REAL-TIME TELEMETRY DATA PROCESSING and LARGE SCALE PROCESSORS

Mr. Harold N. Dreibelbis
H.D. Consultants, Ltd
2521 So. Vista Way, Suite 136
Carlsbad, CA 92008
(619) 942-5525

Mr. Dennis Kelsch
CRAY Research, Inc.
5655 Lindero Cyn Rd, Suite 100
Westlake Village, CA 91362
(818) 991-7738

Mr. Larry James
LPJ Engineering
4618 Harmony Lane
Santa Maria, CA 93455
(805) 937-2707

ABSTRACT: Real-time data processing of telemetry data has evolved from a highly centralized single large scale computer system to multiple mini-computers or super mini-computers tied together in a loosely coupled distributed network. Each mini-computer or super mini-computer essentially performing a single function in the real-time processing sequence of events. The reasons in the past for this evolution are many and varied. This paper will review some of the more significant factors in that evolution and will present some alternatives to a fully distributed mini-computer network that appear to offer significant real-time data processing advantages.

KEYWORDS: Telemetry, Real-Time Systems, Large Scale Processors, Shared Memory, Flight-Test.

INTRODUCTION: In the late 1960s it was readily apparent to most of us in the real-time data processing community, that the use of large scale general purpose computer systems to accomplish the ingest and processing of high-speed real-time data was not only very impractical but also very expensive. The ability of the IBM System 360, CDC 6600/7600, and other such general purpose systems to ingest and process real-time data was just not on a par with their ability to process large amounts of batch data. The ability to service interrupts and move data through their Operating Systems in a timely fashion was something that they could never seem to quite make work efficiently. There was also some hesitancy from the Corporate level to make the necessary changes to the data ingest capability and operating systems to support the real-time world, as that would entail maintaining two different operating systems and sets of data channels. Those of us who still had to play in the real-time data processing arena, took what we had learned in the laboratory at school, and utilizing the existing capabilities of such mini-computer systems as the PDPs and the System Engineering Labs (SEL) Data Acquisition processors, developed our own real-time operating systems and high-speed data ingest systems. As our requirements grew to ingest and process more and more data in even shorter time periods in the 1970s and 1980s, we developed distributed networks of mini-computers and distributed operating systems to support them. As we outgrew their processing capability and the computer industries ability to build faster systems to meet our requirements, we started using shared memory systems, reflective memory systems, and more processing power in the Front End equipment to allow us to continue to meet our real-time data processing requirements. Our success rate in meeting these continually growing requirements for more and faster real-time data processing by the users of these systems is not very good. The complexity of transferring large quantities of data between shared/reflective memories and distributed networks of minicomputer systems has not been adequately solved and becomes more and more expensive each time we try it again. Maybe it is time to relook at the ability of large scale processors to provide the real-time ingest and processing capability to meet our ever growing real-time needs.

PROBLEM STATEMENT: What type of processing platform should by used as the base to meet the real-time telemetry data processing requirements that result due to increased system user needs for higher and higher telemetry downlink rates and higher capacity on-board systems?

TECHNICAL DISCUSSION: We will now look at three successful implementations of real-time telemetry data processing systems: (1) the Telemetry Integrated Processing System (TIPS) installed at Vandenberg AFB, CA in the late 1970s; (2) the Integrated Flight Data Analysis System (IFDAPS) installed at Edwards AFB, CA in the mid 1980s; and (3) the Test Support Facility-Integrated Flight Data Analysis System (TSF-IFDAPS) installed at Edwards AFB, CA in the late 1980s. We will compare the actual capabilities of these three systems against the original user requirements and quantify their shortcomings. We will then look at future telemetry system downlink requirements and large scale processors and describe how these large-scale systems could satisfy those shortcomings, as well as, meet the ever expanding requirements for more and faster real-time telemetry data processing in the future.

TELEMETRY INTEGRATED PROCESSING SYSTEM (TIPS)

The TIPS was designed and developed for the Western Space and Missile Center (WSMC), Vandenberg AFB, CA by Systems Development Corporation (SDC) during the mid to late 1970s to support ballistic and space missile launches and Test and Evaluation programs. There were four significant development aspects of the TIPS system which contribute to the technical discussion that is the subject of this paper. These are: (1) the Fast Multiplexer System; (2) the use of shared memory blocks; (3)

the development of a Distributed Operating System (DOS); and (4) a tightly coupled distributed telemetry processing network so that the entire TIPS could be operated by 2 to 4 operators at full load while maintaining positive control of all TM data. There were a number of other significant development aspects to the TIPS, but they were concerned with peripheral devices and telemetry processing software and are not germane to the technical discussions for this paper.

DISCUSSION: The TIPS currently includes eight real-time telemetry data acquisition and preprocessing processors (TPPs), six interactive real-time telemetry data display area processors (QLDAs), four mass storage control processors (MSCs), three configuration interface control processors (CICs), and a large scale near-real-time batch processor (NBP). The twenty one real-time processors (TPPs, QLDAs, MSCs, and CICs) are System Engineering Laboratory (SEL) 32/55 & 32/75 computer systems. The NBP processor is a Control Data Corporation (CDC) Cyber 840. Figure 1 depicts the original system configuration for the TIPS. See Reference 1 for additional technical details regarding the original TIPS configuration. Figure 2 depicts current TIPS configuration after some ten years of operation. The major differences are that the TIPS has been re-configured from a fully integrated and tightly coupled network of 21 processors supporting eight telemetry streams, to three identical configurations of two TPPS, one MSC, one CIC and two QLDAs that can support two telemetry streams each and one configuration of two TPPS and one MSC which supports the Range Safety function in a stand alone fashion. See Reference 4 for additional details concerning the current TIPS configuration. The Range Safety system is connected to one of the other configurations CIC processors for control functions. This reconfiguration was accomplished to off-set problems encountered in the communications between the various system elements as a result of the tightly coupled network in the original design. The particular problems encountered and their solution will be discussed later in this section. Let us now review the four significant development efforts of the TIPS which are relevant to the technical discussion of this paper. These are:

1. Fast Multiplexer System (FMS): The FMS was developed for the TIPS by SEL to provide a capability to ingest high rate telemetry data into the TPP from the Telemetry Front End (TFE) equipment for processing, digital storage, and display without swamping the internal TPP SEL BUS with the high rate data. The FMS provided and still provides the TIPS with the functional capability to: (1) select the incoming TM data by ID (both compressed and raw TN data) and direct its transmission to the proper subsystem of the TIPS for processing, storage, and/or display. This selection is dynamic and can be modified by the TIPS system during the real-time operation; and (2) at the output side of the FMS to insert the TM data in the proper (and predefined by TM sample ID) memory location of the TPP, utilizing a

true DMA transfer into non-contiguous memory space. With a 26.6 MB per second bus bandwidth the FMS was more than able to provide the TIPS with the capability to ingest TM data from a 5 Mbps stream. Figure 3 depicts a normal TPP configuration for the TIPS FMS. See Reference 2 for additional technical details concerning the FMS.

