Small Payload Launch Integrated Testing Services (SPLITS) – SPSDL

Benjamin Plotner
Kennedy Space Center
Electrical Engineering
KSC FO Fall Session
11/21/2013

My experience working on the Small Payload Launch Integrated Testing Services project has been both educational and rewarding. I have been given the opportunity to work on and experiment with a number of exciting projects and initiatives, each offering different challenges and opportunities for teamwork and collaboration. One of my assignments is to aid in the design and construction of a small-scale two stage rocket as part of a Rocket University initiative. My duties include programming a microcontroller to control the various sensors on the rocket as well as process and transmit data. Additionally, I am writing a graphical user interface application for the ground station that will receive the transmitted data from the rocket and display the information on screen along with a 3D rendering displaying the rocket orientation. Another project I am working on is to design and develop the avionics that will be used to control a high altitude balloon flight that will test a sensor called a Micro Dosimeter that will measure the total ionizing dose absorbed by electrical components during a flight. This includes assembling and soldering the various sensors and components, programming a microcontroller to input and process data from the Micro Dosimeter, and transmitting the data down to a ground station as well as save the data to an on-board SD card. Additionally, I am aiding in the setup and development of ITOS (Integrated Test and Operations System) capability in the SPSDL (Spaceport Processing System Development Lab).

Throughout my experience at the John F. Kennedy Space Center, I have had the privilege of seeing and experiencing the result of decades of hard work and dedication by the engineers and technicians that have made NASA's missions possible over the years and have shaped the nation's space program into the marvel that it is today. I have been presented with the opportunity to contribute to and participate in a number of exciting projects and initiatives that have given me immeasurable experience and have allowed me to develop and advance a vast variety of skills and capabilities, ranging from developing avionics for small scale rockets and high altitude balloons to developing and configuring command and control systems to communicate with spacecraft and satellites.

As part of the duties and tasks I was responsible for during my internship, I aided in the development and testing of the avionics that will control the operation and flight of a small scale two stage rocket. I was presented with the opportunity to partake in a Rocket University initiative that aims to gain and expand skills in developing advanced avionics that are able to control a rocket by detecting and responding to various events throughout a flight. The team I was working with, team ACME, was working to design and construct a rocket approximately twelve feet in length that would reach an altitude of about ten thousand feet. My efforts involved interfacing and configuring a variety of sensors to a microcontroller that collects, processes, and transmits telemetry data. The flight computer consisted of a Digilent chipKIT Max32 development board, which contains a Microchip PIC32MX795F512 microcontroller. This was interfaced with various sensors, including an Inertial Measurement Unit (IMU) that consists of an accelerometer, gyroscope, and magnetometer, an altimeter, a GPS receiver, an SD memory card, and a long range radio. The avionics package, which contains the flight computer and all of the sensors, can be seen below in Figure 1. The interfaces between the microcontroller and sensors were achieved utilizing serial communication protocols. The IMU and altimeter were configured and operated using the Inter-Integrated Circuit (I^2C) protocol, the GPS receiver and radio were communicated with using the Universal Asynchronous Receiver/Transmitter (UART) protocol, and the SD memory card was controlled via the Serial Peripheral Interface (SPI) protocol. Data was collected and stored to the SD card at a rate of approximately 200 Hz while being transmitted by the radio at a rate of about 5 Hz.

Figure 1: Rocket Avionics Package

As a compliment to the avionics, I was also tasked with developing alternative means of data collection and computing by utilizing different processing platforms. While integrating a microcontroller as the main processer of the flight computer proved to be efficient and effective, employing alternative solutions would provide a wide range of applications and make the overall system more flexible and adaptable. To achieve this, I began experimenting and programming an embedded Linux platform called a BeagleBone Black. Due to the embedded Linux environment, some aspects of programming, such as the serial communication interfaces, were much simpler and less code intensive to implement. However, because there is an operating system present, the ability to use software interrupts and events was limited therefore these constructs had to be implemented by other means. The BeagleBone Black also incorporated a much faster processor than the microcontroller, providing more processing power. In addition to the BeagleBone Black, I also attended a Rocket University training course that covered the use and implementation of Field Programmable Gate Arrays (FPGAs). The course covered a Xilinx platform known as a Zynq board. The Zynq architecture consists of an ARM processor integrated with an FPGA to provide the features and advantages of both systems in a single chip. The advantage of utilizing a Zynq board as a flight computer is the ability of the FPGA to perform parallel processing, enabling extremely fast data collection and processing without consuming the resources of the ARM processor. These alternative solutions create a very flexible system that can be adjusted based on specific application needs.

