N91-19004

1990

NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER
THE UNIVERSITY OF ALABAMA

DEVELOPMENT OF SEDS DATA SYSTEM FLIGHT SOFTWARE

Prepared By: William E. Pierson

Academic Rank: Professor

University and Department: West Virginia Institute of Technology, Electrical Engineering

NASA/MSFC:

Laboratory: Program Development
Division: Payload and Orbital Systems Office
Branch: Orbital Support Systems Group

Contract No: NGT-01-002-099
The University of Alabama

DEVELOPMENT OF SEDS DATA SYSTEM FLIGHT SOFTWARE

July, 1990

Background

The Small Expendable-tether Deployer System (SEDS) is being built for a 1991 flight on a Delta II launch vehicle. SEDS will deploy a 50 lb (23 kg) instrumented satellite as an end mass at the end of 20 km tether. This end mass is being developed by Langley Research Center. The objective of the experiment is to test the SEDS design concept by demonstrating that the system will satisfactorily deploy the full 20 km tether without stopping prematurely, come to a smooth stop on application of the brake, and cut the tether at the proper time after it swings to the vertical. SEDS will also collect data that will be used to test the accuracy of tether dynamics models for this type of deployment. The experiment will last about 1.5 hours (or one orbit) with actual deployment lasting about 85 minutes.

Radar tracking of the experiment is planned. In addition, the SEDS on-board computer system will record, store, and continuously transmit data over the Delta II S-band telemetry channel. The system will count the tether windings as the tether unwinds, log the time of each turn, monitor tether tension, and record various temperature readings. The computer will also control the stepper-motor brake system and activate the tether cutter mechanism.

Two major sections of the flight software were developed under the 1990 NASA/ASEE Summer Faculty Fellowship Program: the data telemetry software and the data collection software. Initial testing of the software under simulated flight conditions was performed, also. These tests used both assembly language routines running on SEDS data systems and C-language routines running on an IBM-PC.

SEDS Hardware

The SEDS data system uses a National Semiconductor NSC800 CMOS microprocessor, which has the same instruction set as the Zilog Z80 microprocessor. The system includes various support circuitry to provide additional resources such as parallel and serial I/O, analog-to-digital conversion, and timers. Also provided is ROM for storing the flight software and RWM in the form of main menory and mass RAM for storing program constants and recording flight data. The system is powered by 28VDC, supplied by the Delta II, and operates at a 2MHz clock rate.

Measurements and Events

Table 1 summarizes the measurements taken by the SEDS computer. In addition to logging the time of each software counter change in mass RAM, the times corresponding to a software turns counter which is an exact multiple of 512 is logged into main memory. This serves as a backup source of information should mass RAM fail and also allows real-time monitoring of the tether length.

Two light-activated switches located 180 apart on the tether spool are used to sense tether unwindings. Using two sensors oriented in this manner allows detection of false triggers due to tether flutter and also provides some hardware redundancy. The software is capable of detecting a sensor failure and will record the failure in a Counter Failure Flag, which is transmitted along with measured data. Should one sensor fail, the software will continue to operate, however, false triggers will no longer be detectable by the SEDS data system.

Although the turns counters are updated asynchronously, the hardware counter values are written to the data storage buffers once every two seconds, and the software counter value is written once every five seconds. All measurements except the tension snapshot are taken and updated continuously for the duration of the flight. The tension snapshot measures and stores the high tension range for 16-second intervals at various stages during the flight. This will allow post-flight frequency analysis to be performed on tension data for critical flight periods. Since the tension is being sampled at 500 sps, frequency components as high as 250Hz can be detected.

Normal tether tension is collected as a one-second average. This tension may originate from the low-scale reading or the high-scale reading, and the measurement will carry two extra bits to indicate the scale from which the tension reading was taken.

TABLE 1
Summary of Data Collected During SEDS Flight

Measurement or Event	Data Length (Bytes)	Stored Length (Bytes)	Buffer Length (Bytes)	Flight Time (sec/min)	Sample Interval (sec)
Main Memory					
Turns Count A	2	1	3000	6000/100	async.
Turns Count B	2	1	3000	6000/100	async.
Software Turns Count	2	1	1200	6000/100	async.
Tension, 1-sec Average	2	2	12000	6000/100	1
High Tension Snapshot	1	1	8000	N/A	.002
Software Count Log Time	3	3	300	6400/107	async.
Temperature Readings	4	4	2400	6000/100	10
Turns Counter Failure	4	4	4	N/A	async.
Mass RAM					
Log Time of Software Counter Update	2	2	128K	8192/136	async.

