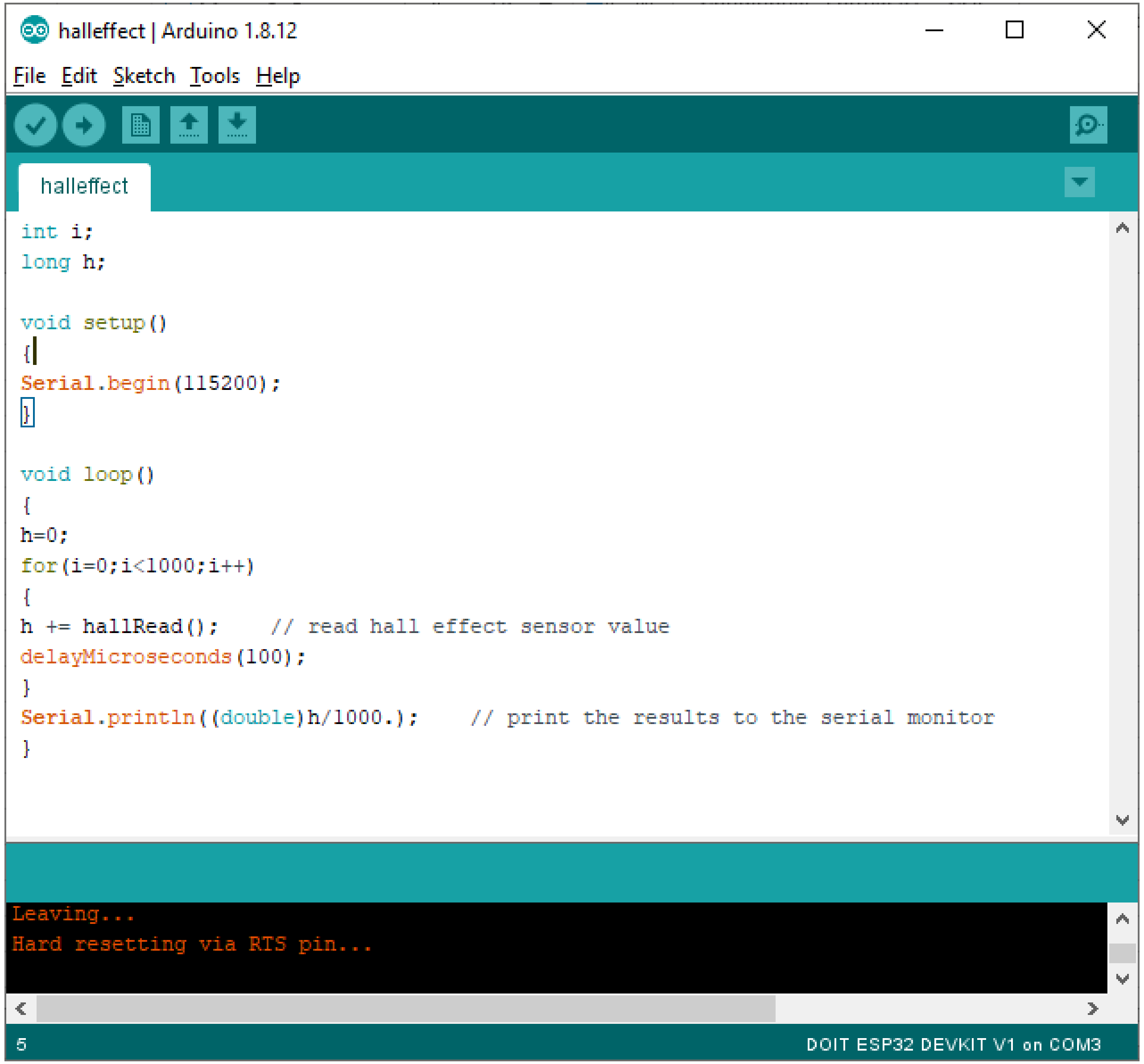
The Hall Effect

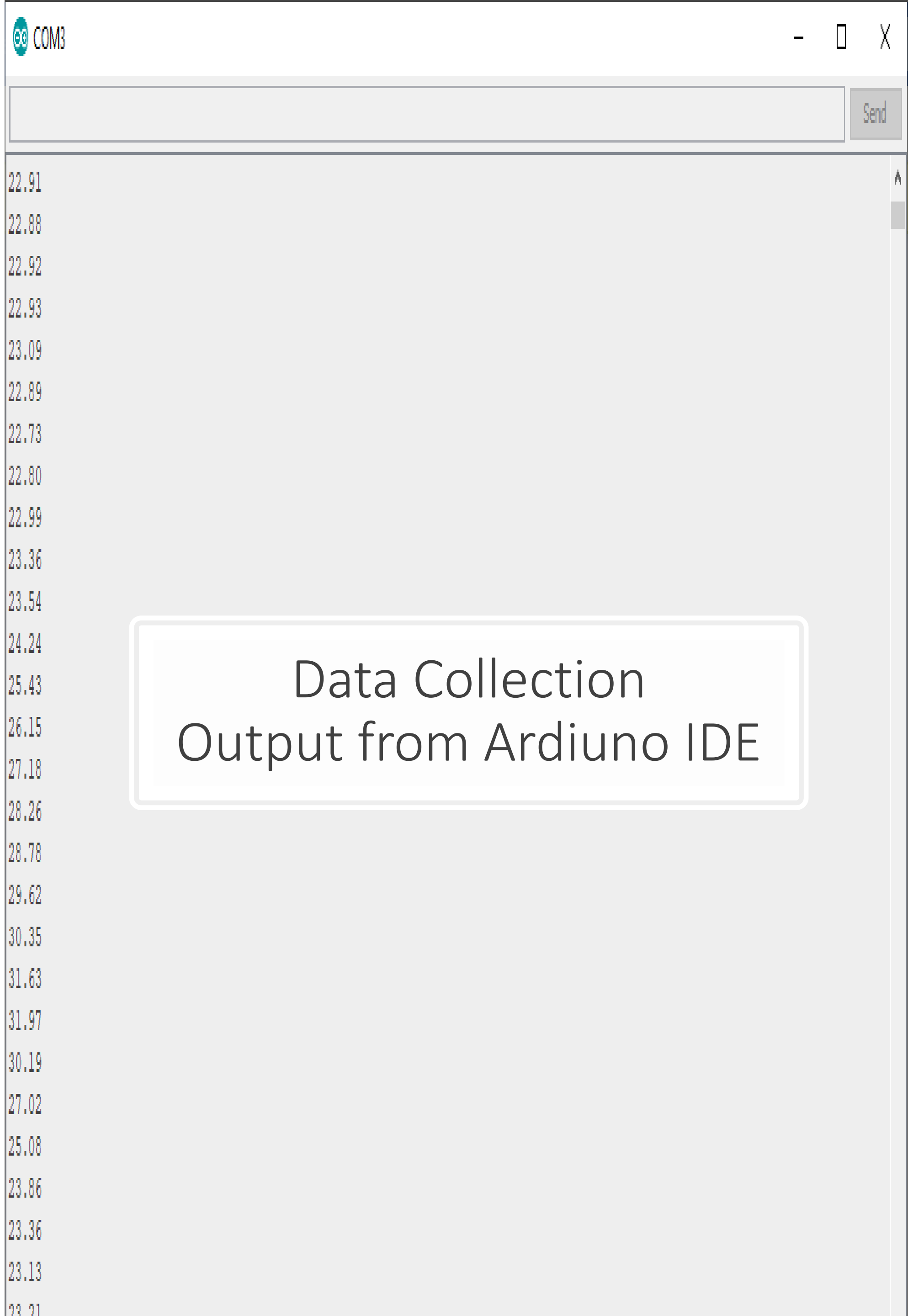
Here we apply wave motion to measure some of the characteristics of waves by investigating frequencies generated from a buzzer using the sound sensor module built into the ESP32 microprocessor

