TURNKEY TELEMETRY DATA ACQUISITION AND PROCESSING SYSTEMS UTILIZING COMMERCIAL OFF THE SHELF (COTS) PRODUCTS

Amro M. Alawady
Test Engineering Specialist - Telemetry and Testdata
Lockheed Martin Vought Systems
PO Box 650003, M/S EM-25
Dallas, TX 75265-0003

ABSTRACT

This paper discusses turnkey telemetry data acquisition and analysis systems. A brief history of previous systems used at Lockheed Martin Vought Systems is presented. Then, the paper describes systems that utilize more COTS hardware and software and discusses the time and resources saved by integrating these products into a complete system along with a description of what some newer systems will offer.

KEY WORDS

Telemetry Processing, Telemetry Acquisition, COTS, Turnkey Systems

Introduction

The Telemetry and Test Data group is responsible for instrumentation, telemetry data transmission, acquisition, real-time displays and post-test processing of test data. Multiple data streams can be gathered from missile telemetry, simulations, launcher, and/or other instrumentation. In general, data is gathered, displayed, processed to engineering units and delivered to project engineering personnel for analysis.

In previous years, the hardware and software used to acquire and process data was a mix of vendor hardware, in house built hardware and custom written software. Because of coexisting projects and multiple telemetry streams within projects, the data acquisition systems were required to handle varying types of telemetry streams. These streams ranged from MIL-STD 1553, PAM and PCM to non IRIG standards or modified versions of the IRIG standards. Up until a few years ago, many vendors worked only with the IRIG or MIL-STD definitions, and the products they developed for these standards offered many more functions than were necessary for data acquisition. Because of this, hardware and software was developed in house to handle varying input streams into a common hardware interface and software processing routine.

Only recently have vendors started offering systems that allow for varying input streams, from IRIG to MIL-STD to user defined streams. They are also providing software tools, by way of off the shelf products, for quick and easy modifications to their systems. This shifts the time and resources spent on custom developed systems to developing complete turnkey systems based on industry standard hardware and software.

The course of this paper will take us from previous versions of data acquisition and processing systems to current and future versions, emphasizing the reduction of in house developed hardware and software. Through this evolution, the role of a telemetry and test engineer changes from one of a hardware designer or software programmer to a systems engineer, concentrating on the whole telemetry process from acquisition to analysis.

Previous Systems

The previous versions of data acquisition and processing systems were a small step towards elimination of custom developed systems. Figure 1 shows this version of a data acquisition and processing system. Hardware was broken down into two main blocks; telemetry decommutation and data acquisition and processing.

Telemetry decommutation consisted of rack mounted equipment, mostly acquired from outside vendors. They included hardware such as time code readers, bit synchronizers, frame synchronizers and telemetry receivers. Any type of non standard telemetry stream required a custom built interface board integrated into the system. This part of the system also contained the interface to the minicomputer, where telemetry data was merged with time, also known as time tagging, and sent via parallel cable to the computer. Again, this interface was an in house developed piece of hardware.

Data acquisition was accomplished using a MicroVAX II minicomputer. A Q-BUS parallel interface board, with direct memory access (DMA) to main memory, was used to connect the telemetry hardware to the minicomputer. Even though this board was an off the shelf item, custom drivers were required to control the board