2. Shared Memory: The TIPS system implementation included one of the first uses of shared memory between real-time telemetry data processors. The FMS was interfaced to the TPP through a separate shared memory bus. This method of interface was selected as it permitted the DMA transfer of TM data from the FMS simultaneously with TPP processing and reading/writing to its own private memory. The TPP shared memory bus also had a third DMA memory interface to the MSC that permitted all TM data that was to be written to the mission history file to be accessed and extracted by the MSC without TPP intervention. Although this implementation of shared memory has worked very well over the past ten years, it does have some disadvantages. The first of these is distance between processors. The adjacent memory connections can be no further apart than a cable length of twenty feet. Since this included vertical as well as horizontal cable runs, this forces the two processors to be in close proximity to each other. The shared memory was synchronous and required very close clock tolerances to operate at all. As a matter of fact the computer manufacturer (SEL) never could solve the problem of the small tolerances for the memory clocks, and the development contractor (SDC) had to solve it for TIPS by judicious usage of wire cutters to trim the clock connections and make the shared memory system operate. The other disadvantage that this shared memory implementation encountered was that the maximum memory size for the SEL 32/55 was only 1MB, and the shared memory had to fit within that limit as additive to the system private memory. For the original TIPS implementation each TPP was configured with 256 KB of private memory and 128 KB of shared memory. All TIPS processors had less than the maximum memory (1 MB).

3. Distributed Operating System (DOS): The only real-time Operating System available for the SEL 32/55 at the time of the TIPS development was the Real Time Monitor (RTM) from SEL. RTM was operable only on a single processor system and would have been difficult to modify to meet the TIPS requirements for a multi-processing environment, SDC developed the DOS to fill this need for the TIPS. The major disadvantage of the DOS is that it is a one-of-a-kind operating system, it will work only in the TIPS environment. Due to this feature it becomes very unwieldy and costly to maintain and no standard software packages are transferable to the real-time TIPS system. Also since it was developed and tailored specifically for the TIPS all changes to the TIPS must operate within the DOS parameters or the DOS must be changed, not a very attractive alternative.

4. Tightly Coupled Distributed Network (TCDN): The TIPS was originally designed and developed as a tightly coupled distributed network. This was done for a number of different reasons, the two main reasons being: (1) to have positive control of all TM data as it was flowing through the TIPS system; and (2) to reduce the number of operator positions necessary to control all aspects of the TIPS. The TIPS was designed as a fully integrated system with six Telemetry PreProcessors (TPPs), expandable to twelve TPPs, and six Quick Look Display Area Processors (QLDAs), expandable to nine QLDAs. Any TPP being able to process TM data and forward it to any QLDA for display. Since there were varying speed communication channels between all the various elements of the TIPS it was felt that only a tightly coupled network could maintain positive control of the data. As indicated above, the TIPS configuration was modified during the early 1980's to overcome some of the communication problems encountered by the tightly coupled distributed network.

LESSONS LEARNED: The major lessons learned from the design, development, and operation of the TIPS system were:

1. That the TIPS system was extremely through-put limited. Even though the TFE and the FMS were able to pre-process and place TM data samples in TIPS processor memory (DMA) at the very high rates required to support a 5 Mbps TM stream, the processors due to BUS bandwidth (26.6 MBps) and CPU power (0.5 MIPS) could not reliably process the TM data at rates much greater than 1.6 Mbps.
2. That the TIPS system processors were extremely memory limited. The SEL 32/55 processors were limited to 1 MB of memory, this included both main and shared memory.
3. Communications between all elements of the TIPS system were not tuned. The problem of being able to ingest the high rate TM data was solved by utilizing the FMS, the ability to move processed TM data rapidly and efficiently between the TPP and the other TIPS processors, especially the QLDA's, was limited by the capabilities of the SEL communications links.
4. The TIPS operating system (DOS), as a one of a kind operating system, increases significantly the software support requirements for the system.

INTEGRATED FLIGHT DATA PROCESSING SYSTEM (IFDAPS)

The Integrated Flight Data Processing System (IFDAPS) was designed and developed for the US Air Force Flight Test Center (AFFTC) at Edwards AFB, CA by Computer Sciences Corporation (CSC) during the early to mid 1980s to support aircraft and aircraft weapons system testing. The IFDAPS was an evolutionary design derivative of the TIPS system. We applied engineering solutions to the design shortfalls of the

TIPS to improve the overall system capability to ingest, process, store, and display real-time telemetry data. Reference 3 provides additional details regarding the design of the IFDAPS system. Figure 4 portrays the most common telemetry data processing element of the IFDAPS.

DISCUSSION: IFDAPS used the same basic building blocks as the TIPS, although in some cases the blocks were enhanced due to evolving technology, Aydin Monitor (AMS) TFEs were used to pre-process the serial TM data into a parallel digital format. The TFEs were almost a duplicate of that used by the TIPS for PCM and PAN/PDM, while the FM subsystem was totally redesigned to fully support the AFFTC mission requirements. FMSs and Shared Memory systems were used to transfer the TM data from the TFES to the first of the real-time processors, the Front End Processors (FEPs). The Shared Memory system had been upgraded from a synchronous to an asynchronous system to reduce timing problems and to permit greater distances between the IFDAPS processors.

The IFDAPS FEP processors were the upgraded Gould evolution from the SEL 32/55 to the Gould Systems 32/7780, which had almost twice the CPU power of the SEL 32/55. The Gould 32/7780 still had a memory limitation of 1 MB, however, each FEP was fully configured with the maximum memory. The IFDAPS is configured into tri-processor subsets to support individual missions. This memory limitation continues to place constraints on the systems capability to meet all real-time requirements. Figure 4 provides a pictorial display of that tri-processor configuration. By utilizing the shared memory between the three processors, each processor could perform the processing steps allocated to it from the full range of all processing to be accomplished for that mission. Figure 5 portrays the entire IFDAPS system architecture.