Each measurement has been allocated sufficient memory to allow at least 6000 seconds (100 minutes) of data to be collected. In the event that the flight should exceed this duration (the data buffers become full), provision has been made to reset buffer pointers and restart the data collection process.

Flight Software

Five major modules to be used by the flight software were developed under the 1990 NASA/ASEE Summer Faculty Program. These modules perform two primary functions:

1) collect and store data from on-board sensors, and
2) continuously transmit data over an RS232 communications port.

A brief description of each module is provided below.

TXINIT: initializes data buffers, mass ram, program constants, and timer and turns counter interrupts

TXMT: foreground program which continuously transmits contents of data buffers and mass RAM using predefined formats

TXMT2: low-level serial output routines which interface with the RS232 port of the SEDS data system

TXINTR: background interrupt service routine for the 2ms timer interrupt; maintains real-time clock and monitors the occurrence of turns counter events; data buffers and mass RAM are updated at the prescribed intervals

TXLOG: routine to place a time stamp into mass RAM whenever the software turns counter changes

The data telemetry software formulates and transmits frames of information organized into two formats: data frames and master frames. One master frame is sent for each 15 data frames transmitted. Master frames allow the data frames to be interpreted by indicating the starting point of the various measurements. In addition, the master frame records the event of a counter failure and allows a real-time display of certain flight data. Data frames contain actual flight measurements.

Software Tests

Several levels of tests were performed to verify correct operation of the data telemetry and collection software. During the first testing phase simulated flight data was generated and stored in an unused area of the SEDS data system's main memory rather than being transmitted over the serial port. The data was later transmitted to an IBM-PC using the PROCOMM communications software and stored on the hard disk for further evaluation. This initial testing phase uncovered several programming errors, which were corrected.

The second phase of testing allowed the SEDS data system to communicate in real time with the IBM-PC over an RS232 communications channel. The test software was written so that sets of frames from the beginning, middle, or end of the simulated flight could be collected for analysis. The data collected by the PC software was converted to an ASCII (readable) format once data transmission had ceased. Additional errors were uncovered and corrected during the second phase of the software testing. The software written for the IBM-PC to perform the second phase of testing was written using the C programming language (Turbo C by Borland, Inc.) and C Async Manager, a interrupt-driven set of PC communications functions by Blaise Computing, Inc.

The final tesing phase tested both the data collection software and the data telemetry software running under simulated flight conditions. In order to perform realistic testing of the combined data collection and data transmission routines, signals emulating turns counter inputs were generated using a second prototype SEDS data system (serial number 1). This second system was programmed to generate well-behaved turns counter inputs for both hardware counters occurring once every 200ms. Test data was collected using an IBM-PC and revealed several programming errors which have been corrected.

Summary and Recommendations

The original objectives outlined for the NASA/ASEE Summer Faculty Program have been achieved. That is, software to collect and transmit SEDS flight data has been written and tested to the extent available resources will allow. Additional testing will have to be performed over the next few months to further assure that the software operates as intended. As the software now stands, relatively few additions are needed before the entire set of flight software is complete. Specifically, modules need to be added which will initiate the collection of tension snapshot data, activate the tether brake, and activate the tether cutter. Finally, the complete set of flight software needs to be tested in near-flight conditions to verify proper operation and results.

References

1. Harrison, J. K. and Rupp, C. C., "Small Expendable-Tether Deployer Systems (SEDS) Development Status", Paper Presented at the AIAA Third International Conference, May 1989.

2. "SEDS Data System Functional Requirements", Energy Sciences Laboratories, Inc. and Program Development, MSFC, May 1988.

3. SEDS Data System Data Collection and Telemetry Software", Final Report, NASA/ASEE 1990 Summer Faculty Fellowship Program, Contract No. NGT-01-002-099, Marshall Space Flight Center, Huntsville, AL.