

# **IP-BASED NETWORKING AS PART OF THE DESIGN OF A PAYLOAD CONTROL SYSTEM**

**Stephen Horan, Ryan Aaronsooke, and Daniel Jaramillo**  
**Klipsch School of Electrical and Computer Engineering**  
**New Mexico State University**  
**Las Cruces, NM 88003-8001**

## **Abstract**

As part of a project to develop small satellites, we have developed a combined ground station and flight computer that use IP-based networking for the command and telemetry data communications. The network uses a private IP network between the payload and the ground-station. Commands are sent to the payload as UDP short message packets. Status and real-time telemetry are sent as UDP text strings. Production data are sent as files using a ftp-type of data exchange. Production data types include numeric data (sensor data) and JPEG-formatted picture data (full pictures and thumbnails).

Details of the software used, challenges of making the system work over a low-quality radio link, and integration with the operating system will be discussed.

## **Keywords**

Protocols – Network, telemetry system, ground systems, remote site automation, and system management

## **INTRODUCTION**

The faculty, staff, and students at New Mexico State University (NMSU) began the design and fabrication of a nanosatellite under the Air Force University Nanosatellite program in 2003 [1]. The basic science goal of the satellite is to measure the earth's background brightness level at 300 - 450 nm. This level is needed to design cosmic ray detectors that observe the interaction of cosmic rays with the earth's upper atmosphere. The science objectives are to measure the 300 - 450 nm intensity at least once per minute with associated position and timing information. The payload originally developed for this mission was not chosen by the Air Force to be launched into low-earth orbit. An alternative flight was obtained on a high-altitude research balloon. For this flight opportunity, the mission was renamed to *BalloonSat*. These research balloons fly at 100 kft for 24 hours. With this type of mission, the sensor design and the software design can be tested. Also, the operations of the payload can be fully tested. While the science and engineering environments do not exactly match the orbital case, they are considerably more challenging than a bench test and can be used to validate the orbital requirements.

In this paper, we describe part of the design of the communications subsystem for the payload. The traditional method for this design is to use some form of serial communications link to support the command and telemetry operations. An alternative method was developed to support

these same operations: utilize UDP and ftp packet communications to transmit the data. In the subsequent sections, we will describe the data formats, the computer systems involved, the two ways of performing the communications, and the advantages of the UDP-based command communications link. Additionally, we will look at the control of the ground station over the Internet as part of the system design.

## DATA FILE STRUCTURE

The command and telemetry system for the payload is described in the companion paper [2]. The communications link needs to support both data types exchanged between the operator at the ground station and the payload that is remotely located. Commands are transmitted as short ASCII text strings of the form

\$\$*SSFN**data*\*

where \$\$ is the start delimiter

*SS* is the subsystem mnemonic

*FN* is the subsystem function

*Data* is an optional data field if required by the function

\* is the ending delimiter.

These commands can be sent as either discrete text strings from operator input or they can be transmitted in a file that forms a schedule of commands. The schedule file structure is one record per entry with each record being a time in seconds and the desired command to be executed at that time. An example record looks like: 3294583200,\$\$GPON\*

Telemetry data can come in three forms

- Command acknowledgement message
- Hourly data from the photomultiplier tubes and related payload data
- Imaging data from the camera system

