COMMUNICATIONS PROCESSOR ARCHITECTURE:
A MICROPROCESSOR DESIGN

J.A. BOLES
L.R. CARLSON

SPERRY UNIVAC
SALT LAKE CITY, UTAH

ABSTRACT
The Communications Processor Architecture is implemented in the
Sperry Univac® DCP/40 (1), and is targeted towards high performance,
secure operation and network functionality. The architecture supports the utilization of large
scale integration, microprocessors and microprogramming in every module. In
addition to providing functionality and performance in the communications
environment, the DCP/40 provides excellent general purpose processing
capability.

Increased performance is achieved through advanced components and
functional distribution of tasks among the various processing elements.
Security and integrity is attained through address and queue access
protection. As a unique feature of the architecture, all I/O is treated in
common whether it drives peripheral devices or communications
lines. This technique simplifies software as well as hardware design.

INTRODUCTION
The history of communications and
computers has been one of constant
evolution. Each new product off-loaded
the off-loader. As the needs for
attachment of communications and the
overhead to handle communications grew,
multiplexors were attached and front-
end processors with their own
multiplexors were attached. Front-end
processors, remote concentrators and
cluster controllers have represented the constant migration of functionality away from the host.

Today it is possible to even further
off-load and become more independent from the host. It is no longer true that the host is the center of a
communications network; rather, in
many cases the host appears as a
terminal to the network. Now
communications networks can truly
stand alone and are transparent to the
host computers.

In 1976, Sperry Univac announced a
major communications product,
DCP/TELCON (2), and the Distributed
Communications Architecture (DCA),(3).
The architecture is the umbrella under which present communications
products and future communications
products, both hardware and software,
are being developed. DCA is supported
by the Universal Data Link Controller (UDLC) which is a bit-oriented
protocol. UDLC includes ADCCP, HDLC,
SDLC and other bit-oriented procedures that currently exist. The present
hardware product, the Distributed
Communications Processor (DCP),
operates under the network software
TELCON. This product is the initial
DCA offering. The goal of our new
product offering, the DCP/40, is to
augment and enhance DCP/TELCON
functionality. At the same time the
functionality is enhanced, the DCP/40
product also ensures compatibility with the present TELCON software base.

SYSTEM GOALS
The architecture of the DCP/40
consists of two modes: The DCP
compatible mode and the DCP primary
mode (4). The compatible mode is one
that protects investment in the
present TELCON network software by
executing the same software as the
present DCP hardware with increased
performance. The performance is two
times that of the DCP or a 1.0
microsecond average instruction
execution time. DCP/40 architecture
provides for a migration to primary
mode. Primary mode is specifically
targeted to provide DCA network
functionality and performance.
Primary mode provides functional
distribution of networking tasks so
that the maximum benefit can be
derived from new technology that has
become available. At the same time,
increased security and protection is
required. Therefore, primary mode
architecture has provided control and data structures that are visible to both the hardware and the software providing the best in security and protection. The hardware technology utilized in the DCP/40 has much in common with that of the 1100/80 (5). The Communications Processor (CP) and the Input Output Processor (IOP) utilize Emitter Coupled Logic (ECL) and Multi Layer Packaging (MLP). The DCP/40 utilizes a multiplicity of microprocessors, USARTS, and 16k X 1 storage chips. Large scale integration has allowed the hardware design to be extremely modular. The IOP is the same hardware for host channel interfaces, peripheral interfaces and communications line termination. The system performance of the DCP/40 in compatibility mode is two times that of the DCP. In primary mode, the performance of the DCP/40 increases to 2.4 times that of the DCP. The maximum capacity of storage that may be attached increases from 128k in compatibility mode to 2 million bytes in primary mode.

