HART Avionics

Release 0.0.0

OSU HART

Nov 13, 2020

CONTENTS

CHAPTER

ONE

INTRODUCTION

This is the documentation repo for the Oregon State Univerity AIAA High Altitude Rocket Team's ECE & CS avionics sub-team. This covers any system-wide and user-level documentation for the project. For more detailed documentation on project internals and developer documentation, see the corresponding project's repository.

CHAPTER

TWO

PROJECT OVERVIEW

2.1 Goals

The HART team's ultimate goal is to launch a rocket up to 150,000 feet above ground level. Any change to the system must be done with this goal in mind. As part of the Avionics-EECS subteam, our goal is to work on the rocket's avionics systems as well as any supporting systems that move us closer to our ultimate goal.

2.2 Basic Requirements

The most basic requirements of a rocket avionics system are *triggering events* and *tracking the rocket*. Tracking the rocket will be how we determine if the rocket reached its goal altitude or not. There are several ways in which this step could go wrong or produce incorrect results, so redundancy and rigorous testing should be preferred. In addition to this, the rocket's avionics system must be able to trigger events under the right conditions in order to maximize the chances of reaching higher altitudes and, more importantly, the chances of successfully recovering the rocket.

2.3 Sequence of Events

The rocket avionics system requirements are structured around the following sequence of events:

2.3.1 Pre-Launch

- Hardware Safety Check
- Configure Flight Software
- Software System Check
- Calibration

2.3.2 Launch

- Send launch command
- Ignite booster
- Detect launch

2.3.3 In Flight

- Record raw sensor data
- Record flight data
- Record rocket vitals (engine temperature/pressure)
- Record video
- Estimate current state (position/velocity/etc.)

2.3.4 Stage Separation

- Detect booster depletion
- Ignite separation charges

2.3.5 Upper-Stage Ignition

- Delay to save fuel and increase peak altitude
- Ignite sustainer

2.3.6 Parachute Deployment

- Detect apogee
- Trigger chute deployment charges

2.3.7 Recovery

- Transmit signal for locating the rocket once it lands
- Download flight data for further analysis

2.3.8 Analysis

- Run more computationally intensive algorithms on the recorded raw sensor data to determine if the flight computer is functioning properly.
- Replay the flight in greater detail
- Use data analysis techniques to create a model of the flight
- Compare flight model to predicted/hypothesized model in order to inform future decisions & improvements

2.4 Project Structure

2.4.1 Launch System

Control Box

The enclosure with the big red button

Launch Box

Ignites the booster when launch signal received from Control Box

2.4.2 Avionics

Flight Computer

The hardware & software that triggers the rocket events

- TeleMegas & EasyMegas running flight software on AltOS

2.4.3 Payload

Student-Developed Avionics

Eventually replace commercially-developed flight computer

Tracking Beacon

A way to track the rocket for recovery in case the flight computer's telemetry fails

Camera

Record video through a window in the side of the rocket

Rocket Vitals

Monitor rocket internals, mostly for post-flight debugging

- Examples include motor temperature & pressure, stage-separation detection, parachute deployment detection, and dedicated state estimation

2.4.4 Ground Station

Transceiver

Communicates with the Avionics

- Antennas
- TeleDongle

Ground Computer

- Configures flight software
  - AltOS
- Saves the telemetry for later use, processes the telemetry, and serves the processed data to the GUI Frontend

Frontend GUI

- Displays the processed data in a GUI
- Can run on Ground Computer or a computer connected to the Ground Computer