EXECUTION REVIEW - AVIONICS

Motor Test November 6, 2023

STATUS: CATO

I. INTRODUCTION

The Motor Test Project undertaken by team ThrustMIT 2023-24 aims to test and analyze the working and response of a Student Researched and Developed (SRAD) M-Class Solid Rocket Motor. This document will review the performance of the Avionics Subsystems for the Motor Test executed on 6th November 2023.

II. BACKGROUND

The Motor was situated on a testbed, where it was placed upon a pancake Load Cell. Additionally, there was a Strain Gauge module that was attached to the nozzle. The Motor was fired via an ignitor device. For the purpose of operations and reference, we shall divide the Avionics Systems into two: *Motor* and *Ground*. The microcontroller used for both systems is a Teensy 4.1, and the RF Module required for wireless communication between the two systems is an XBee S2C.

The Ground System employs a switch-controlled Ignition System with three states, SAFE, ARMED, and LAUNCHED. These are coordinated between the two systems via a Finite State Machine on both sides to ensure safe transition between phases. The Motor Side also employs logging of peripheral data in the built-in SD Card via the peripheral modules.

The Strain Gauge Module (BF350) was powered via a Power Distribution Board (PDB), providing approx. 5V voltage to the module, converting the 7.4V it receives from a LiPo Battery. The Load Cell was powered by an Arduino Uno R3 to provide stable 5V input; Both the Arduino and Teensy on Motor Side are powered via a Power Bank through respective USB Cables.

Additionally, a live video telemetry system was employed via a Runcam Split 3 Lite and associated transceiver module.

Module	Motor Side	Ground Side
Teensy 4.1	✓	✓
XBee S2C	✓	✓
RS232/TTL Converter	✓	✗
BF350 Strain Gauge	✓	✗
Built-in SD Card	✓	✗
Load Cell Indicator	✓	✗
Ignition Control	✗	✓
TPT Color Monitor	✗	✓
RD945 Transceiver	✗	✓
RunCam Split 3 Lite	✓	✗

a. Checklist for Modules on both systems

III. CONCEPT OF OPERATIONS

The execution of the project is divided into four phases: Final Assembly, Initial Arming, Final Arming and Launch. Each phase involves the integration of implementing practical safeguards, ensuring reliability, and finalizing safe and secure ignition.

PRE-LAUNCH: Both Systems apparatuses, required tools and backup materials are required to be packed safely. LiPos shall be kept in LiPo Guards when not in use. The Materials Checklist must be referenced the morning of the test i.e., by 10AM. All Materials shall be brought to the test site immediately post-securing both systems and materials checklist, after coordinating with the other involved subsystems.

Upon reaching the test site, the testbed must be confirmed to be secure/stable for arming. Power Sources for the Load Cell Indicator and Video Telemetry Module must be secured. <u>All Connections</u> (Jumpers & Terminal Blocks) <u>must be rechecked upon reaching the test site</u>. The working of Peripherals (PDB and Load Cell) must be confirmed by providing LiPo power. Test codes may be run on the motor side to ensure both modules are providing data. SD Card datalogging must be tested <u>at least once</u>.

<u>Ranging</u> is necessary to ensure smooth working of ignition system during final arming and launch phases. This involves broadening the range of the RF Modules by testing transmission and reception at regularly increasing intervals. The Motor Side's RF System will carry a Dipole Antenna, and Ground system via a Yagi. Ranging shall begin at 2-5m, with ground apparatus moving back in Line of Sight (LoS) in intervals of 5-10m (depending upon increasing reliability of modules). Once a range of 150m has been secured, range can be confirmed, and may be checked up to 200m. Additionally, Range shall be confirmed from the motor's end through the testbed cage. If Interference is found, the RF Module must be placed outside the cage. Finally, D4184 Latching must be confirmed at the maximum distance. Both Systems must be reset after ranging.

The Transmission testing sequence is as follows: KEY shall remain switched on at ground side, and ARM_ON and ARM_OFF sequences must be sent at each increasing distance for testing. Reception of data may be checked either via Serial Monitor (if using Laptops) or via the associated LED.