**PRIMARY MODE ATTRIBUTES**

A key attribute of the primary mode architecture is the increased security and integrity that is provided over what has been available in previous architectures. Security and integrity is achieved by a combination of hardware and software. In all cases the hardware is involved in recognizing and controlling data structures. The various data structures provided by queues and link areas may be addressed by software. Since the hardware architecture is aware of these structures, it is possible for the hardware to ensure that all control and data accesses by software are proper. In addition, the hardware itself has built-in error detection. Examples of this built-in error detection are data path parity, adder duplication and storage single error correction/double error detection. An equally important attribute of primary mode is performance. Performance is achieved in three principal ways. The architecture provides for a multitude of parallel operations. Each I/O activity is a process in its own right so that each active communication line has processing going on concurrently with every other active communications line. Pipeline processing is a second means of providing performance. Communications data is processed at each module of the DCP/40. For example, as the data enters the DCP/40, it is processed by a microprocessor at the line module. Next it is processed by a port processor executing on an IOP. Once it has entered storage, further processing takes place by the CP. This activity occurs in reverse as the data leaves the system through the IOP and line module. The architecture is storage centered and relies on table and queue control structures to provide the third means of increased performance. Throughout the design of the hardware and the software, the need for interrupts is minimized. In previous architectures it was necessary to interrupt the CP many times during the input of a message. Each interrupt incurred a significant amount of processing that is now unnecessary in primary mode. This is a result of providing for each port processor (PP) executing on an IOP the means to entirely handle the line discipline and protocol of the particular communications line. The PP may, for example, request retransmission of a message that was received in error. A PP never needs to interrupt the CP.

Communications is an extremely diverse environment. The need for instrumentation is critical. Therefore, the architecture and the hardware have built-in instrumentation capabilities. The instrumentation is controlled by software, but the actual instrumentation data is collected by the hardware and processed by the software. All I/O hardware in the DCP/40 is designed as commonly as possible. As a result of this commonality, one I/O processor is able to multiplex the full range of I/O devices. In primary mode one IOP can connect to communications line modules, peripheral line modules and host channel interface line modules. The program for these various peripheral devices is also very similar. The storage addressing extension that is provided by primary mode allows the hardware to address storage exactly the same as the DCP in compatible mode while providing for up to 16 million bytes of addressing in primary mode. This addressing algorithm has provided for not only compatibility with the present TELCON, but also, the ease of migrating from compatible mode to primary mode. In other words, some software modules that presently execute on the DCP can execute on enhanced mode with little or no modification.
The architecture provides for two primary processing elements. The communications processor which is similar to the traditional instruction processor or general purpose processor in our current DCP system. The dramatic addition to communications processing is the port processor which is dedicated to input/output. The port processor has its own instruction repertoire and is a processor in its own right. There is one port processor per communications interface. The port processor has a capability to do buffer allocation, message retransmission, error logging, and, in general, most of the protocol associated with the communications interface. A PP can execute two instruction streams for full duplex operation. Processes that are executed by the CP are the traditional dispatchable and pre-emptable processes that are executed by present architectures. The processes that are executed on the port processor are dedicated to an interface type; for example, a UDLC trunk line. Therefore, the PP process is unique and determined by the type of interface to which it is attached and is not pre-emptable. This type of functionally oriented I/O process allows for efficient software design and enhances the ability to distribute network functionality among the various microprocessors and microprogrammed processors in the DCP.

Storage management is a virtual addressing system (6) with 4K byte segments and 128 byte granules or subsegments. Hardware maintains usage counts on segments that are transient. This allows many processes to share the use of one segment. This shared segment may be rolled in and out of main storage when not in use. In addition, the virtual addressing system, when mapped properly, looks identical to the current DCP product addressing. The primary interface between CP processor and PP processor is the queue structure. Queues are first-in first-out (FIFO) data structures. For data structures that need to be managed in a non-PO manner, link areas are provided that can be managed according to the particular need of a specific procedure. In primary mode, events (interrupts) fall into two classes: exigent events and repressible events. The goal of primary mode is to minimize the need for exigent events. These are the events that would have caused a hard interrupt in the DCP.

For example, terminal polling can be handled entirely by processes that are executed by the PP without interrupting the CP.

FUNCTION RESIDENCY

In the DCP, many functions were done exclusively by the processor. The multiplexor could handle polling and data transfers and interrupt the processor for many of the other tasks such as buffer allocation, message status and retry. In primary mode, the role of which does the most work, I/O or processor, is almost reversed. This allows the system to handle many more transactions than the present DCP architecture. For example, the I/O now allocates its own buffer space, affects retry, acknowledges messages, and the like.