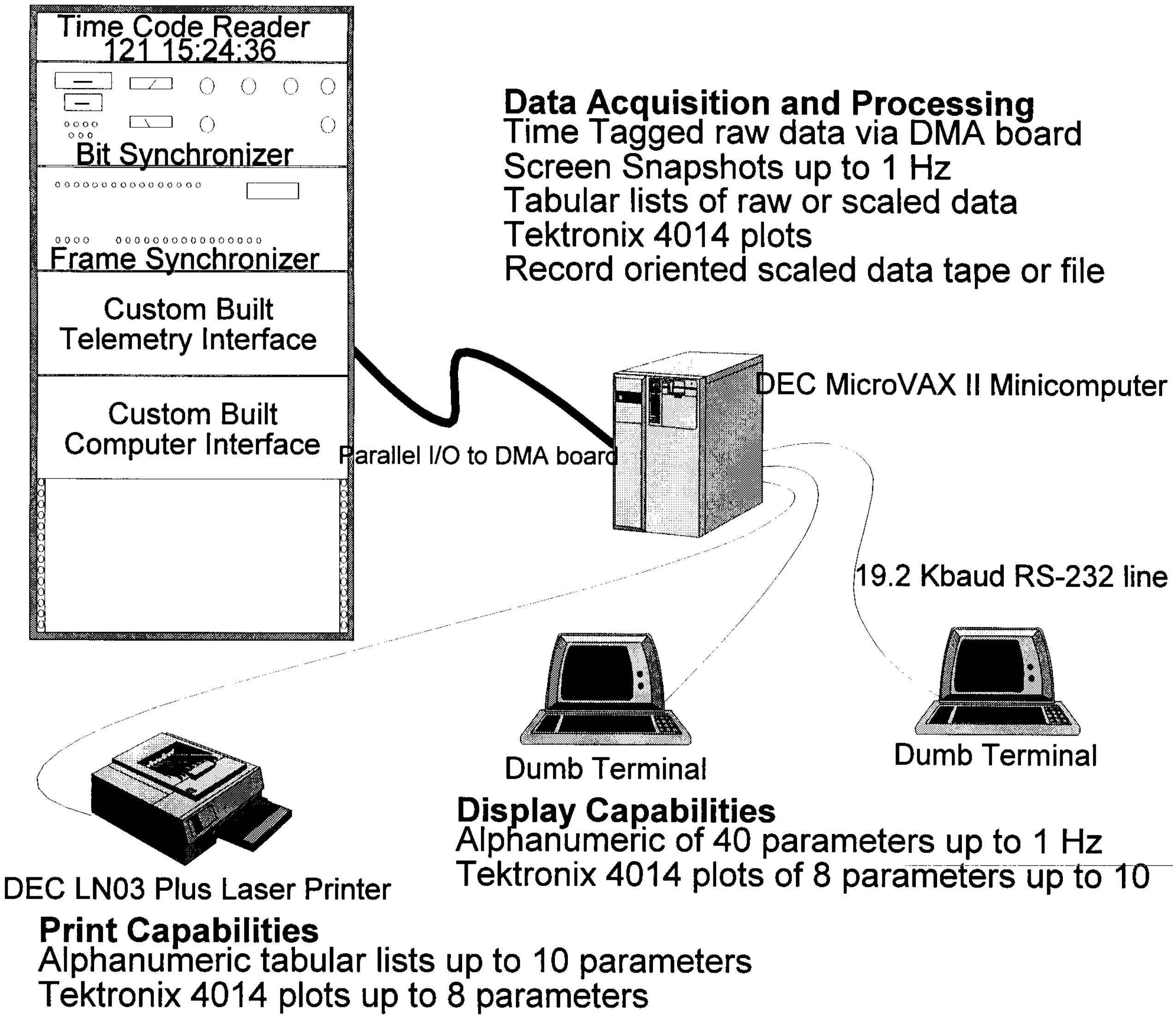

Figure 1

and the flow of data in the computer. Once data was in memory, more custom software was developed to take the data, acquire it to disk, and/or display the data to dumb terminals. Real-time displays consisted mostly of scaled data displayed in alphanumeric format in two columns of twenty parameters, with screen updates to 1 Hz. An alternate format was to display the data in a time history based plot of up to eight parameters per screen with screen updates to 10 Hz. The alphanumeric displays were made compatible with the VAX screen manager subroutines and the plots were made compatible with the Tektronix 4014 format.

Data processing was also run on the MicroVAX. Again, custom software was developed so that databases and processing routines could handle the mix of telemetry streams. The output generated from the post test processing routine was in the form of tabular listings, time history plots and further calculation files (scaled data written to a record oriented file).

The main advantage this custom built system had was that any type of telemetry format could be input into the system. The interface board not only time tagged each frame of data, but a header word was also added to the beginning of each frame. This had the effect of creating a common data format. Once in the system, software was used to identify the differences between the streams, and data was scaled accordingly. Also, since this system could handle any type of stream, it could be used for any project and only parameter scaling information had to be changed to acquire and process the data. The disadvantage of this system, was that a lot of resources were used to build the system and little time was spent on the actual scaling and processing of data.

Current Systems

The MicroVAX system utilized almost all custom written software in three languages. The DMA board software and acquisition software were written in assembly language. FORTRAN and C were used for real-time and post test processing. The rack mount hardware was big and bulky, and setup for each unit was generally a manual procedure. Around this time, telemetry hardware as well as software had been developed to comply more openly with industry standards.

Rack mount hardware was now being redesigned to board level products conforming to the industry standard VME bus architecture. A VME based computer was the natural selection for the acquisition and processing system. The selected computer also met new requirements of industry standards. A UNIX based system was selected due to the greater amount of off the shelf software being written for the operating system. Also, the selection of a VME base computer used in conjunction with VME based telemetry hardware combined the previous separate systems of rack and computer to one rack mount computer with telemetry products integrated into the computer. See figure 2. Time code boards and frame synchronizer boards now plugged into the same bus as the computer. These boards were provided with a set of software drivers to help cut down on custom software development. With the call of a few subroutines, a PCM decommutator board could be initialized and loaded with a telemetry format ready for decommutation.