To improve communications capability between the various elements of the IFDAPS system, a standard HyperChannel network from Network System Corporation was provided. See Figure 6 of Reference 3 for a pictoral display of the IFDAPS communications/data network. The initial design for the IFDAPS system called for the HyperChannel protocols and drivers to be rewritten to permit small message blocks of real-time data to be passed on the network efficiently. However, the design was changed during the development and the implementation was made using the standard NSC HyperChannel hardware drivers and protocols and the standard NSC NETEX software. The result of this was that the HyperChannel network was never able to meet the data transfer requirements for the non real-time data let alone the real-time data.

The Gould standard real-time operating system, MPX-32, was provided as the operating system for the IFDAPS. This was a good decision in the early 1980's, but as more and more UNIX operating systems became available and the number of third party applications programs available on UNIX operating systems grew, MPX-32 became less attractive as there was no convenient methodology to transition the third party programs over to the IFDAPS. It should also be pointed out that very few real-time UNIX operating systems exist yet today.

LESSONS LEARNED: The major lessons learned during the design, development, implementation, and operation of the IFDAPS were:

1. Although the average CPU power was almost doubled and the memory size per processor was still 1 MB for the IFDAPS system, like it's predecessor it is still CPU and BUS bandwidth limited. The IFDAPS can process TM data streams at the 5 Mbps rate but has limited capability beyond that.
2. Memory capacity was still a major limitation to the full utilization of the IFDAPS.
3. Utilizing a Loosely Coupled Network for the IFDAPS improved the overall communications efficiency for the system, but standard network systems such as the HyperChannel are not really suited to real-time transfer of data without major rework of their software and protocols.
4. Shared Memory, while initially used to support the high-speed transfer of TM data into the data processing elements of the real-time system, has now become the central element in the system that ties together sufficient processors to perform the required processing.

TEST SUPPORT FACILITY-INTEGRATED FLIGHT DATA PROCESSING SYSTEM (TSF-IFDAPS)

The Test Support Facility-Integrated Flight Data Processing System (TSF-IFDAPS) was designed and developed for the US Air Force Flight Test Center (AFFTC) at Edwards AFB, CA by Computer Sciences Corporation (CSC) during the mid to late 1980s to support aircraft and aircraft weapons system testing. The TSF-IFDAPS was an evolutionary design derivative of the IFDAPS system. We applied engineering solutions to the design shortfalls of the IFDAPS to improve the overall system capability to ingest, process, store, and display real-time telemetry data. Reference 5 provides additional details regarding the design of the IFDAPS system. Figures 6 and 7 portray the evolving design of the IFDAPS and the TSF-IFDAPS systems.

DISCUSSION: The Telemetry Front-End Equipment (TFS) portion of the TSF-IFDAPS was completely redesigned from that used for the TIPS and the IFDAPS.

Aydin Monitor Systems (AMS) had made some evolutionary enhancements to their product line to provide better performance for more complex TM data stream structures. The AMS System 2000 (S2K) (See Figure 7) was selected as the TFE system that best matched the TSF-IFDAPS PCM telemetry requirements. The original TM requirements for the TSF-IFDAPS specified that each of the three identical TFE systems must be able to simultaneously handle three (10 Mbps) PCM streams, and three FM/FM streams (72 FM channels). The system was to be configured to support a total of 25,000 separate TM word ID's. The final design for the TSF-IFDAPS TFE was reduced in capability (2 PCM streams of 5Mbps and 2 FM/FM streams of 36 channels each with a system total of 8000 ID's) due to the internal processing limitations of the TFE and the Acquisition Processor. Table I provides additional details regarding the TIPS, IFDAPS, and the TSF-IFDAPS telemetry system capabilities.

The data processing element of each of the three identical Flight Monitoring Subsystems (FMS1, FMS2, and FMS3 of Figure 6 and should not be confused with the Fast Multiplexer System (FMS) described for the TIPS)) for the TSF-IFDAPS is composed of three Gould computer systems. The Acquisition Processor (ACQ1), a Gould 32/9780, a History Recording Processor (HIS1), a Gould 32/6705, and a Display Processor (DPS1), another Gould 32/9780. The Acquisition Processor (ACQ1) also has a Floating Point Systems 5310 Array Processor attached to it. The three Gould systems are coupled together via a Gould Reflective (shared) Memory System (RMS). The RMS contains 4 NH of memory. The RMS has not performed as expected, based on its specifications. The Shared Memory System used on the IFDAPS would have probably worked much better. The Gould 32/9780 and 32/6705 minicomputer systems could each be configured with a total of 16 MB of memory, including the RKS memory. Each 32/9780 has a compute power rating of 9 MIPS, the 32/6705 is rated at 5 MIPS and the FPS 5310 has a power rating of 30 MFLOPS. The use of the FPS 5310 on the TSF-IFDAPS is one of the first uses of array or vector processing for a real-time telemetry data processing system. Prior to the TSF-IFDAPS most systems treated TM data as only being scalar data.

HyperChannel and HyperBus communications networks from NSC were utilized by the TSF-IFDAPS for interprocessor communications of non-real-time files and data. No real-time data flow was to be accommodated by these networks.

The Gould MPX-32 operating systems was used for the TSF-IFDAPS.

LESSONS LEARNED: The lessons learned during the design, development, implementation, and operation of the TSF-IFDAPS were:

1. Although there was a significant increase in the CPU power and the total memory size for the tri-processor element of the TSF-IFDAPS the TM data requirements still were more than the system could process and had to be significantly reduced to meet processing capabilities.
2. Shared Memory still remained a central element in the system to tie together sufficient processing power to process the real-time data.
3. Distributed processing adds a significant software overhead to maintain the status and control of all of the systems elements.
4. Vector processing of real-time TM data will continue to grow and expand.
5. Telemetry system downlink rates and number of ID's will continue to grow, especially due to the increased emphasis to be placed on Test and Evaluation by the Services as production budgets for military hardware continue to decrease in the future.

