Cameron Oehler
200384643
ENSE 481

Cube Visualizer

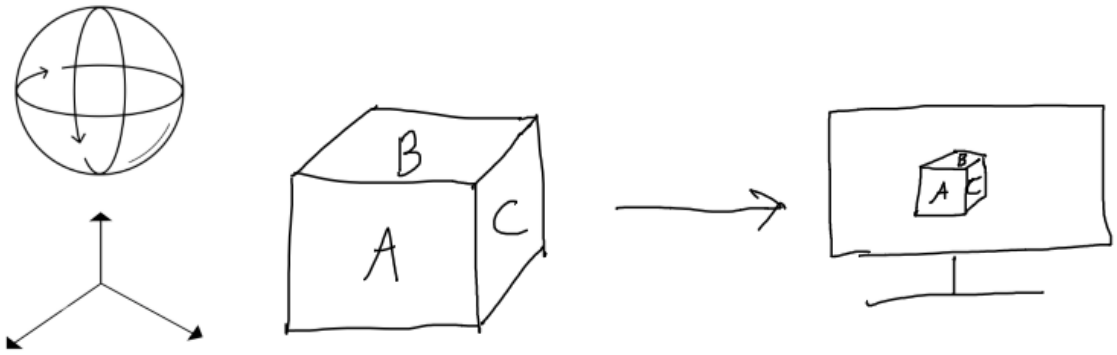
Project Proposal

Introduction

For this project, I will design a visualizer for the rotation and translation of a cube. The cube itself will have sides that are either differentiated by colour or markings. The visualizer will take information sent over a BLE (Bluetooth Low Energy) connection and display the cube in a 3D environment with the same colours or markings, oriented the same as the physical cube.

I will be using an STM32F103RB board to do all the data collection and processing and plan to only send rotation (i, j, k) and translation (x, y, z) data to the visualizer (referenced to location on bootup). Rotation and translation data will be collected using the Adafruit 9-DOF IMU which contains a LSM303DLHC chip (3-axis accelerometer) and a L3GD20 chip (3-axis gyroscope). I may be utilizing the LSM303DLHC's magnetometer capabilities to make the system more accurate. I will also be utilizing the Adafruit Bluefruit LE board which will allow the cube to communicate its rotation and translation back to the visualizer.

I would like for this project (the cube portion) to be battery powered, so I will also be utilizing an Adafruit coin cell battery holder. I also don't want to use a power switch, so I hope to use the STM32 board's low power mode to prevent the battery from draining when the cube is not being interacted with.



Rotation and translation on the cube are replicated on the visualizer

Requirements

- The system shall read rotational acceleration data from the IMU
- The system shall read translational acceleration data from the IMU
- The system shall process acceleration data into (x,y,z) and (i,j,k) values
- The system shall transmit that data over BLE to a receiver
- The receiver shall use that data to display a cube with the matching rotation and translation as the cube

Desirable Goals

- The system should go into low power mode when not being moved for 30 seconds
- The system should wake up from low power mode when moved (a few degrees or a few centimetres)
- The system should transmit values “in sync” with reality (ex. only drift by a few degrees for rotation, only drift by a few units for translation)

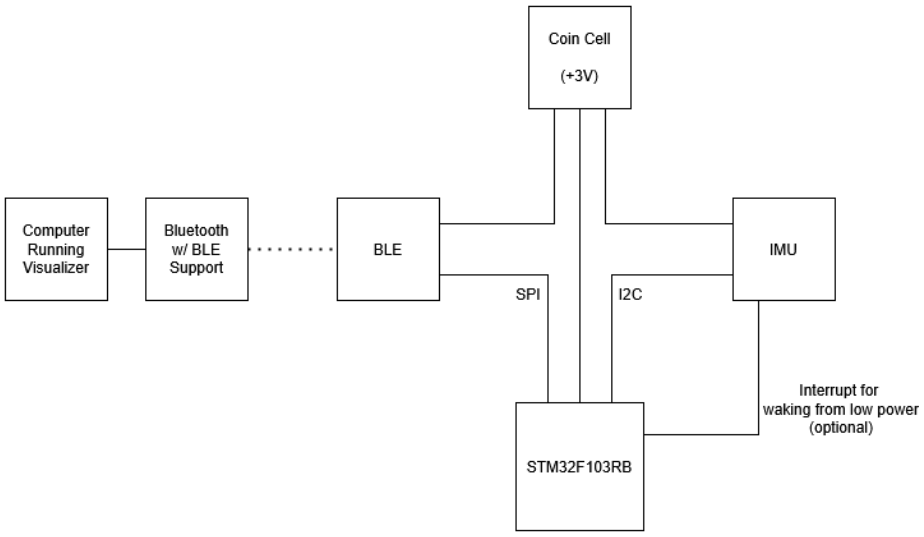
Scenario

1. User pairs visualizer to the cube
2. User moves/rotates cube
3. Cube processes rotational acceleration and translational acceleration
4. Cube transmits data over BLE to visualizer
5. Visualizer receives data and displays it in a Godot based visualizer

Functional Components

- IMU
    - Read values from IMU
- Data processing
    - Convert acceleration values from IMU to (x,y,z) and (i,j,k) values
- BLE
    - Transmit processed values over BLE to receiver
- Visualization
    - Get BLE data
    - Visualize the cube using the received data in the Godot engine
- Low Power
    - Put systems into low power mode when cube is not moved
    - Wake systems when cube is moved (IMU interrupts)

Block Diagram



Design Alternatives

- Could use another engine such as Unity to visualize the data
- Could use cameras to track position and rotation of the cube (to improve tracking)