

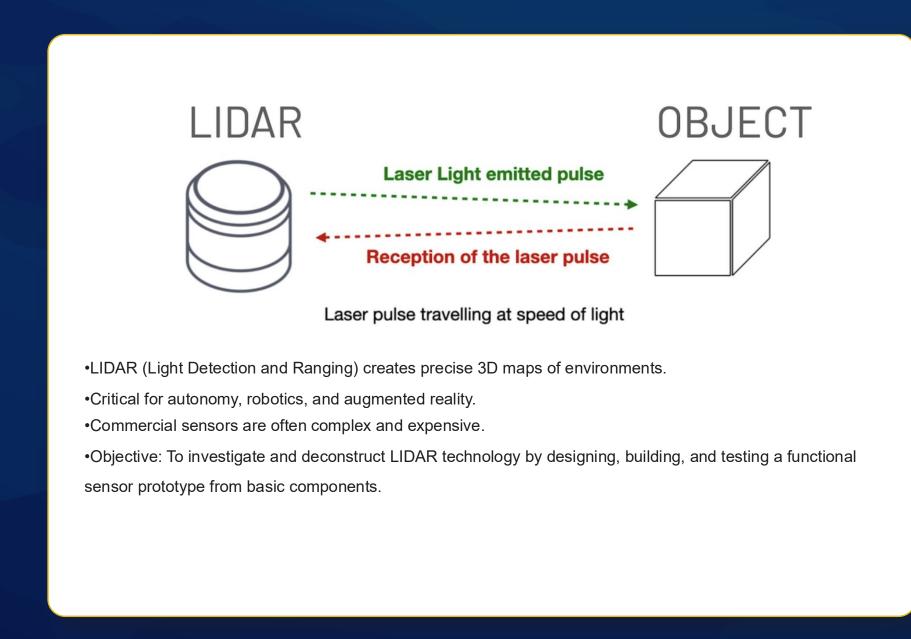
## Jayden Brown

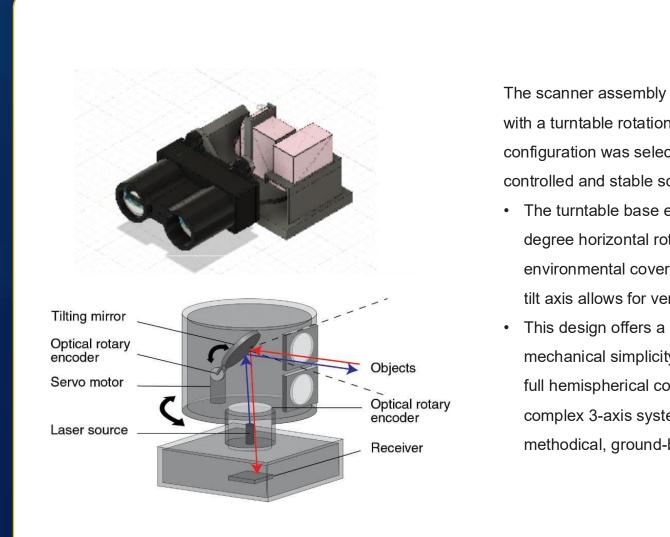
**Georgia-Pacific Award** Applied Physics, Computer Engineering Major First Year ARCS Scholar



## Development and Characterization of a Functional LIDAR Sensor Prototype

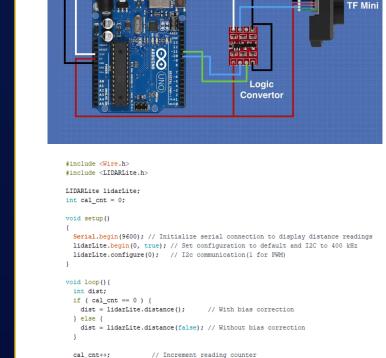
This research explores the design and implementation of a low-cost LIDAR sensor prototype within the Morehouse College engineering laboratories. The project integrated laser ranging principles with microcontroller systems to enable accurate spatial mapping and object detection, demonstrating a scalable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





configuration was selected to provide a controlled and stable scanning platform.

- The turntable base enables precise 360degree 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.



cal\_cnt = cal\_cnt % 100;

Serial.print(dist); // Display distance 

- 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

rial.begin(115200);

- Serial Communication
- Data Parsing & Processing

 Real-Time 3D Visualization float range, angle\_rad, angle\_deg;

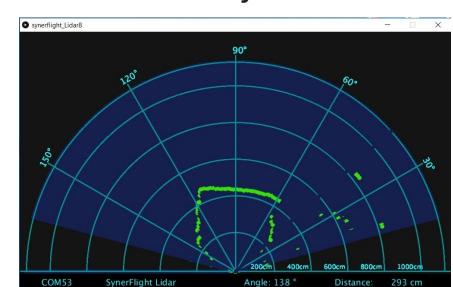
lude <SoftwareSerial.h> lude "TFMini.h" // Uno RX (TFMINI TX), Uno TX (TFMINI RX) ni tfmini; Step 1: Initialize hardware serial port (serial debug port)

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

uses 4520 bytes (14%) of program storage space. Maximum is 32256 bytes. variables use 477 bytes (23%) of dynamic memory, leaving 1571 bytes for local

## 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