SUMMARY: The design, development, integration, and operation of the three major test range real-time telemetry processing systems described above, TIPS, IFDAPS, and TSF-IFDAPS, points out that there is a serious shortfall in the real-time data processing capabilities required to stay up with the increasing capabilities of the on-board telemetry systems. These shortfalls in capability are spread between both hardware and software. The major hardware deficiencies are:

1. Insufficient CPU processing power to adequately process the real-time telemetry data.
2. Processor and I/O system bandwidths insufficient to ingest, process, store and display the quantity of real-time data required to meet user requirements.
3. Insufficient memory capacity to meet real-time requirements.
4. Significant reliability decreases due as a result of having to attach multiple CPU’s together via shared memory to apply sufficient processing power to meet the total processing requirements.

The major software deficiencies are:

1. No industry standard real-time operating system, such as UNIX or POSIX, which would permit application software portability between CPU’s.
2. Lack of higher order language compilers that have been optimized for real-time applications.
3. Limited availability of standard Data Base Management Systems for use with real-time Systems.

A major real-time telemetry processing system deficiency which has not been included above is the inability of the existing real-time telemetry processing systems to adequately process vector data. The TSF-IFDAPS is the only one of the above three systems which has a vector processing capability. Although most telemetry system analysts when asked, will state that they do not have any vector processing requirements as their data is strictly scalar, the authors firmly believe that Test and Evaluation community will significantly increase its requirements for additional real-time vector processing capability in the near future.

It should also be noted that in all of the above examples of current real-time telemetry data processing systems (TIPS, IFDAPS, and TSF-IFDAPS), they each have one or more CDC Cyber systems providing non-real-time batch processing support.

LARGE SCALE REAL-TIME TELEMETRY PROCESSING SYSTEMS

Can Large Scale SuperComputers or MiniSuperComputers be effectively utilized to process real-time telemetry data? Before we can answer that question, we should take a look at the structure of telemetry data words, what the downlink rates are in the future for on-board telemetry systems and make some predictions regarding the real-tine data processing capabilities required to meet the requirements of these future on-board systems. The material presented in References 6, 7, 8, and 9 provided excellent background data to the authors in preparing the following section of this paper.

TELEMETRY WORD STRUCTURE: The telemetry word that is input to the computer for real-time processing normally consists of the following four elements:

1. The telemetry data itself. This typically ranges from 1 - 16 bits in length.
2. The Identification (ID) Tag. Normally from 12 - 16 bits.
3. The time of occurrence for that word. Up to 16 bits.
4. Processing and Limit Flags. Up to 16 bits.

Since the average telemetry word is only 8 bits in length this represents an eight to one expansion from the raw data sample to the digitized telemetry word sample (64 bits) that is input to the host computer for real-time processing. If the host computer is a 32 bit machine, (as most real-time computers are today), this requires a double word fetch and store for all processing operations. The following paragraph presents an example of the data ingest requirements for a typical (current) TM system and what processing requirements that can generate.

TM INGEST EXAMPLE: A 5Mbps telemetry downlink stream with 4000 ID's would consist of approximately 500,000 samples per second (sps) (5Mbps/10 bits per sample). Ten bits per sample were used for this example to simplify the mathematics and to provide for downlink synchronization and control. If the system under consideration is processing, displaying, and storing raw, compressed, engineering unit converted (EUC), and derived TM data this could result in over 625,000 sps being ingested in to the computer system (500,000 sps raw, 50,000 sps compressed (a ten to one compression ratio), 50,000 EUC sps, and 25,000 derived sps). A 625,000 sps ingest rate equates to 5 MBps (625K sps X 8 Bytes per sample) of data flow into the host computer. If the number of ID's are increased to say 8000 for this example, the compression ratio should be reduced to a 5 to 1 ratio to insure sufficient samples are received during the time period for plot stability. This could double the number of sps for compressed, EUC and derived data samples. Also as the number of ID's are increased the amount of main memory required to store the Current Value and Display Tables increase at the same rate. The TSF-IFDAPS, described in Reference 5, utilizes three Gould processors rated at a total of 23 MIPS to process two 5 Mbps TM streams with a combined total of 8000 ID's.

During the 1970's telemetry downlinks for real-time test range processing were typically in the 300K bits per second (bps) to 5Mbps, with from 1000 to 4000 different measurements (ID's). By the end of the 1980's these downlink telemetry streams had increased to 10 - 20 Mbps with over 10,000 ID's per stream. The USAF has been working on a 20 Mbps airborne system with an evolution to a 50 Mbps airborne system as a DOD standard. The DOD 50Mbps std airborne instrumentation system should be available by the mid 1990's. Due to the increased capabilities of todays on-board systems, more compact and higher capability airborne computer systems and more plentiful and higher capacity avionics bus structures for handling digital data, we predict that the DOD standard 50Mbps airborne telemetry systems will generate ingest rates to the ground station real-time data processor in excess of 90MBps for periods of peak data transfer. NASA currently has a 300 - 600 Mbps downlink requirement for supporting the Space Station. This NASA requirement will require serious consideration being given to the use of large scale supercomputers in support of the real-time processing of the downlinked data. Although this NASA requirement will be composed of multiple TM systems multiplexed together into a 150Mbps or greater downlink system, the composite still represents a significant integrated real-time processing load.

The Telemetry Processing System (TPS) being developed for the Pacific Missile Test Center (PMTC), is one of the last major Test Range Telemetry System upgrades to enter into the development arena during the 1980's had an initial requirement to support eight (8) 20 Mbps TM streams with 70K ID's. Upon industry review this requirement was decreased to eight (8) 10 Mbps TM streams with 65K ID's total for the system, since the capability to process that much data in real-time was not available in 1988. In the reduced mode of operation this system could generate a TM ingest rate of approximately 80 MBps. The CPU processing power needed to satisfy the reduced requirements of this system is between 100 and 120 MIPS.

FUTURE DOWNLINK TELEMETRY SYSTEM CHARACTERISTICS:
The authors have also conducted a separate review of the expansion of the on-board telemetry system characteristics. The results of this review are fully described in reference 10 and are summarized below. This summary is somewhat biased, as the authors have tried to concentrate on on-board telemetry systems which are of sufficient high capacity that existing systems such as the TIPS, IFDAPS, TSF-IFDAPS, and TPS cannot process the digitized data after it has been decommutated and converted to a parallel digital stream. Existing systems, such as TIPS, IFDAPS, and TSF-IFDAPS, etc. were designed to process single TM streams up to 5 Mbps (TIPS and IFDAPS) and 10Mbps (TSF-IFDAPS) and are not capable of being upgraded to handle the increased rates. The summary below was extracted from TM systems that will be implemented in the 1990's and will require the real-time processing of data that has been downlinked at rates in excess of 20Mbps for a single stream and 10Mpbs for multiple streams for the same mission.

