



Montgomery County Community College

Contents

Overview	2
Arduino Code	3
Main Features	3
Operational Problems	3
Future	3
Arduino Shield	4
Main Features	4
Design Decisions	4
Manufacturing Problems	4
First Revision/Eagle	4
Future	4

Overview

The Motor Test Bench (MTB) is a device created to facilitate easier telemetry collection for characterizing quadrotor propeller and motor combinations. The iteration of the MTB this document is applicable to is a major platform transition from the Raspberry Pi to the Arduino Mega. The main reason for the shift was a higher entry barrier to get a working project on the Pi and a large majority of the Raspbian operating system was unleveraged for the application.

Arduino Code

Main Features

Telemetry data is averaged over *NUM_READINGS* readings.

The BLE UART Friend is interfaced with *Serial1* and wrapper functions *ble_tick()* and *sendATCommand()* to abstract the serial connection just a bit.

Each BLE module must be provisioned with *setupBLE()*. **Once.** A single full provisioning can take a few attempts – serial monitoring recommended.

During the loop, all telemetry is read from the external sensors. If BLE is enabled, the new telemetry is pushed to the Android app.

Operational Problems

As mentioned above the BLE provisioning function can sometimes fail. This is believed to be caused by (insufficient time to diagnose further) improper timing with the *Serial1* abstraction. Effects are inconsistent and range from no issues for multiple runs to a full Arduino reboot (**not** a reset).

Uploading while the ESC is connected to the Arduino will cause the ESC to spin-up at maximum. Presumed cause is Arduino's internal PWM timers are reset during upload.

Future

Improve *Serial1* abstraction. CTS was grounded to allow bidirectional communication. Utilizing these pins may also provide a solution to the *Serial1* miscommunications.

Using the BLE module that communicated to the Arduino over a different protocol (I2C, SPI)

Arduino Shield

Main Features

Connection between Arduino and external components

Easy access to probe points and insertion points for passives (resistors, etc.)

Design Decisions

Single sided PCB

Voltage dividers for ADC inputs (voltage & force)

Manufacturing Problems

Keep pin population of the shield to a minimum or else it'll be impossible to separate the Arduino and the shield

First Revision/Eagle

Transition from Fritzing schematic/layout to Eagle

Easier definition of custom parts, footprints, and symbols

Easier handling of routing and double sided

Just because it's professional

Future

Further optimization, i.e denser component placement

Further double-sided PCB leverage