Experiment 6 Material List

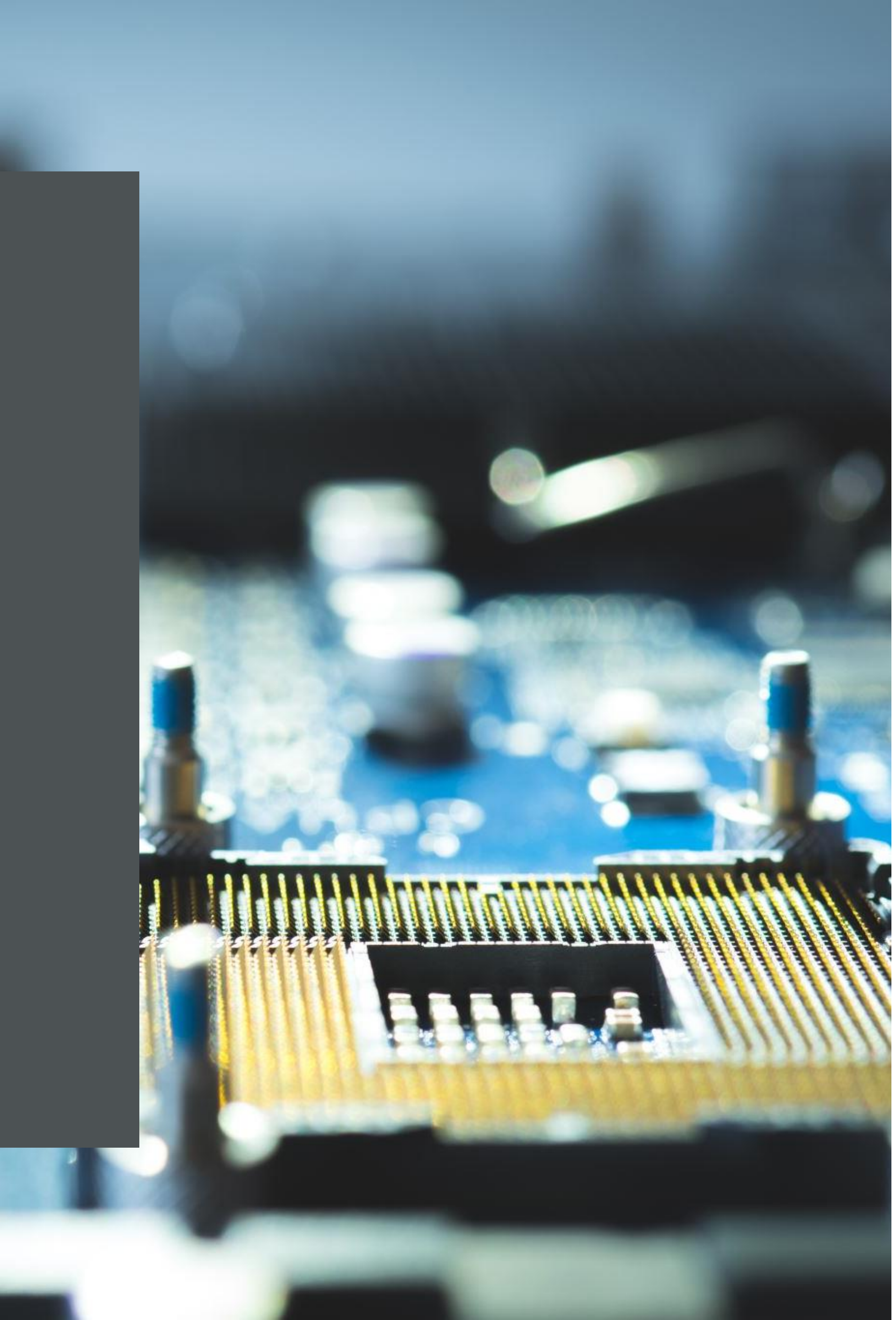
- ESP32 microprocessor
- Micro-USB cable
- Magnet

Code in Arduino IDE





Data Collection
Output from Ardiuno IDE



Conclusions Experiment 6

- Hall effect sensors work by measuring the changing voltage when the device is placed in a magnetic field. Once the sensor detects that it is now in a magnetic field, it is able to sense the position of objects.
- These sensors can be used for monitoring flow rate and valve position for manufacturing, water supply and treatment, and oil and gas process operations. In fluid monitoring applications, analog these sensors are also used to detect diaphragm pressure levels in diaphragm pressure gauges.

Conclusion

- In this course we applied what we have learned about numbers and units to determine the precision of an ultrasonic sensor. The sensor is an electronic device wit transducers that are used to detect an object within its range. The range of the sensor is from 3 CM to 3 M. Determining the precision of the data being gathered from the senor is critical in experiments as they may become inaccurate over time. The inaccurate data may also be from temperature, humidity and component issues.

- We also experimented with motion as a change of position to determine gravitational acceleration of a free-falling object. An analysis of motion helps introduce the real nature of physics-to understand behavior and nature of the physical world.

- Circular motion was also involved in our experiment by converting radial displacement to linear displacement using a rotary encoder.

- Lastly, we calculated wave motion by investigating frequencies generated from a buzzer using the sound sensor module.,