TEST ARTICLE	DOWNLINK RATE	NUMBER ID's	UPLINK RATE	TIME FRAME
Aircraft Weapons Sys				
Single Stream	10 Mbps	8000	N/A	Current
Multiple Streams	10 Mbps	65000	N/A	1995
Ground Weapons Systems				
Single Stream	5 Mbps	4000	N/A	Current
Multiple Streams	5 Mbps	8000	N/A	Current
Ballistic Missile Sys				
Single Stream	20 Mbps	16000	5 Mpbs	1995
Multiple Streams	20 Mbps	65000	10 Mbps	1995
Aircraft				
Single Stream	50 Mbps	65000	5 Mbps	1995
Multiple Streams	20 Mbps	65000	10 Mbps	1995
Space Systems				
Single Stream	50 Mbps	65000	20 Mbps	1995
Multiple Streams	150-300 Mbps	65000+	50 Mbps	1995

LARGE SCALE TELEMETRY PROCESSOR CHARACTERISTICS: Over the past 5 - 6 years the authors have reviewed future TM processing system requirements and have established the following list of key processor characteristics for high-speed real-time telemetry data processing systems:

- 64 BIT CPU, MEMORY, & ARCHITECTURE
- MINIMUM 250 MBps SYSTEM BUS
- INPUT/OUTPUT CHANNELS 100 MBps PLUS
- REAL-TIME OPERATING SYSTEM (UNIX/POSIX)
- SUPPORT STANDARD DATA BASE MANAGEMENT SYSTEMS
- PROVIDE LARGE-SCALE BATCH PROCESSING
- MINIMUM 256 MB MEMORY
- MULTIPLE CPU's WITH 25 - 40 MIPS EACH
- STANDARD I/O INTERFACES (VME, HIPPI, F/0)
- STANDARD COMPILERS (FORTRAN, ADA, C)
- SUPPORT VECTOR PROCESSING
- HAVE INHERENT EXPANSION/GROWTH

There are a number of reasons for each of the above characteristics, but for the sake of brevity, just a few of the more important ones will be addressed. No attempt has been made at assigning a priority to these characteristics, however the ones that are discussed would have to be considered of prime importance in selecting a real-time data processing system in the future.

The first characteristic in this lead group is the architecture of the system. As indicated in the discussion regarding the Telemetry Word Structure, above, by the end of the telemetry preprocessing by the TFE (bit synchronization, decommutation, serial-parallel conversion, identification tagging, time tagging, limit checks, processing flag identification, and preparation for transfer to the host processor) the individual telemetry data sample is now 64 bits in length. Current telemetry data processing systems treat this TM data sample as two 32 bit words. The majority of steps in the processing, storage, and display of that TM data now requires double word transfers and processing as the sample progresses through the real-time processing sequence. This adds a significant amount of latency and processing overhead to each and every TM data sample processed through the real-time system. The selection and use of systems that already have a full 64 bit architecture will significantly enhance the overall system throughout by eliminating dual reads/writes for all processed samples as compared to todays 32 bit architecture systems.

The next characteristic to be discussed is the capacity system bus for the real-time processor. As we have indicated above, future on-board TM systems will be downlinking TM data to the ground station at a rate of 50Mbps or greater. The 50 Mbps rate could generate a peak input load to the real-time processor in excess of 100 MBps. Given that the input data can be input directly to main memory (true DMA) at that peak rate, the real-time processing system would require a main system bus structure that has the capability and capacity to move that data internally for additional processing, storage and display at a rate at least 2 to 3 times greater than the input rate.

The third characteristic for discussion is the ability of the system to ingest the TM data into the system via it's Input/Output (I/O) system. This characteristic is intimately tied to the previous two characteristics discussed. Without the capability and capacity to meet the peak TM data input rates (100 MBps) mentioned earlier, the first two characteristics are somewhat academic for real-time processing. The system must be able to support a true DMA 100 MBps input rate. All too often in the past, when we have reviewed vendor literature or discussed their systems architecture with vendor personnel for what initially appears the be a very interesting real-time processing candidate, we determine that external I/O to all non-vendor devices is limited to RS-232, IEEE-488, or low rate Ethernet devices with no true DMA capability. At these peak rates, if one has to use CPU cycles and main Bus bandwidth to input the data to memory for further processing, the system will normally not be able to provide the required processing capacity.

Previous, successful real-time TM systems (TIPS, IFDAPS, TSF-IFDAPS, etc.) have made up for their shortfall in processing capability by interfacing multiple CPUs via shared or reflective memory where the TM data is stored. This process, although it has worked in a limited fashion in the past, has some serious overhead penalties which can not continue to be paid for the 50 Mbps or greater downlink systems of the future. The data processing industry has made some significant strides in the past few years in terms of producing multiple processing systems which are fully integrated and fully support multiprocessing. The system architecture and software have been designed to fully support multiple processors without the penalties we saw in the earlier TIPS and IFDAPS systems.

The remaining characteristics are of somewhat lesser importance, however they must be given full consideration when selecting the final system to perform the real-time processing for these future TM systems. Memory size is directly dependent on the number of ID's in the TM system, the types of real-time processing to be accomplished, scalar, vector, or both, the number and types of real-time displays, etc. We have determined the 16 MB of main memory is insufficient for a 10 Mbps TM stream. We feel fairly comfortable that a 64 MB memory would probably be more than sufficient for a 10 Mbps TM stream, but have not analyzed thoroughly the memory requirements for higher capacity TM systems. Programmer support is also a key consideration when selecting a TM system. The current systems, discussed above, all used unique operating systems. Although MPX was vendor supplied it had limited support utilities for the programming staff and little or no third party software availability. An operating system that is compliant with UNIX System V and has an efficient real-time kernel appears to be the leading OS today. The availability of third party software coupled with the availability of programmers who are UNIX and C experienced weights this in the favor of UNIX.