QUEUE EXAMPLE

Primary mode hardware and software work hand in hand through control and data structures. An example (Figure 1) is shown here of how a particular instruction addresses messages. To address a message control table (MCT), only one instruction is needed. The instruction together with control information provides the hardware with the necessary parameters to index into the interface control area with the line number to the queue list pointer (QLP). The queue list pointer points to a queue list (QL) which contains a queue pointer (QP). The terminal number is used by the hardware to index into the queue list to the right item. This item, the queue pointer (QP), in turn points to the queue (Q) that holds addresses for all the messages that are currently associated with a particular terminal. The
header information in this queue contains a field that indexes into the queue to the next message item. This item is a message control table pointer, (MCT P). The message control table pointer directs the hardware to the location of the message control table which contains descriptor parameters for a particular message plus history and control information. The descriptor data are segment descriptors which the hardware and software uses to access storage. This is a good example of how the hardware and software can work together to provide the attributes of primary mode such as performance and protection. One instruction can traverse these tables and queues to address a particular MCT.

A further example (Figure 2) of the data structure that is recognized by the hardware is the buffer pool. This is an example of a linked list. There are actually four pools. Two pools contain 4k byte segments, one pool contains zeroed segments and the other non-zeroed segments. Two more pools contain 128 byte granules, one non-zeroed and the other one zeroed. The hardware is aware of the link structure and therefore can remove and add segments or granules to these pools without the assistance of software routines. In fact, a background task of both the CP and PPs is to zero buffers on the non-zeroed pool and add them to the zeroed pool.

ADDRESS TRANSLATION

Virtual addressing on the DCP/40 is done in two ways. The virtual addressing hardware algorithm starts with a 17-bit virtual byte address (Figure 3). The five most significant bits point to one descriptor in a 32 word segment descriptor table. The descriptor contains access privilege information, segment length information, and the address of the segment itself. The segment address field is added to five next most significant bits of the virtual address (subsegment number) forming the 17 most significant bits of a 24-bit real address. The 7 least significant bits of the virtual address are then concatenated to the 17 most significant bits, forming a 24-bit real byte address. Subsegment addressing is similar, except that another level of descriptor addressing is inserted allowing for a non-contiguous 128 byte subsegment. The hardware contains two 32 word segment descriptor tables and two 32-bit word subsegment descriptors. The 32-bit subsegment descriptors are the two most recently used subsegment descriptors.
HARDWARE MODULES

The DCP/40 hardware system is composed of a number of modules. The modules can be classified into four categories: Processor, Storage, Input/Output Processors and Line Modules. The hardware modules execute CP processes. The second category is the storage module, which is the heart of the storage centered DCP/40 system. The third category is the Intelligent controller or the Input/Output Processor. This module executes PP processes. The fourth category is line modules. This category is the most diverse and contains the hardware modules that provide connections to host channels, peripheral controllers and communications lines.

The purpose of the storage port expander is to allow more than four requestors to be attached to any one storage bank. The maximum DCP/40 System has 17 requestors — one processor and 16 Input/Output processors. The minimum system has two requestors — one processor and one Input/Output processor. All the storage ports are identical, allowing for the maximum in flexibility. The system is so designed that it can be enhanced into a tightly coupled multi-processing system by the attachment of the second processor.

PROCESSOR MODULE

The processor has been designed using emitter coupled logic and multilayer packaging. It is fully microprogrammed, and the microcode resides in RAM and is, therefore, loaded from an integral diskette. The processor has a 16-bit data path and utilizes a 32-bit microinstruction. The effective rate of microinstruction execution is 65ns. Microinstruction execution is overlapped three deep (7). While one microinstruction is being fetched, the previous one is fetching operands, and the one before the previous one is storing results. The processor executes the virtual addressing scheme in hardware taking a 17-bit virtual byte address and forming a 24-bit absolute byte address. The processor has through checking by providing byte parity on the data paths and by providing other through checking schemes such as adder duplication.