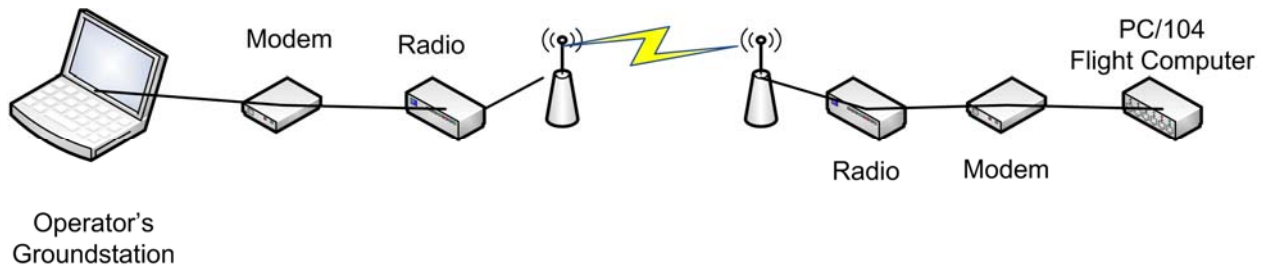
The command acknowledgement messages are short text strings, usually fewer than 50 characters in length. The hourly data files are also text-based and in record format: time and measurement. For example, a photomultiplier tube measurement would look like: 3294584325, 005C7823, 00002358. The first number in the record is measurement time; the second and third numbers are the measurement values from the tubes. These files are typically 10 to 20 kB in length, depending upon on how active the operator was in making measurements. This is similar to the command schedule file. The imaging data is in standard JPEG format. There are two image formats: full image JPEGs and reduced thumbnail images with sizes of 3 to 5 kB. The command acknowledgement messages are always immediate. The file data can be transmitted in background mode and/or stored on the flight computer flash drive for collection at mission end.

## COMPUTER SYSTEMS

There are two computer systems required in this system: one comprising the flight computer and one in the ground station. In this section, we will look at the individual computer systems and their associated requirements.

The flight computer is a Diamond Systems Athena II CPU. The computer has 256 MB of RAM and a 500 MHz clock speed. The computer has a 4 GB flash drive on the IDE bus acting as the hard drive. The operating system chosen for the flight computer system is Windows 2000. This provides the necessary support for development software and its size is compatible with the constraints of the 4 GB flash drive. A design goal is to have the operating system take up no more than 25% of the available drive.

The ground station is designed to operate completely with laptop computers. This need arises from the fact that to support a balloon flight, there may need to be two mobile ground stations in addition to the main ground station to support flight operations. Therefore, the entire ground station can be assembled with a laptop, portable radio, and VHF antenna. The software will need to run on any of the three laptops in the ground station locations. The laptops chosen are standard laptops running Windows XP.



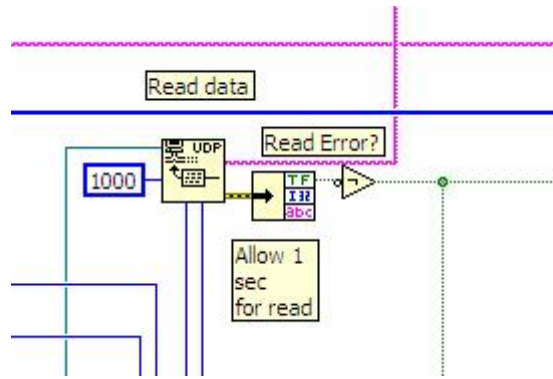
**Figure 1** -- Communications components for command and telemetry data transmission.

## COMMUNICATIONS METHODOLOGY

The “problem” to be solved for the command and telemetry system is to find an efficient means to transmit the command and telemetry data between the payload and the ground-station. The architecture of the communications link is illustrated in Figure 1. The endpoints are the operator’s laptop and the PC/104 flight computer. The intermediate hardware is a standard modem/radio/antenna chain. For the *BalloonSat*-type of mission, the link is a VHF radio link. For testing in the lab, the radio and antenna combination can be replaced with a cable between the modems. The modem provides channel-level HDLC framing with error detection and stop-and-wait ARQ. This is important for the command link since it removes the need for explicit error checking in the command format. The baseline method for transmitting the data is to use serial communications techniques. Figure 2 shows the typical code for receiving these transmissions. The code in the illustration is a segment of a LabVIEW<sup>1</sup> Virtual Instrument (VI) for receiving command strings from the operator. In this technique, the receiving unit needs to

<sup>1</sup> LabVIEW is a trademark of National Instruments Corporation





**Figure 3 --** A UDP-based read of the serial data stream.

For our application, we use *FlashFXP* as the networking control program. In this case, the payload is set up as the master node. The payload then assigns the groundstation the fixed IP address when the user logs into the payload from the groundstation. Once the IP addresses are established, then communications between the two computers can flow. If we use multiple groundstations, the payload is always the node to assign the IP addresses. In this application, we have the control programs in both the groundstation and the payload computers utilize the fixed IP addresses. This makes the addressing easier to manage in the control programs. It also makes training of operators easier because there are fewer parameters to manage.

