Implementation of a Remote Acquisition and Storage System

Jason R. Hess
Kenneth D. Wright, mentor

Internal Operations Group
Experimental Testing Technology Division
Acoustical, Optical, and Chemical Measurements Branch

Abstract

The existing system for gathering and processing acoustical test data had several shortcomings and limitations in the areas of microphone array size, sampling rate, and background noise. A new remote acquisition and storage system (RASS) is being designed for applications not suited for the existing acquisition system. One of the first tasks in the design of the RASS was to redesign the microprocessor card of the existing system to include RS-232 serial ports to accept communications through the radio modem used in the RF link. Cost and parts availability comparisons were made between the newly designed board and commercially available models, and a commercially made model was selected. This model was tested for basic I/O operations. The prototype of the RF telemetry system was set up and tested. Plans are now being developed for integrating the RF telemetry system with the other RASS subsystems.

Introduction

The standard configuration for acoustical testing is an array of microphones that are set up to capture the acoustic signature of a test aircraft. This test requires a data acquisition system that can retrieve the data from the microphones, convert it to digital format, do any necessary processing, and store the information for post-test analysis. In the current data acquisition system, each microphone is wired directly to a remote digitizer box containing an analog to digital (A/D) converter. The converter digitizes the microphone output, improving noise immunity and data processing. A microprocessor controller card keeps track of the run numbers and time codes and generally controls the execution of the test. Each remote digitizer box is directly wired to a base station in an acoustic van that collects, processes, monitors, and stores the data.

This system has several limitations that are inherent in its structure. As noted above, each remote box is directly wired to the base station. The connecting cable transmits the collected data in real time, and also sends and receives control information from base station. It is desirable for the microphones to be at least 1000 feet away from the base stations because of noise reasons, and there are some acoustic arrays that require booster boxes every 1000 feet to retransmit the signals. Thus, for a typical test session with two base station vans each handling nine microphones at 3000 feet apiece, approximately 54,000 feet (more than 10 miles) of cable must be laid. This is assuming that the cable can be laid in a straight line. There are instances where the test terrain contains lakes or swamps that do not permit a straight-line cable from the base to the remote sites. In this case, a new, longer route for the cable must be chosen, extending the time and work required to set up the experiment. It is not enough to just lay the cable, however. Since some animals are prone to chew through exposed cables, each cable has to be suspended with special rods for protection. Thus, large arrays can require extensive setup times.

The current setup also has a couple of other limitations. For one, the finite amount of cable and time to lay it means that the ultra-sensitive microphones will pick up background noise from the vans and generators. While this noise usually can be filtered out from the significant data, there are some cases where the background noise interferes and makes data analysis very difficult.

A third limitation with the current system deals with the way the data is collected. Currently, as soon as the data is collected and digitized, it is transmitted to the vans via the cables. A computer at the base station then receives these different data streams simultaneously and must multiplex the signals to store them together on a single medium. Since the data is collected in real time, the processors back at the vans must be able to work at or above the sample rate of the data. Because of the physical limits on the speed of the storage medium, there is a ceiling on the sample rate and the number of channels that can be sampled with the current system.

To solve these problems and to provide more flexibility when testing, the acoustics team in the AOCMB began developing a system that would rely on radio communications to relay information between the base and the remote sites. It is called the remote acquisition and storage system, the RASS. This system has a similar basic structure to the current system, except there are no cables. The RASS relies on a radio link in the 403-416 MHz range to handle all of the control and diagnostic information from the main base. Because the radio modems in the RF transceivers are not very fast (9600 baud), it is not possible to transmit the data to the main base as it is collected at a sufficient sample rate. So, each remote unit in the RASS will contain a data storage device that can hold a full test day of data. The system information handling will also be upgraded to handle not only the run numbers and time codes, but also to monitor health data of the system, like temperature and battery voltage. The microprocessor controller card will act as the interface between the microphones, the health data sensors, the data storage device, and the RF transceiver at the main base station.