PROCESSOR ARM

The Availability, Reliability, and Maintainability (ARM), attributes of the processor are quite extensive. The adder unit has been duplicated. All 16-bit data paths have two bits of parity. Control storage has byte parity. When a control storage parity error occurs, it is possible to automatically retry the microinstruction fetch. Parity prediction/check hardware has been designed for the micro F incrementor and the virtual address translator. Micro-interrupts are taken on non-recoverable hardware errors. There is an error status register that saves the type of hardware error that has occurred, plus there is a special error log interrupt which allows logging of recoverable hardware errors.
HIGH PERFORMANCE STORAGE MODULE

The storage module for the DCP/40 utilizes an NMOS 16k by 1 IC RAM. The access time of the chip is 150ns. The read access time of the storage module itself is 300ns. Storage is organized around a 32-bit word. Each storage module has its own controller, which has four ports serving four requestors, and has the necessary logic to do single error correction and double error detection. The storage bank has 128k byte modularity. The DCP/40 primary mode system can address up to four storage banks. This is equivalent to 2 million bytes. The processor module backpanel physically has space for one storage bank. Storage expansion above 512k requires an expansion cabinet with a storage expansion backpanel. It allows the addition of up to three more storage banks. Each of these banks can operate concurrently.

STORAGE ARM TECHNIQUES

Just as the processor has extensive hardware for ARM, so does the storage module. Each 32-bit data word has 7 check bits associated with it to provide for single error correction/double error detection. In addition, all of the data paths within the storage modules have byte parity. In addition to the normal read and write modes, there are diagnostic modes that allow a processor attached to a storage module to perform diagnostic routines on a particular storage module. Each module has its own error log for recording corrected single errors. Also, when a non-corrective error occurs, there is an error status register that may be read by a requestor.

INPUT/OUTPUT PROCESSOR

The design of the Input/Output Processor is very similar to that of the processor. The design utilizes emitter coupled logic and MLP packaging. It is a microprogrammed processor with an 8-bit data path and a 16-bit microinstruction. The IOP is fully microprogrammed and the microcode resides in RAM and is loaded from a integral diskette. It has an overlapped microinstruction execution unit similar to that of the processor. Three microinstructions are in various stages of execution resulting in an 80ns effective microinstruction execution time. The Input/Output Processor, as its name implies, is designed to handle I/O processes.

Towards this goal, the IOP consists of 16 port processors and one background processor. Each of the port processors can execute two instruction streams so that it has a full duplex capability. Therefore, the I/O processor is a 33 state machine. The IOP also has through checking similar to that of the processor; for example it has data path byte parity. The IOP addressing algorithm is the same as that on the processor.

Some of the address translation is controlled by the microprogram.

INPUT/OUTPUT PROCESSOR INTERFACES

The Input/Output Processor is the same piece of hardware whether it is controlling peripheral interfaces or communication interfaces on the DCP/40. Therefore, the various types of interfaces can be mixed on one IOP in primary mode. The type of interfaces can be grouped into three categories. The first group is the host channel adapter. Channel adapters serve for direct channel attachments to host computers. The second category of interfaces is for peripheral attachments such as the cartridge disk, magnetic tape and flexible diskette. There are two types of peripheral interfaces. One is the 16-bit parallel channel and the other is the 8-bit parallel channel. A third and most important category of interfaces is for communication line attachments. There are line modules that attach to all the required electrical interfaces and various communication protocols. Examples of some of these protocols are UI100 (Synchronous), Asynchronous, UDP, REMI, 1100 full duplex and BSC.

Most of the line modules that attach to the IOP ports contain microprocessors. A line module microprocessor helps do the particular function that the I/O port is assigned. The function of the particular line modules helps determine the type of microprocessor used. Low speed communications utilize the INTEL 8048 microprocessor, medium speed communications utilizes the ZILOG 280A microprocessor, and the high speed line module utilizes the AMD 2900 microprocessor bit slice. All the microprogrammed processors are implemented with loadable RAM control storage with the exception of the INTEL 8048. It is this microprocessor technology that has enabled the efficient implementation of the CPA.
SUMMARY

The DCP/40 is an exciting, innovative entry into the communications network arena. It utilizes advanced hardware and architectural concepts and it is specifically oriented towards network performance, security and functionality requirements. It provides a compatible mode while at the same time providing ease of migration from compatible mode to primary mode. It is the beginning of a family of DCP hardware products. These products complement the software product TELCON and allow TELCON to grow in terms of network functionality, performance and security.

REFERENCES

(1) Sperry Univac DCP/40 System Reference, Primary Mode, UP8720
(2) Sperry Univac TELCON System Description, UP8455, Revision 1
(3) Sperry Univac DCA System Description, UP8469
(4) Sperry Univac DCP/40 System Description, UP8695