By using the networking capabilities, we are able to have both the LabVIEW control program and the *FlashFXP* program running simultaneously. The LabVIEW program is responsible for command transmission and the associated responses. The *FlashFXP* program is used to transmit the hourly data files, the image files, and the schedule files. In this mode, the LabVIEW program does not get caught in timing loops waiting for serial communications to initiate or terminate. The program concentrates on command processing and instrument control.

### REMOTE GROUNDSTATION CONTROL

For the *BalloonSat* mission, we need the ability to monitor the payload from multiple locations. For our mission, the main ones are at Fort Sumner, NM and Holbrook, AZ. These sites are over 300 miles apart. There is also a desire to provide support from the home location in Las Cruces, NM, which is over 200 miles away from either control station. By using the Remote Desktop utility in the Windows operating system, we can actually control the LabVIEW-based user interface over the Internet. An example of this is illustrated in Figure 4 where the screen shot of the user interface was taken over the network between the computers. In this mode, the user on the client computer sees the host computer's desktop. The remote user can start the LabVIEW program and exercise the user interface as if actually at the control computer. This permits user support for these remote operations.

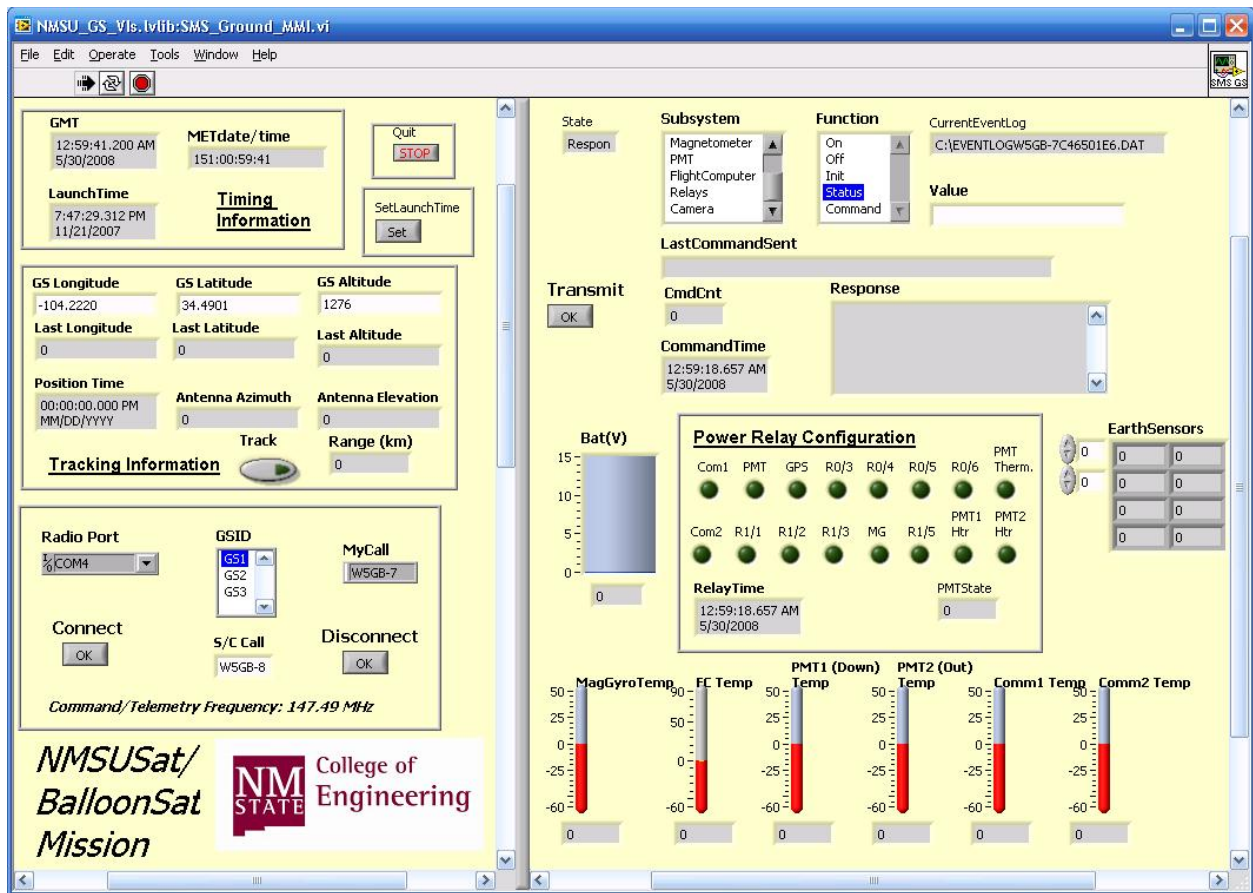


Figure 4 -- Groundstation control interface as accessed over the Internet.

During the testing of this capability, it was noticed that there was a need to make the link through a VPN connection. The remote groundstation at the Fort Sumner launch facility is connected to the Internet via an internal private network. This makes it difficult to access the laptop from the NMSU campus. In this case, we first connect the laptop to the NMSU network via VPN to give the laptop an IP address that is part of the NMSU domain. Using the VPN-based IP address, the connection is achieved. There was no need for the VPN between NMSU and Holbrook, AZ or between Fort Sumner and Holbrook. The Holbrook IP address was fixed and public. Naturally, the VPN might be desirable if link security is desired.