Another advantage of moving into the supercomputer arena is that most of the vendors in this line can and will support all required features. This reduces the number and different types of I/O interfaces, shared memories, reflective memories, etc. to the prime computer system supplier inventory and also improves the overall system reliability over having to have multiple processors interfaced to each other through shared/reflective memory. Another advantage that occurs because of using a supercomputer for the real-time portion of a TM data processing system is that each of the existing systems described also has a significant amount of non-real-time work associated with it. That work is currently performed today on large scale computers, such as the CDC Cyber line. By using a single processor to perform both the real-time and non-real-time portions of the TM systems data processing a significant cost savings could be realized.

There are a number of 64 bit computing machines available now that seem to offer promise with regard to fulfilling the above characteristics. A few that the authors have considered over the past 5 - 6 years are the Elxsi 6400, the CDC Cyber and the new CDC RISC 4000, and the CRAY Y-MP and Y-MP(EL) product lines. More recently the authors have taken a serious look at the evolving real-time capabilities of the CRAY Research, Inc product line. The principal characteristics which led us to consider the CRAY product line to be a serious contender for real-time telemetry data processing are:

CHARACTERISTIC	CRAY Y-MP	CRAY Y-MP(EL)
- 64 BIT CPU, MEMORY & ARCHITECTURE	YES	YES
- 256 MB MEMORY OR GREATER	YES	YES
- SYSTEM/CPU BUS GREATER THAN 250 MBps	YES	YES
- MULTIPLE CPU's	YES	YES
- I/O CHANNELS GREATER THAN 100 MBps	YES	YES
- STANDARD I/O INTERFACES	YES	YES
- SUPPORT VECTOR PROCESSING	YES	YES
- SUPPORT DBMS	YES	YES
- SUPPORT LARGE SCALE BATCH PROCESSING	YES	YES
- STANDARD COMPILERS	YES	YES
- OPERATING SYSTEM	UNICOS	UNICOS
- INHERENT EXPANSION/GROWTH	YES	YES & WITH Y-MP

CONCLUSIONS: That current real-time telemetry data processing systems are becoming saturated by the growth of the onboard systems and their capabilities. Having to operate on all TM data samples (64 bit TM words) in a 32 bit environment requires almost all operations to take twice the amount of time over a 64 bit system. The capabilities of input/output devices such as the Fast Multiplexer System and the Shared Memory Systems have enabled those current systems to reach well above their individual capabilities, but the much higher rate downlink systems being developed for the future will not be able to be accommodated by those existing technical capabilities. As system engineers and system designers we must start looking now for solutions to match the problems we will encounter in the next few years as the onboard systems continue to grow in capability and the Test & Evaluation community presses for more rigorous testing of weapons systems prior to start up of production.

ACKNOWLEDGMENTS: The authors wish to acknowledge the efforts of Dr. Jim Means, SRI, in his review and comments on this paper.

REFERENCES:

1. Van Dolsen, L.L., "TIPS - An Integrated Solution for Multi-Mission Telemetry", Proceedings of International Telemetering Conference, Vol XIV, International Foundation for Telemetering, Los Angeles, 1978, pp. 1051-1054.

2. Straehley, E.H., "TIPS Real Time Acquisition, Processing and Display Subsystem", Proceedings of International Telemetering Conference, Vol XIV, International Foundation for Telemetering, Los Angeles, 1978, pp. 1061-1069.

3. Shelly, L, and McCauley, R., "Accommodating Telemetry Data Acquisition Systems", Proceedings of the International Telemetering Conference, Vol XXIII, International Telemetering Foundation, San Diego, 1987, pp. 51-60,

4. Straehley, E.H., "The Case for Non Real-Time System Integration", Proceedings of the International Telemetering Conference, Vol XXIII, International Telemetering Foundation, San Diego, 1987, pp, 461-465.

5. Quesenberry, D, and Reed, G., "Telemetry Processing - Realtime to Postflight", Proceedings of the International Telemetering Conference, Vol XXIV, International Telemetering Foundation, Las Vegas, 1988, pp. 237-246.

6. Curtin, M.J., "Advances in Real Time Architecture for Telemetry Processing", Proceedings of the International Telemetering Conference, Vol XXV, International Telemetering Foundation, San Diego, 1989, pp. 675-682.

7. Kendall, R.A., "Unique Systems Through Reusable Software", Proceedings of the International Telemetering Conference, Vol XXV, International Telemetering Foundation, San Diego, 1989, pp, 583-589,

8. Waggener, W.N., "UNIX and The Real-Time Telemetry System", Proceedings of the International Telemetering Conference, Vol XXV, International Telemetering Foundation, San Diego, 1989, pp, 433-442.

9. Strock, O.J., "Data Distribution in the Telemetry Ground Station of the 1990's", Proceedings of the International Telemetering Conference , Vol XXV, International Telemetering Foundation, San Diego, 1989, pp. 535-542,

10. Dreibelbis, H. N., James, L. P., and Means, J. A., "Telemetry System Characteristics of the Future", 31 May 1991, report provided under contract to CRAY Research, Inc.

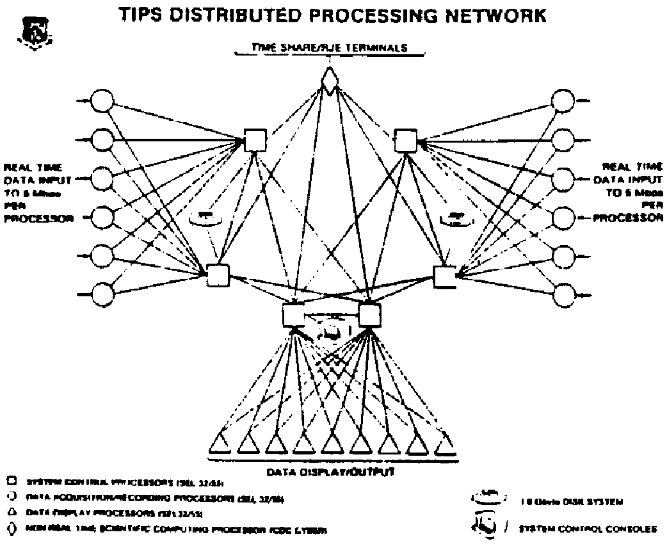
TIPS DISTRIBUTED PROCESSING NETWORK
TIME SHARE/RJE TERMINALS
REAL TIME DATA INPUT TO 5 Mbps PER PROCESSOR
REAL TIME DATA INPUT TO 5 Mbps PER PROCESSOR
DATA DISPLAY/OUTPUT
SYSTEM CONTROL PROCESSORS (SEL 32/55)
DATA ACQUISITION/RECORDING PROCESSORS (SEL 32/55)
DATA DISPLAY PROCESSORS (SEL 32/55)
NON REAL TIME SCIENTIFIC COMPUTING PROCESSOR (CDC CYBER)
1.6 Gbyte DISK SYSTEM
SYSTEM CONTROL CONSOLES