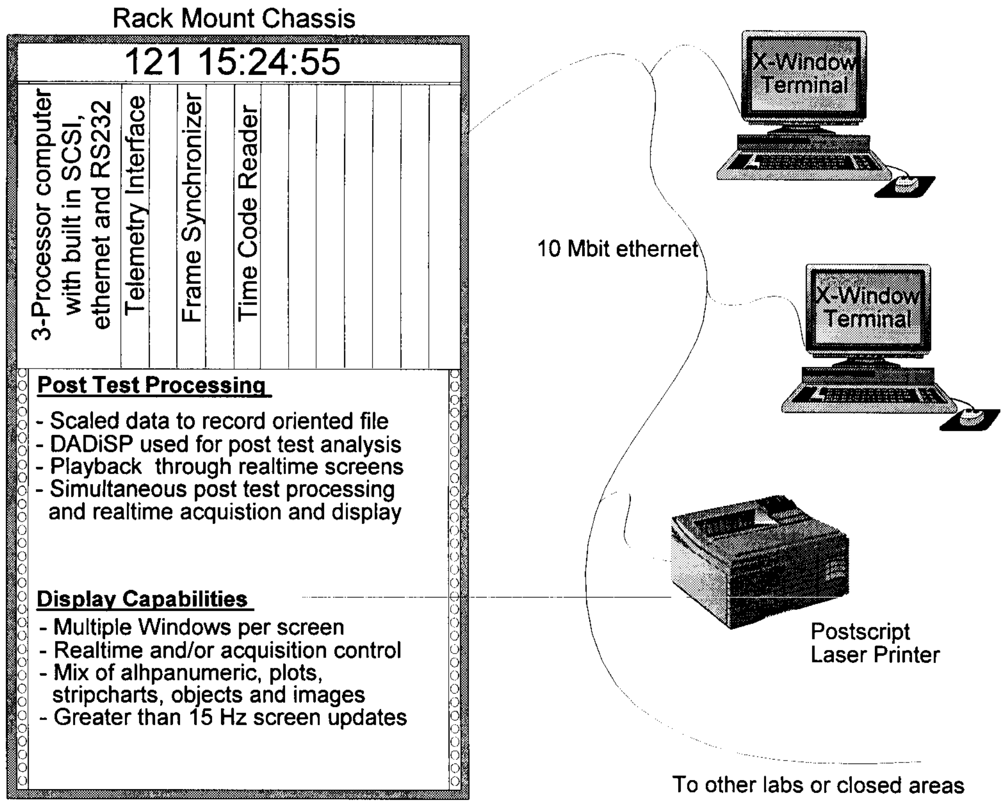

Figure 2

Although custom written software was still developed, the focus of the code turned to real-time scaling and processing. No more code was written for memory management, real-time displays and post test analysis. DMA boards were provided with a set of application programming interfaces (API's) for setup and initialization. Dataviews software was selected to develop real-time displays. It contained both a draw section to develop the look and feel of the displays and a set of API's to integrate real-time data with the displays. Unlike the older MicroVAX systems, views can be a combination of alphanumeric, plot or strip chart data and objects such as buttons, dials and images.

On previous systems, post test processing and analysis generally occurred in the form of tabular listings of scaled data or hardcopy time history plots. Test data could last anywhere from a few seconds to several hours. This meant that vast amounts of paper were printed or "thinned" data was printed to get an overall view of the test, and data was rerun to narrow the focus to a particular time period. With the ever increasing complexity and data rates of telemetry data, the standard hardcopy outputs became unmanageable. Because of this, it was determined that scaled data should be written to a file or tape and taken elsewhere for workstation or PC analysis.

Data was created on a per user request. Because of this, many individual runs were created, each run possibly containing similar parameters from other runs. To eliminate the redundant parameter scaling, the idea of scaling all the data in a telemetry stream and writing it to a file grew in popularity. A system needed to be created to allow a test engineer to analyze his or her particular parameters. The solution was the use of workstations and/or PC's and off the shelf software developed specifically for data analysis. In the past, workstations were fast but prohibitively expensive and PC's were slow and unsophisticated. Only within the last five to six years have workstations become affordable and PC's more versatile. Several data analysis packages are now available which run on these systems. DADiSP software was selected for post test analysis. It is a graphical type program with each parameter of a telemetry stream displayed as a graph in its own window. From one to one hundred windows can be displayed at a time. Math functions, data reduction, overplotting and window manipulation can be applied to each or all windows. Data from separate runs can coexist in one session. Batch files and macros can also be used to automate sessions.

The combination of workstations and third party packages, creates a new powerful tool for test data analysis. All data is available to all engineering, on-line and easily accessible. Hardcopy processing is all but eliminated and used only for presentation purposes. Because third party packages are used, they can be integrated into a network of computers that already exist within a project.