In addition to the avionics, I was responsible for writing ground station software to monitor and command the rocket during a flight. This involved writing a graphical user interface (GUI) program to receive, process, and display data for the user. To achieve this, I utilized the Java programming language. The program, as seen below in Figure 2, contains various fields to display telemetry data, buttons to command and control the rocket events, a 3D graphics rendering showing rocket attitude, a Google Maps plugin to display current location, and real-time graphs of altitude and acceleration experienced by the rocket. To receive the various data that is transmitted by the rocket, a radio receiver had to be connected to the ground station computer. The receiver, in combination with a Java serial communications library called RXTX, allowed the program to be able to collect the UART data that was transmitted. After the data is received, the UART string is then passed to a decoding function that separates the various values by searching for a delimiter between fields. The OpenGL graphics API was utilized to create and display the 3D graphics rendering of the rocket by accepting quaternion values that represent the orientation of the rocket in reference to the earth frame, and rotating the model appropriately to show attitude. The integrated Google Maps plugin is passed latitude and longitude values from the GPS receiver to show current rocket location. In order to display real-time graphs, a Java library called JChartFree was utilized to create and continuously add data points to represent altitude and acceleration.

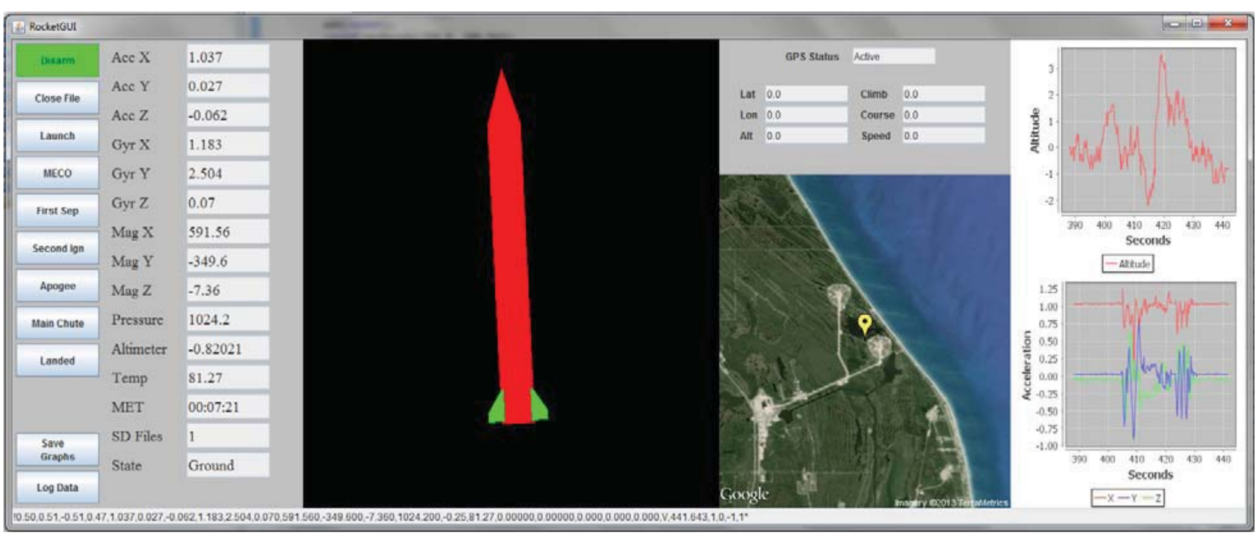

Figure 2: Ground Station Software

My duties and tasks also included the design and construction of the avionics that will control and guide the flight of a high altitude balloon as part of a Rocket University initiative. The purpose of the flight is to collect and process data from a sensor called a Micro Dosimeter that will measure the radiation build up on electronic components in the atmosphere. The avionics, consisting of a chipKIT Max32 development board, a GPS receiver, an SD memory card, a long range radio, and the Micro Dosimeter, will collect and process data from the dosimeter, and store the data to the SD card at a rate of one sample per second. The microcontroller communicates with the GPS and radio via the UART protocol while the SD card is controlled using SPI. The dosimeter outputs analog voltages on four different scales that represent the total ionizing dose of radiation. These voltages are then collected and processed through an analog to digital converter (ADC) within the microcontroller. The processed data is then transmitted to a ground station using the radio. I developed a breadboard version of the avionics for the purpose of testing and debugging, as seen below in Figure 3.