FINAL ARMING & LAUNCH: Once Prelaunch Operations have been carried out successfully, the Motor Side's connections shall be confirmed. Load connections shall be made at motor side from the D4184 terminal block and confirmed via DMM. There should be no continuity between the Load Power terminal and load pinouts. The key shall be placed in the key switch. Video Telemetry shall begin and a live feed of motor side with <u>clear view of the LEDs</u> must be confirmed. Upon confirmation, the system shall be armed. Once safety provisions are confirmed and everyone is confirmed to be a safe distance from the testbed, <u>System is ready to Launch</u>.

IV. PHASE TRANSITIONS

The phase transitions are initiated via the switches on the Ground system and handled via the FSM installed onto the Teensy 4.1 Microcontroller. The FSM defines three states involved on either side: SAFE, ARMED, and LAUNCHED. The motor side includes an additional FAILURE state provided to add fail-safes or additional safety provisional programs. These states are structured as such:

SAFE: At Ground side, the KEY is inserted into the key switch, allowing power from the LiPo to reach the Peripherals. This operation is purely mechanical and does not initiate any transition or hold any connection to the microcontroller. At both Motor side and Ground side, the key-LED will be shown as ON, showing that the Motor Side is in SAFE state. In this state, no peripherals are called i.e., datalogging does not occur, and the D4184 firing pin is pulled LOW to ensure load remains unpowered in this state.

ARMED: The ARM switch is switched ON at the ground side, sending a transmission string “ARMON” to the motor side. These cause a state transition at the motor end from SAFE to ARMED. Both sides should now show both YELLOW and GREEN LEDs as ON. This shall be confirmed via the Live Video Feed. The system being *armed* is defined as it having only one control operation separating it from ignition. In this case, that is the LAUNCH switch at the Ground system.

LAUNCHED: The LAUNCH switch is turned on, causing a transmission string “LAUNCH” to reach the motor side, causing the D4184 to ‘latch’ or pull HIGH, completing the load circuit and allowing power to reach the load connection. The Ground system will enter an infinite loop, where it will log data from the Strain Gauge module and Load Cell in intervals of approximately 100ms (processing time between each loop). The data is logged onto the SD Card as a CSV file.

```
case LAUNCHED:
  digitalWrite(key, HIGH);
  digitalWrite(arm, HIGH);
  digitalWrite(launch, HIGH);
  digitalWrite(D1,  HIGH);    //D4184 is pulled HIGH
  while (true) {
    Peripherals();
  }
  break;
```

b. Code Snippet of the D4184 ‘Latching’ operation

V. PERFORMANCE

For the purpose of comprehensive analysis and review, the execution and performance review of the Avionics will be done in two segments: Telemetry and Datalogging. The final assembly and initial arming were conducted successfully according to the Guidelines and Documentation prepared prior to the test.

IGNITION SEQUENCE (STATUS: SUCCESSFUL)

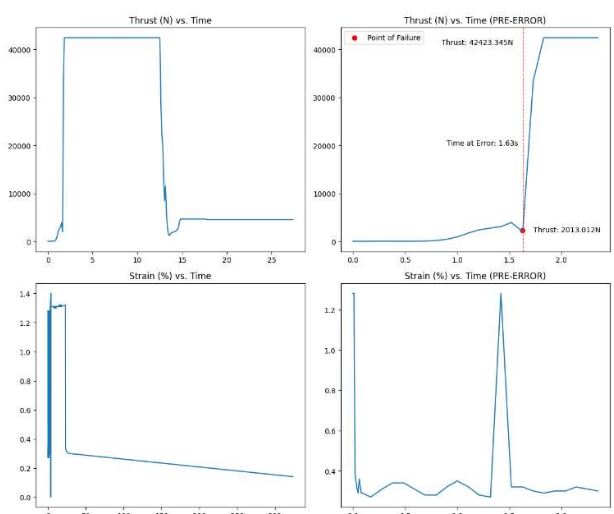
through the initial arming process, both ARMING and D4184 Latching were confirmed prior to final arming and launch. The ignition sequence was carried out via ground station in proper order with all safety provisions in place and was Successful in implementation without any error in transmission, or unexpected transmission delays.