These systems are built on the same philosophy as the previous systems, yet take advantage of the latest release of hardware and software. The telemetry interface module is essentially the same as the previous system except built to VME specifications. The data format acquired into the computer is the same as in the previous systems. However, since these systems are based on more open standards, they are easier to design and implement. Resources are now concentrated on the delivery of data, whether in the form of real-time displays or post test analysis by way of third party software.

Future Systems

Currently, two new types of data acquisition and processing systems are being implemented to support new projects. Both of these system's goal is to further reduce custom hardware and software development. The current systems eliminate most of the custom written software for hardware control and post test analysis. However, custom hardware still exists in the acquisition of differing telemetry formats, and custom software is still used to scale the data for either real-time displays or post test analysis. The new systems all but eliminate the last bit of custom design. They are more of a complete "turnkey" system. Telemetry hardware, acquisition software, parameter scaling, real-time displays and post test processing are all integrated into one package.

The first major step these systems take is creating a common format for multiple telemetry inputs of varying types. They do so by “id” tagging each word within a telemetry stream. This has the effect of eliminating differences between formats. They have also created interface boards which allow for any type of non MIL-STD or IRIG standard stream to be acquired into the system. These streams are also id tagged. Parameters, along with their id tags, are placed on a high speed bus. Once on the bus, the differences between the telemetry formats disappear. They exist on a common bus in a common format. At this point, anything can be done with a parameter or mix of parameters. The data can be either output through a digital to analog (D/A) converter, scaled, acquired or displayed; or, the data can run through all these functions. Once a common format is created, a standard set of software routines can be developed to process this data for real-time displays, acquisitions or post test analysis.

The complete integration of telemetry hardware also allows the systems to grow in function. These systems can now be used to control telemetry transmitters and other external equipment. A flight termination system is being integrated into one of these systems. There is no proprietary hardware or software. The simple fact that the hardware and software comply with industry standards allows integration of a flight termination system with little overhead or cost.

These systems also take advantage of off the shelf software. Both systems utilize third party packages to help develop real-time displays. They also use the UNIX operating system as the base system. The common windows environment (X11 and Motif) are used to eliminate proprietary software development. This also permits vendors to retain systems open enough for future modifications and allow some customization.

By using these systems, time and resources can now be spent on data delivery. The acquisition and processing systems are no longer constraints to the analysis process. The risk with this approach is that these systems, no matter how open they claim to be, are still somewhat specialized. The high speed data buses are not compatible between the two vendors and are not based on any industry standard high speed bus. Therefore, the interface cards are not interchangeable between systems.

Conclusion

Table 1 shows the evolution of past, present and future systems from distinct key components to an integrated system. The combination of hardware and software into a “turnkey” system allows the telemetry engineer to concentrate more on the delivery of data to project personnel. Custom hardware or software development is reduced to a minimum. Experience has shown that, although a few analysts know what to do with their data, many do not. Simply acquiring and scaling the data is not enough these days. Resources can be focused on working directly with project personnel and helping them to better analyze the data they receive. Third party packages are just now beginning to cater to the needs of test data analysis. By improving this process, the overall project benefits from a reduced turnaround time of data analysis and gains an increased knowledge of test data performance.

Also, as an industry, we can work on getting more standards developed for turnkey systems. Proprietary high speed buses should be a thing of the past. Several industry standard high speed buses, such as Raceway or PCI for VME, currently exist. By conforming to these standards, telemetry board makers can offer their products on more systems, and the turnkey system vendors can concentrate more of their effort on a complete system. They have done so by integrating third party packages into their systems. They could take the next step and eliminate proprietary hardware interfaces.

	Previous Systems	Current Systems	Future Systems
Telemetry Hardware	Rackmount with manual setup	VME card with software drivers	VME card integrated into a turnkey system.
Telemetry Interface	Custom built. Supports only 1 telemetry stream	VME custom built. Supports only 1 telemetry stream	VME based high speed bus, with multiple telemetry streams.
Acquisition Software	Custom written in C, assembler, FORTRAN	Custom written in C through third party API’s	Vendor integrated into turnkey system
Real-time Displays	Custom written from scaling to displaying	Third party package integrated via software API’s.	Vendor integrated third party package into turnkey systems.
Parameter scaling	Custom written.	Custom written.	Vendor integrated into turnkey systems allowing for custom software module integration.
Post Test Analysis	Hardcopy tabular lists and time history plots or written to tape to be analyzed elsewhere.	Network of workstations with third party data analysis packages.	Network of workstations with third party data analysis packages.
Computer	MicroVAX	Concurrent 7000	Any UNIX based system

Table 1