FIGURE 1. ORIGINAL TIPS CONFIGURATION

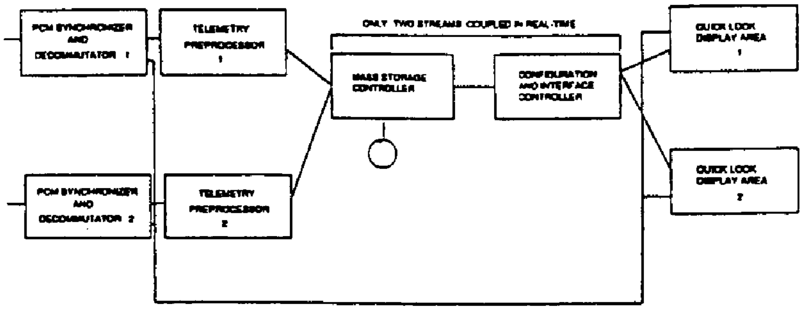
PCM SYNCHRONIZER AND DECOMMUTATOR 1
TELEMETRY PREPROCESSOR 1
ONLY TWO STREAMS COUPLED IN REAL-TIME
MASS STORAGE CONTROLLER
CONFIGURATION AND INTERFACE CONTROLLER
QUICK LOOK DISPLAY AREA 1
PCM SYNCHRONIZER AND DECOMMUTATOR 2
TELEMETRY PREPROCESSOR 2
QUICK LOOK DISPLAY AREA 2