These improvements provide several benefits that remove many of the limitations of the current system. The combination of the RF transceiver and the remote data storage device eliminate the need for cables, saving a great deal of time and work in setting up the system. The remote setup also eliminates any problems caused by terrain that would not be compatible for direct wiring. This freedom in system configuration combined with the five mile range for the RF transceiver gives the RASS great flexibility for acoustical testing. For example the RASS can be used in residential neighborhoods to sample the acoustic footprint left by a supersonic aircraft without having to run a large amount of cable through the neighborhood. It also makes it possible to conduct tests in places where it would be difficult to run cables, like underground tunnels. All that is required is that the antennas be placed somewhere with RF visibility. The five mile range of the RF transceivers also greatly reduce any possibility of noise interference from the vans or generators. The addition of the storage devices at the remote sites removes the need for real time transfer of the data. The large storage capacity of the devices means that researchers could run a full day of tests and then go out to the remote sites to pick up the data, which would be analyzed later. This also removes the limit on the sample rate and number of channels. Since the data can be stored on separate channels and then multiplexed into a single stream at a later time, there is no hardware speed limit to restrict the accuracy and breadth of the test.

Project Summary

The remote acquisition and storage system consists of four main parts: the analog/digital converter, the data storage device, the system information handling board, and the RF transceiver. Full implementation of the system involves designing or acquiring and testing the required components and writing code to make the parts work together. The A/D converters and the data storage systems are not too difficult to implement since the A/D converters are already in use in the current system and the data storage systems are SCSI tape drives. More work is required to have the other two components working properly. The controller cards from the system information handling have to be modified to use RS-232 serial communications so that they can communicate with the radio transceiver. Then a program has to be burned into its ROM chips that tells it how to communicate and interface between the various parts of the system that it is attached to. The RF telemetry system is the only totally new link in the RASS, so it requires a good deal of testing to check its capabilities, its reliability, and its ability to communicate with its client ports. The RF system also has to be programmed to set up the network configuration and to fill in the communications parameters like baud rate and polling rate.

My project focused on the two latter components. Specifically, my tasks for the summer were to design and test RS-232 ports for the microprocessor controller board and to set up and test a prototype RF system. To prepare myself for these tasks, I started out by training on the CHAS 68000 microcomputer and by reading some background material on RS-232 communications. The CHAS is an open-board computer that introduces the user to microprocessors, specifically the Motorola 68000 microprocessor. The computer can be hooked up to a wide range of training peripherals through its I/O lines. The user then learns how to use the computer by programming it to manipulate the inputs and outputs of the peripherals.

After training with the CHAS, I moved on to my first task for the summer, the integration of RS-232 communications into the existing board. The controller board was 3.3" x 8.3" and it contained two CY7B144 dual port RAM chips, a MC68000 microprocessor, three MC68230 PI/O chips, an EPM5128JC programmable logic device (PLD) chip, two 27256 EPROM chips, two external ports, and miscellaneous resistors, bypass capacitors, switches, and LED's. My first design for the addition of RS-232 communication required an additional three chips and two external serial ports. Two of the chips were actual RS-232 drivers that would convert TTL/CMOS signals from the chips on the board to RS-232 values for transmission. Each chip had two transmitting and two receiving channels, one for data and the other for a handshaking signal, and they could each service one serial port. The third chip was a DUART (dual universal asynchronous receiver/transmitter), which served as an interface between the RS-232 chips and the rest of the board. The main function of this chip was to convert parallel signals from the microprocessor and PI/O chips to a serial form that the RS-232 chips could use. Since the current board was already highly populated, it would be difficult to find room for three extra chips, two ports, and all of the required interconnections on the PCB. If they could not fit, we would have had to consider using a board with two levels or a multi-layer board that had the printed interconnections running through several layers in the board's thickness, which would give more space for actual chips on the board. I was able to simplify the design by replacing the two RS-232 chips with a single chip that had four receiving and transmitting channels. I then used P-CAD, a circuit drawing and modeling program, to redraw the board schematic with my suggested modifications included. My next step was to order the chips so that a prototype board could be manufactured and tested. I discovered that many of the chips that were currently on the controller board were being phased out of production as they had been replaced by models with more integrated functions. This gave us the option of either buying the last of the parts on the market or redesigning the board. Trying to use the current design would have been difficult because of the unavailability of the chips in a temperature-resistant ceramic packaging, the long lead times to make the chips since they were rarely in stock, and the scarcity of spare parts if any of the chips were to fail once the system was up and running. Redesigning the board, while it could have saved space on the board, would have incurred serious costs and time losses for redesign of the hardware and possibly converting the software if the new processor were incompatible with the old one.

