Jayden Brown

Georgia-Pacific Award

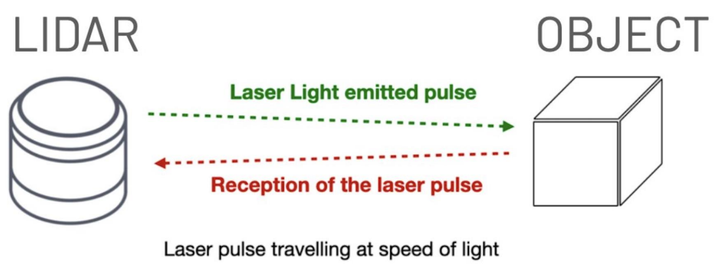
Applied Physics and Computer Engineering Majors

First Year ARCS Scholar

MOREHOUSE COLLEGE

Development and Characterization of a Functional LIDAR Sensor Prototype

This research explores the design and implementation of a low-cost LIDAR sensor prototype within the Morehouse College engineering laboratories. The project integrated laser ranging principles with microcontroller systems to enable accurate spatial mapping and object detection, demonstrating a scalable approach for autonomous navigation applications.

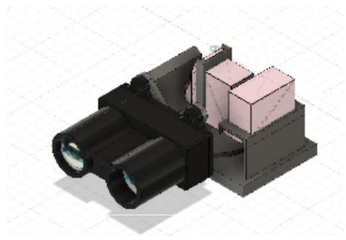


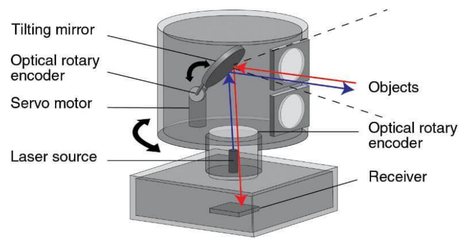
•LIDAR (Light Detection and Ranging) creates precise 3D maps of environments.

•Critical for autonomy, robotics, and augmented reality.

•Commercial sensors are often complex and expensive.

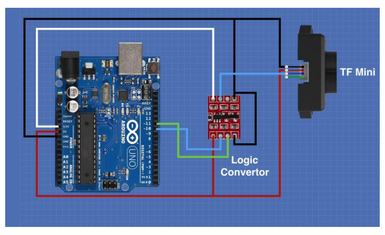
•Objective: To investigate and deconstruct LIDAR technology by designing, building, and testing a functional sensor prototype from basic components.





The scanner assembly utilizes a 2-axis gimbal with a turntable rotation mechanism. This configuration was selected to provide a controlled and stable scanning platform.

- The turntable base enables precise 360-degree horizontal rotation for comprehensive environmental coverage, while the secondary tilt axis allows for vertical angle adjustment.
- This design offers a superior balance of mechanical simplicity, control accuracy, and full hemispherical coverage compared to more complex 3-axis systems, making it ideal for methodical, ground-based spatial mapping.




```
#include <Wire.h>
#include <LIDARLite.h>

LIDARLite lidarLite;
int cal_cnt = 0;

void setup()
{
  Serial.begin(9600); // Initialize serial connection to display distance readings
  lidarLite.begin(0, true); // Set configuration to default and I2C to 400 kHz
  lidarLite.configure(0);   // I2c communication(1 for PWM)
}

void loop(){
  int dist;
  if ( cal_cnt == 0 ) {
    dist = lidarLite.distance();      // With bias correction
  } else {
    dist = lidarLite.distance(false); // Without bias correction
  }

  cal_cnt++;          // Increment reading counter
  cal_cnt = cal_cnt % 100;

  Serial.print(dist); // Display distance
  Serial.println(" cm");
  delay(10);          //Delay for 10ms
}
```

- The objective was to write and upload code to an Arduino microcontroller to serve as the central brain of the LIDAR system. This involved programming the Arduino to perform three core functions
- Hardware Control
- Data Acquisition
- Data Processing & Communication

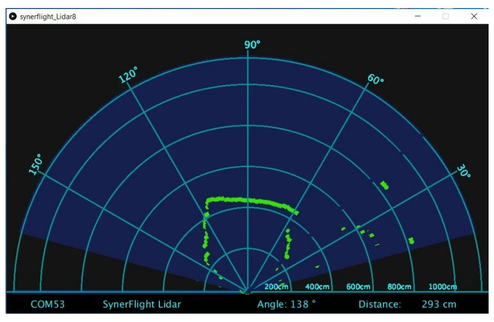
The objective was to write a Python program using the PySerial library to establish a real-time data bridge between the Arduino and a computer. This program performed three critical functions

- Serial Communication
- Data Parsing & Processing
- Real-Time 3D Visualization

```
import processing.serial.*;

Serial myPort;
float rangx, angle_rad, angle_deg;
ArrayList<Float> ranges = new ArrayList<Float>();
ArrayList<Float> angles = new ArrayList<Float>();
float maxRange = 200; // Default max distance (mm), auto-updates
float scaleFactor = 2; // Adjust for better visualization

void setup() {
 size(600, 600);
 smooth();

 // Replace "COMx" with your Arduino's port (e.g., "COM3" or "/dev/ttyUSB0")
 String portName = "COMx";
 myPort = new Serial(this, portName, 115200);
 myPort.bufferUntil('\n');

 background(0);
}
```

```
lude <SoftwareSerial.h>
lude "TFMini.h"

etup software serial port
wareSerial mySerial(10, 11);      // Uno RX (TFMINI TX), Uno TX (TFMINI RX)
ni tfmini;

 setup() {
 Step 1: Initialize hardware serial port (serial debug port)
rial.begin(115200);
 wait for serial port to connect. Needed for native USB port only
ile (!Serial);

rial.println ("Initializing...");

 Step 2: Initialize the data rate for the SoftwareSerial port
Serial.begin(TFMINI_BAUDRATE);

 Step 3: Initialize the TF Mini sensor
ch uses 4520 bytes (14%) of program storage space. Maximum is 32256 bytes.
al variables use 477 bytes (23%) of dynamic memory, leaving 1571 bytes for loca
```

Analysis



Results:

- Successful 3D Environmental Mapping
- Created complete hardware-software pipeline from sensor control to 3D visualization

Future Applications & Development:
Robotics Navigation: Enable autonomous obstacle avoidance and path planning
Next Steps: Miniaturize components, enhance scanning speed

Scholar Awards Celebration

November 12, 2025

SCHOLAR AWARDS CELEBRATION ARCS FOUNDATION ATLANTA

Igniting Innovation in Georgia