DATALOGGING (STATUS: SUCCESSFUL)

The datalogging process was successfully implemented post-launch i.e., once motor system received launch-signal transmission. The data was recorded onto the built-in SD Card in a file titled ‘dataLog.csv’, in CSV file format. The data was recorded for a total interval of 27.32s.

VI. POINTS OF FAILURE

The primary point of failure was the **Load Cell**. The data recorded from the pancake load cell was of the format (+/-)XXXXXX.X, where X represents digits received from the serial comms. Post-Data analysis, it is apparent that the load cell output failed at a timestamp of 1.63s. The point of indication was identified in post-processing, upon generating a thrust-curve based on the raw-data provided. From 1.63 to 1.73s, the thrust value jumps from a value of 2013.012N to 33357. 924N, after which it remains fixed (assumedly due to some internal failure) at a value of 42423.345N for a time of 10.68s. Additionally, the test itself resulted in Catastrophic Failure (CATO), causing a destabilisation in the testbed and irreparable damage to the load cell itself. The time of damage cannot be identified from the raw data received. It is assumed that any useful data recorded pre-failure was lost due to error of the load cell.



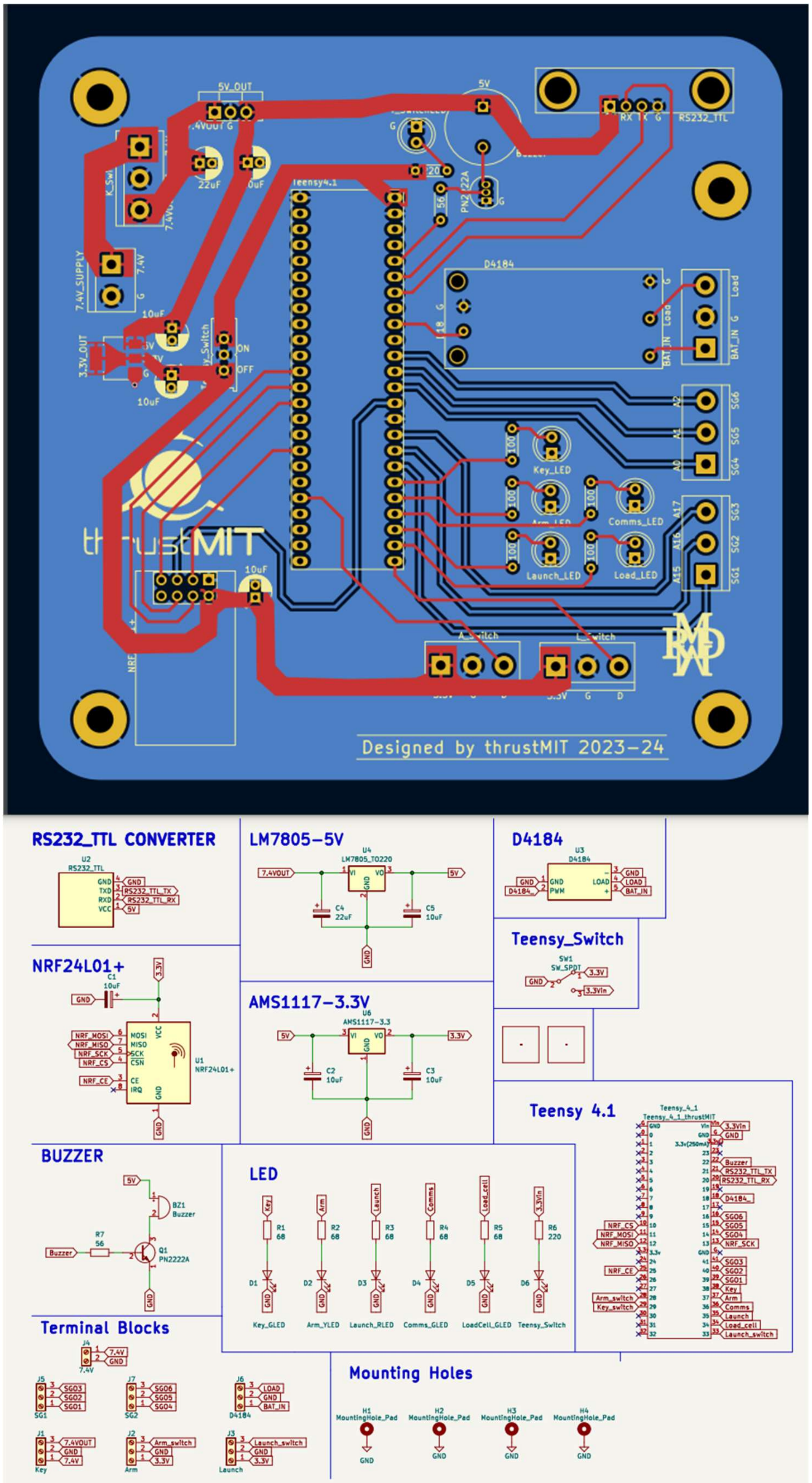
c. post-processed data graphics

VII. PREPARATION REVIEW

1. *RF MODULE*:
   The original RF Module designed for both systems was an nRF24L01+ (PA/LNA). The module itself had been tested extensively since August. However, 2 weeks prior to the test, the nRF devices failed to connect with the Teensy 4.1 microcontroller, failing to initialise. The potential causes of error and their status are listed as such:
   a. *nRF INTERNAL FAILURE*
      STATUS: Not cause of error.
      REASONING: works with Arduino Uno R3, tested 5+ nRFs.
   b. *TEENSY 4.1 INTERNAL FAILURE*
      STATUS: Not Cause of error.
      REASONING: tested with other peripherals.
   c. *SPI COMMUNICATION*
      STATUS: Not cause of error.
      REASONING: SD Card works successfully on Teensy 4.1, also requires SPI Support.

2. *DATA TELEMETRY:*
   The original purpose for transceiver modules was for data to be sent via RF communication as well as be datalogged on either side. The XBee S2C was unable to perform duplex RF communication. The possible reasons for this are a slower data rate of 250kbps, issues in configuration, etc. Reason for failure is still being investigated.

3. *PCB BOARD* (MMRD):