The solution to the problem came unexpectedly. We discovered a commercially available single-board computer that fit our needs. This board, Z-World's Little Giant ™, had all of the features of our current board plus RS-232 ports, the ability to program it in assembly or C, an adequate operating temperature range, and an optional expansion board with 96 bits of digital I/O. Even better, buying this board was more cost-effective than modifying the existing one.

I spent almost two weeks writing simple C programs to test out the board's I/O capabilities. Unfortunately, a large portion of this time was spent trying to find errors caused by a mislabeled jumper position in the documentation. After the error was corrected, the board performed flawlessly.

My other main task for the summer was to set up and test the prototype RF telemetry system. The system that was used as a prototype was the MAVRIC 2000 ™ from Metric Systems Corporation. This system used a STAR-RING topology, meaning that there was a primary station in a central hub position which could send and receive data from secondary stations spread out in all directions. Each of the stations was equipped with three client serial ports which allowed peripheral devices to hook into the system. A fourth serial port, the radio port, allowed each station to hook up with a PATHFINDER 9600 ™ radio modem that handled the actual RF transmissions. The primary station polled each of the secondary stations at a user-specified rate to send data and check for data that needed to be received. The stations could be configured to route data between each of their client ports and any of the ports on any of the other stations. Thus, the main purpose of the MAVRIC system was to create a virtual circuit (no wires) between any two peripheral devices.

For expandability, each primary station could handle up to 100 secondary stations, depending on the desired polling rate and data packet size. If this was not enough, a network of primary stations, each controlled by a centralized hub station, could be configured to give maximum networking capability. Also, the transceivers could be set to use any of 15 different frequency channels to keep signals from getting crossed in complex networks.

The MAVRIC system was also highly interchangeable. Each MAVRIC was identical as far as hardware was concerned. The only thing that made one unit differ from the other was the user-written configuration file. This file defined the type of node (primary or secondary), the data path over the RF connection for each of the client ports, and various other parameters like poll rate and baud rate. The only other difference was that the transceiver for the primary station contained an optional power amplifier and would use an omnidirectional antenna, whereas the secondaries would use directional antennas. This allowed for flexibility when debugging because problems within the actual unit could be checked by quickly switching in a different one to see if the problem persisted.

The testing of the RF system was no small endeavor. The test setup included two MAVRIC stations with radio modems, each with a VGA monitor and a keyboard. Each station was connected with a null modem cable on its client port 1 line to an 80286 computer (another keyboard and monitor!) that was running the PATHWORKS ™ test software for the MAVRIC. Also, each modem required its own 12V power supply. For the first test, another null modem cable was run between the radio ports on the two MAVRIC's, effectively taking the modems out of the loop. The test software sent a data packet to one of the stations, which then transmitted it over the hardwired radio lines to the other station, where it was routed back to the test computer. This test would have been simple enough had it not been for my inexperience coupled with some key omissions in the documentation and some alterations that had to be made to the configuration files for the test to work.

The next phase of testing was identical to the first except that the radio link between the two MAVRIC's was enabled. This time, I encountered problems with a confusing setting in the test software and with the power supplies not delivering enough current to the modems. In fact, we are still working on supplying enough current to the modem with the power amplifier. After these hurdles were cleared, the system passed with flying colors, even when test data lengths were constantly varied in content and length and the distance between the modems was increased.

Looking toward the future, there are several steps left to be taken before RASS is operational. The RF telemetry unit must be tested for speed and accuracy at distances similar to those that will be used in the tests. Also, the tests must be expanded to include more MAVRIC's with the single board computers acting as the peripheral clients. Code must be written for the Little Giant, so that it can handle the different devices that it will be controlling and monitoring. But for now, the existing parts of the RASS appear more than adequate to do their respective jobs.