CC3501 Weekly Report

**Group number:** 2 **Team members:** Ethan Waters, Lachlan Pryce  
**Week number:** 12

**Progress this week**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Who did it?** | **What were the outcomes?** | **What did you learn?** |
| Write LSM9DS1 drivers using previously written MPU9250 driver as a reference. The MPU9250 driver successfully communicated using SPI. | Lachlan | Completed LSM9DS1 driver. |  |
| Test SPI communication between IMU and RP2040 | Lachlan | Magnetometer data was inaccessible. This was found to be due to a design error where both chip selects for accel/gyro and mag were tied to the same GPIO pin. Mag and accel shared the same registry value (0x28), hence only one being accessed with CS was driven low. | Revise datasheets thoroughly to avoid misinterpreting information. |
| Develop arm band for mounting PCB and securing it to user’s arm/body. | Lachlan | Used fusion to create a 40x50mm platform with retention straps. See render below. | Basics of 3D fusion. |
| Implement CAN bus driver for RP2040 and MCP2515 controller | Ethan | Found open-source driver for MCP2515 and RP2040. Read through source code to understand how to implement. Begin testing CAN bus basic code, currently not working. | How CAN Ids can be utilised to prioritise data being sent and trigger different functions / the same function on multiple devices. |
| Write main for RP2040 Master Node | Ethan | Write code to request sensor data over can bus and packages it in a way suitable for transmission to pi via USB and then over socket from the PI. This code will only be utilised if the ESP32 fails. A priority message ID system was developed to indicate required function to be performed (e.g. calibration / what sensor to read from etc) |  |
| Write main for RP2040 node | Ethan | Listens for request over can bus and sends sensor data back to master node. A priority message ID system was developed to indicate required function to be performed (e.g. calibration / what sensor to read from etc) |  |
| Trouble Shoot ESP32-S3 | Ethan | Initial error of constant disconnection reconnection was solved. Was able to upload code, connect to wifi and send data over socket. However then would not reupload code to ESP32 as it was not able to find COM port again. |  |
| Solder wires and resistors to test CAN bus | Ethan | These are makeshift and not suitable for long term use. |  |

**Overall project tracking:** Still working on all subsystems functioning.

|  |  |
| --- | --- |
| **Week number** | **Milestones** |
| 1 | Confirm project topic and begin |
| 2 |  |
| 3 | Arm can move with an input from a socket. The input is an automated test script executed by a client to mimic the embedded system output |
| 4 | Select components & review datasheets |
| 5 | Begin schematic |
| 6 | Complete Schematic, forward to Bronson for feedback. |
| 7 | Submit complete schematic to Bronson for feedback. Complete PCB design for feedback, |
| 8 | Submit board for manufacture. |
| 9 |  |
| LR | Work on vision based movement with PI while waiting for embedded systems. Work on sensor calibration and Kalman filter code. |
|  |  |
| 11 |  |
| 12 |  |
| 13 | All subsystems working to start testing arm |
|  | Demo day Wednesday Swotvac |

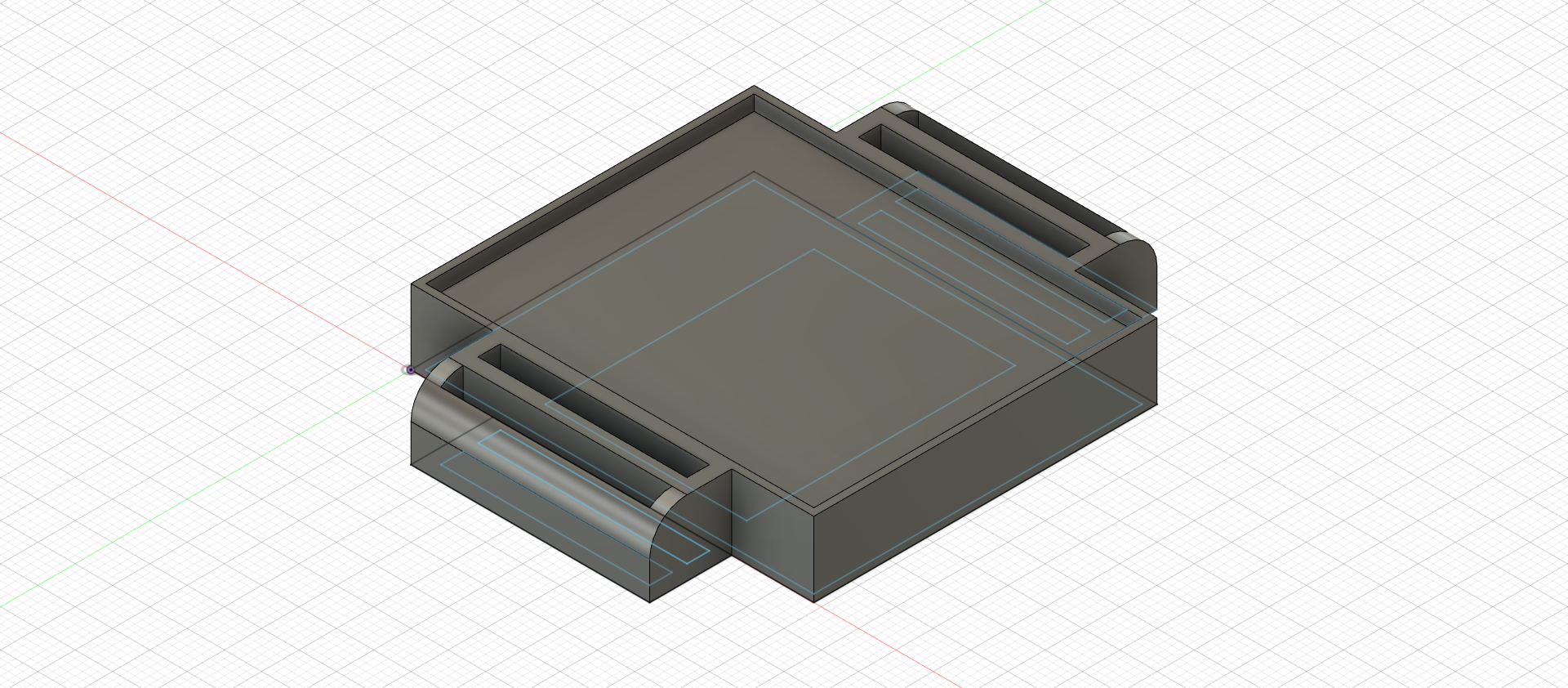


Figure 1: PCB platform