   the PCB board designed for motor test (titled *MMRD*) had a common design and fabrication for both *motor* and *ground* systems. The PCB could not be utilised as per its intended manner, due to several defects found during the preparation phase:
   a. Post-fabrication, an issue was found where the <u>supply voltage</u> to the Teensy 4.1 microcontroller was setup as 3.3V, which is outside of the 3.6-5.5V *Vin* (Input voltage) range. This resulted in the intended pin-headers being unsafe to use due to the Vin power fill.
      Alternative: Jumpers were used to connect the microcontroller to its peripherals, and a power-bank was used to power the Teensy 4.1 via the USB port.
   b. Similar to the issue faced by Teensy 4.1, the nRF24L01+ (intended as per board) was not getting powered by the 3.3V supply provided by AMS1117 Voltage Regulator.
      Alternative: power was supplied to the nRF directly from an Arduino Uno. Similarly, the finalised XBee S2C module used was powered via Teensy 4.1s 3.3V output supply as well. Reason for failure is still being investigated.
   c. The buzzer-transistor circuit that was intended as an audio indicator, was buzzing untriggered upon receiving 5V voltage supply. The *solution* reached upon was to eliminate the buzzer from the circuit altogether. The buzzer-transistor circuit itself works nominally outside the board as a breadboard setup.
   d. A '*Teensy switch*' was designed to control the voltage being supplied to the Teensy 4.1 but was defective due to the wrong slide-switch pin receiving 3.3V power supply.



d. visuals for MMRD board design

VIII. ADDITIONAL REMARKS

Following key-points were drawn as a response to problems faced during the design-process of the avionics systems. They may be taken as reference for similar future projects.

Power Supply for components was not proper - proper analysis of power consumption with reference to voltage and current supply must be done comprehensively prior to including such circuits in a PCB Design.

Testing for the same component/circuit was being done repeatedly – Proper documentation must be done for every test carried out to ensure time and resources are not wasted in the testing phase.

Multiple resources were wasted during testing and design – maintenance must be done properly for all relevant components, regardless of their price or availability. This includes having proper testing procedures to ensure no component/device is falsely labelled as faulty.

Communication devices failed at the end moment – redundant systems must be put in place for all components so that last moment malfunctions do not lead to complete system failure.