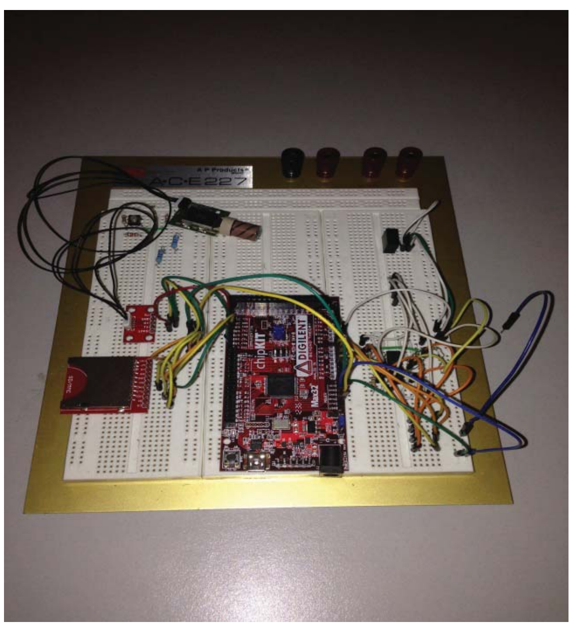

Figure 3: Breadboard Version of Avionics

My duties also included the installation and configuration of the Integrated Test and Operations System (ITOS) software suite on various computers in the Spaceport Processing System Development Lab (SPSDL). ITOS is a real-time command and control system developed by Goddard Space Flight Center to control spacecraft and satellites. My efforts included installing the software suite and configuring and developing test programs to demonstrate its capabilities and features. This involves creating and integrating databases containing definitions and formats of telemetry and command packets as well as writing a GUI to display data on the screen. The system was designed to be implemented on Linux platforms, therefore I installed the system on both RedHat and Ubuntu operating systems in the SPSDL. Execution of an example program that was included as part of the ITOS installation can be seen below in Figure 4.

Figure 4: Execution of Example Program

My responsibilities also included the installation and development of software developed by Gensym called G2. G2 is a real-time rule-driven expert system that automates decisions and monitors the performance of critical applications that require constant operation. My efforts included completing training to gain familiarity and develop skills in the development and implementation of expert systems as well as create test programs to demonstrate software capabilities. The G2 software has been identified as the potential platform for the cryogenic fueling of the SWORDS rocket being developed by the army. The graphical nature of G2 as well as the object oriented language it is implemented in (C++) resulted in a relatively quick and intuitive process for development.

My experience here at the Kennedy Space Center has left me with a wealth of knowledge and experience that will be invaluable to me when I return to school and in future career opportunities. Over the course of my three years thus far in college, electronics and embedded systems were among my favorite subjects to study and experiment in. I was extremely lucky to have been able to work in this area during my internship. The level of complexity and intricacy of the systems and technologies that I have contributed to have greatly increased my confidence and interest in embedded systems and I look forward to pursuing this area in my career. The past few months have been an incredible experience that I will never forget.