## RESULTS

The IP-based control program has been tested in the NMSU *flatsat* laboratory. It was not currently used in the *BalloonSat* mission that was flown earlier in 2008 due to the need to freeze the software configuration and the uncertainties in the accommodations at the support facilities when the mission was first designed. If we obtain subsequent flights, the plan is to move the communications mode to this format.

The remote desktop method was used during the actual mission operations during the *BalloonSat* flight earlier in 2008 to support controlling the payload from the NMSU campus through both

the Fort Sumner and the Holbrook, AZ groundstations. These remote operations times exceeded two hours in each case. This permitted running the mission with two operators with only one needing to be located at the launch point. In principle, the operator at the launch point could have been eliminated and the mission could have been run from just the NMSU location.

## CONCLUSIONS

We are developing networked-based communications techniques for both the command and telemetry links between the payload and the groundstation. We see that the UDP method of command transport is convenient to program and supplies the required functionality for command operations. Once the network between the nodes is established, an ftp application can be used to move configuration and data files between the payload and the groundstation. Having the payload control IP address assignment has been found to work better than having the groundstations assign the addresses.

We have also found the Windows remote desktop utilities to be an effective way to permit remote operators to control the payload across the network. This is important when payload support operations range over several hundreds of miles.

## ACKNOWLEDGEMENTS

The development of the software and the acquisition of the hardware were supported under Air Force Office of Sponsored Research grants FA49620-03-1-0168 and FA9550-05-1-0267. Support for the flight was provided by the International Foundation for Telemetry and NASA's Columbia Scientific Balloon Facility.

## REFERENCES

- [1] Horan, S., et al., "The New Mexico State University Satellite (NMSUSat) Mission," *Proc. 17<sup>th</sup> AIAA/USU Conference on Small Satellites*, SSC03-IX-5, Logan, UT, August 2003.
- [2] Horan, S., "Using LabVIEW to Design a Payload Control System," *Proc. International Telemetry Conference*, San Diego, CA, October 2008.