FIGURE 2. CURRENT TIPS OPERATIONAL CONFIGURATION

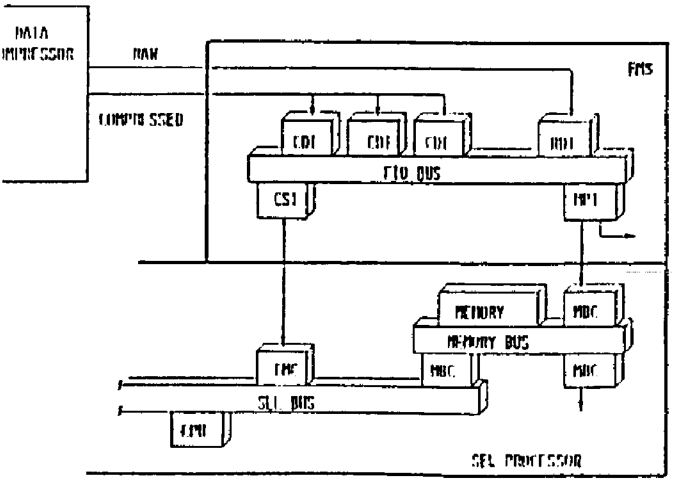

FIGURE 3. FAST MULTIPLEXER SYSTEM

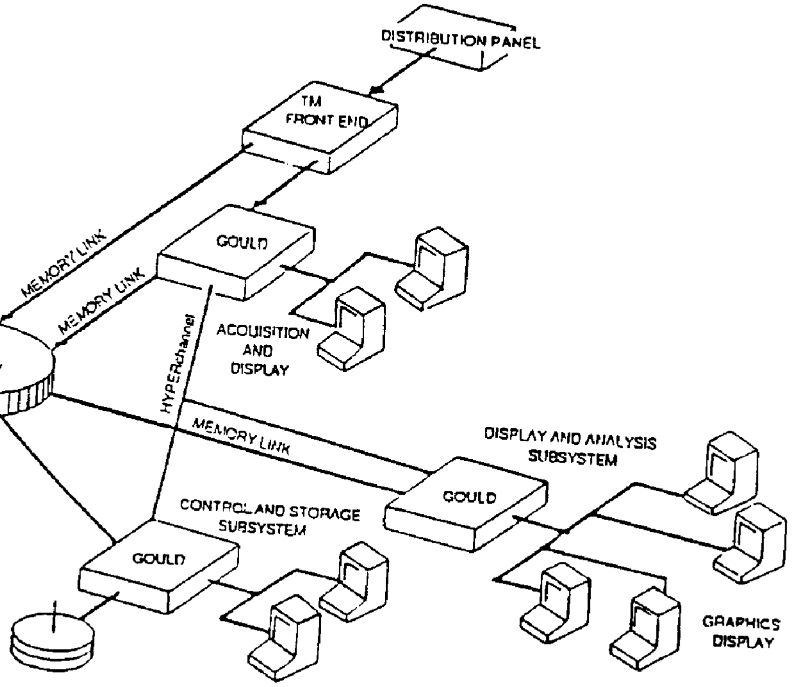

FIGURE 4. IFDAPS MISSION CONFIGURATION

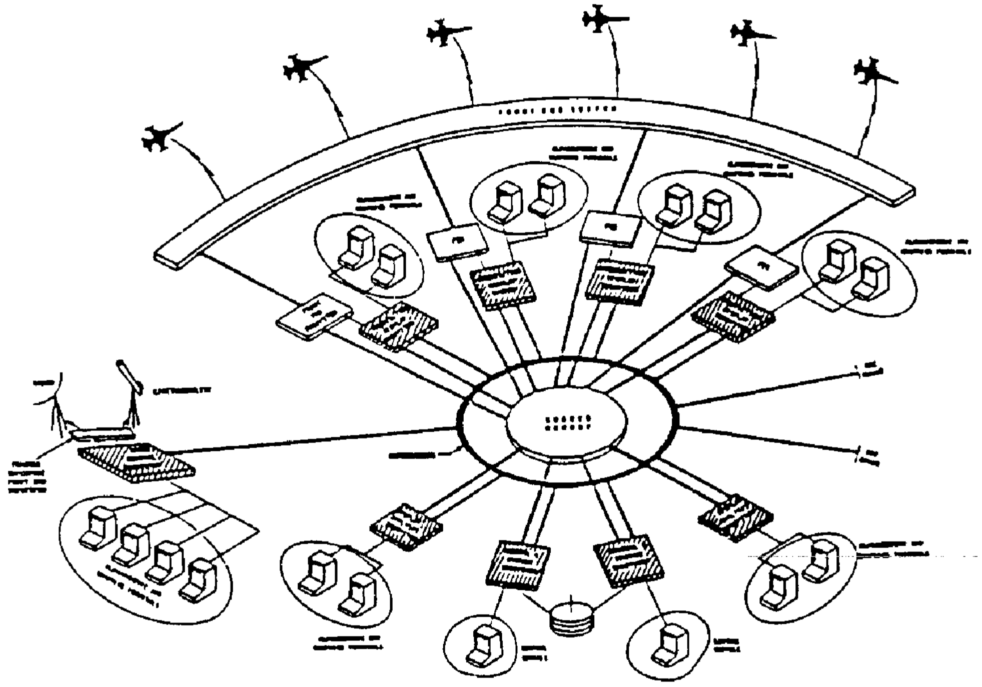

FIGURE 5. IFDAPS SYSTEM ARCHITECTURE

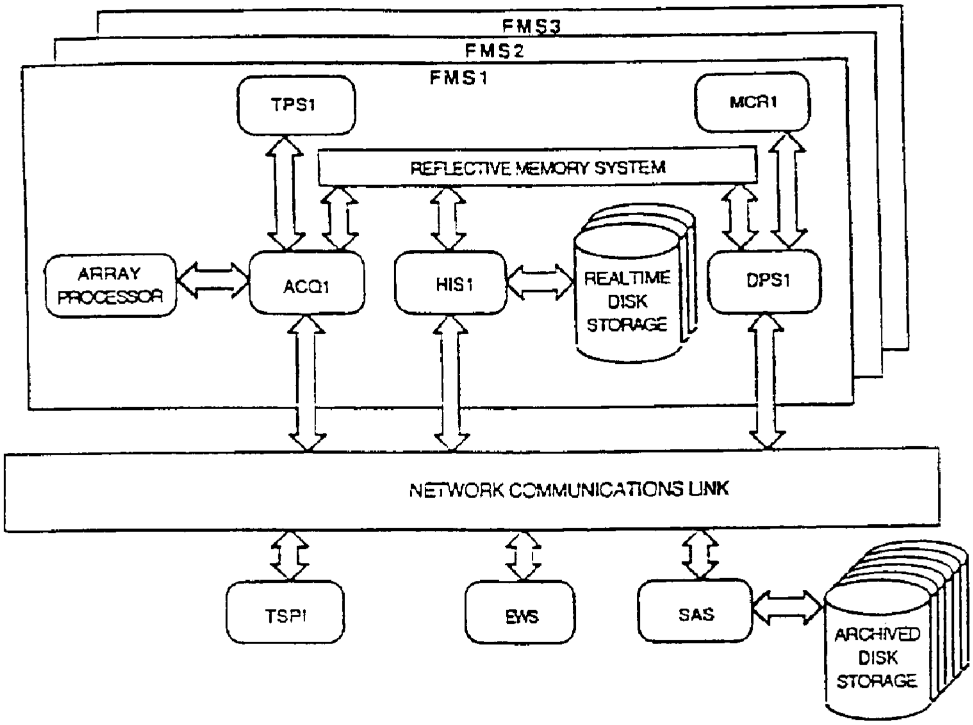

FIGURE 6. TSF-IFDAPS OVERVIEW

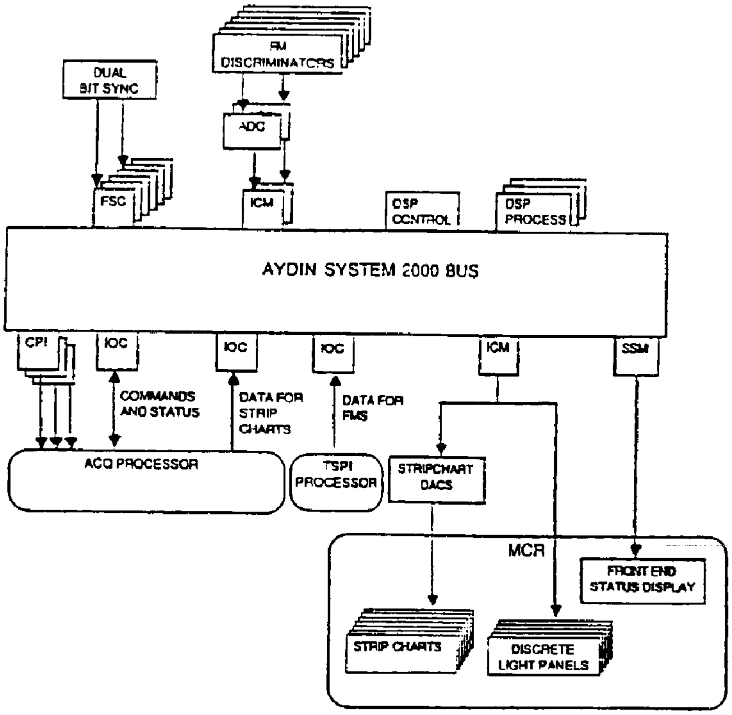

FIGURE 7. TSF-IFDAPS TM PROCESSING SYSTEM

	TIPS	BASELINE IFDAPS	TSF IFDAPS
Number of "streams"	6	5	3
For each stream			
FM Input	N/A	36 Channels 10K sps/channel aggregate 250 sps	72 Channels 200K sps/channel aggregrate 300 sps
PCM Input	1 Source 5 Mbit/	3 Sources 5 Mbit/second per source	2 Sources 1.2 Mbit/second per source
Engineering Unit Conversion	N/A	20K sps	100K sps
Recording	EU - N/A Raw - 20K	EU - 20K sps Raw - 35K sps	EU - 100K sps Raw - 200K sps
Maximum Number of Measurement	4096	4096	10,000
Graphics	0	6	11
Alphanumeric	6 plasma	6	1
Strip Chart Pens	256	128	128

* A "stream" is the subset of the system that is required to support one mission (i.e., one aircraft).

TABLE 1 COMPARISON OF TELEMETRY SYSTEM CAPABILITIES