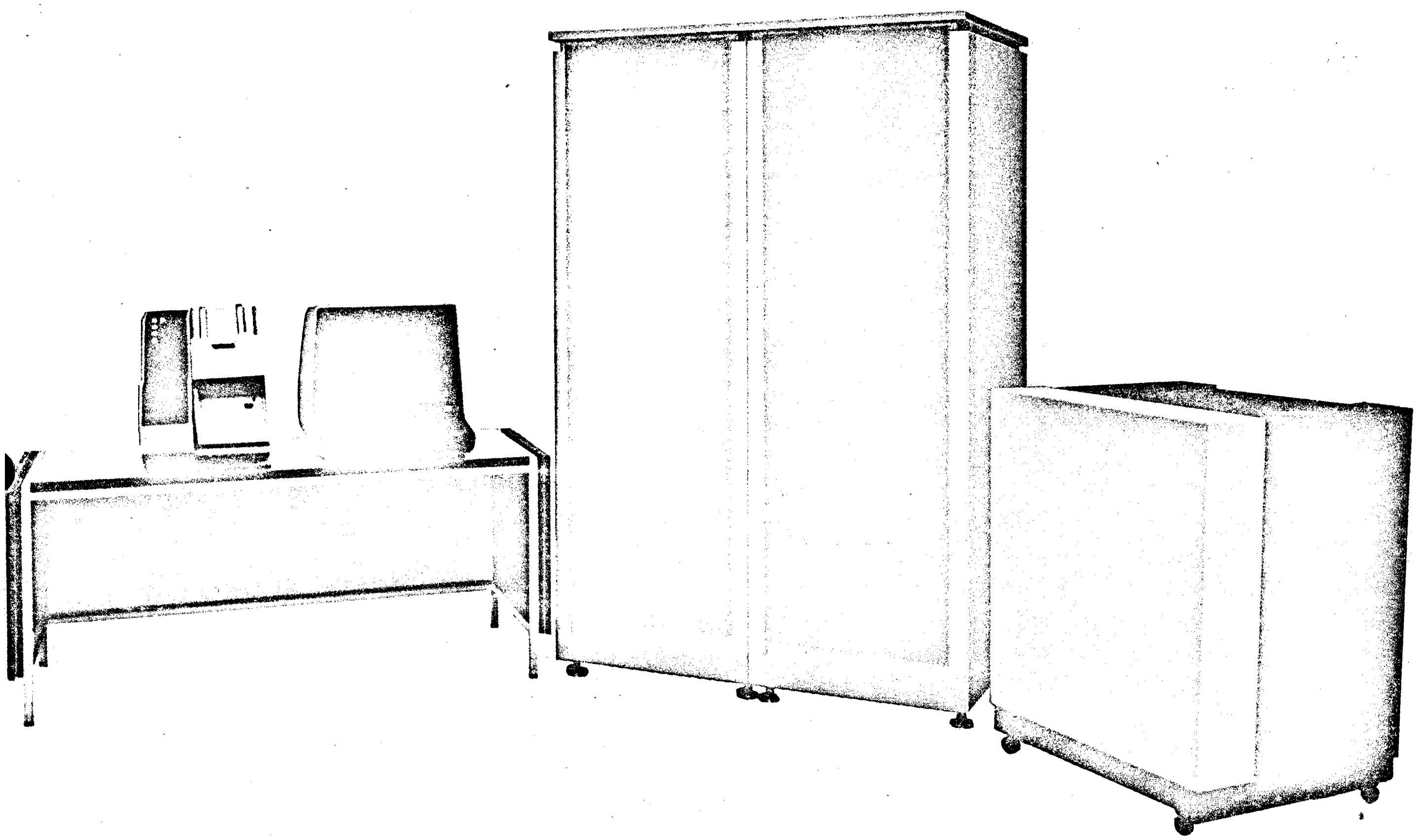
CONTROL DATA®
2550 SERIES HOST COMMUNICATION PROCESSOR



PRODUCT DESCRIPTION

TABLE OF CONTENTS

PREFACE

This document is intended to provide the reader with an overview of the latest CONTROL DATA CYBER 170 related data communications handling products and their interrelationship to CDC's computer network developments. This document is intended as a general treatment of the technical and functional details of the hardware and software as well as the products' relationships to, and interdependencies with the host computer operating system with which it will interface. Additional technical details regarding the hardware and software will be forthcoming in the form of Reference Manuals and General Information Documents.

The Network Communication System concept and the products discussed in this document represent a significant change in Control Data's approach toward the offering of data communications. With the CDC CYBER 170, the Network Operating System (NOS) and the Network Communication System (NCS), the task of information movement takes its place as a co-equal with the task of data processing. To the CDC salesman this means the necessity to better understand the data communications environment and these new CDC products. To the CDC CYBER system user, it means having the ability to plan and implement effective computer networks.

INTRODUCTION

Computer Network Structure

One of the primary objectives of users of a computer system is to *maximize the system's utilization.* This can be accomplished in various ways. Within recent years, technical advances and cost reductions in both computers and communications facilities have created a strong trend toward accomplishment of this objective by providing access to the computer power through the use of communications networks. As a result, an ever increasing number of companies are either buying data processing services or developing their own computer networks to take advantage of economies of scale and resource sharing. The relaxation of tariff restrictions and increased competition will cause further expansion of digital networks, microwave systems, and satellite communications facilities. This aspect of higher quality, lower cost communications will accelerate this trend toward computer networks.

A data communications network is a system of interface and switching devices which utilizes communications circuits to transport data. A network processing facility is a total information processing system consisting of computers and terminals interconnected by a data communications network. CDC has already had considerable experience in this area with its CYBERNET® Service and is committed to provide its system users with hardware and software that will facilitate development and operation of resource-sharing networks.

The primary goal of network processing is the provision of improved service at lower cost through economies of scale and resource sharing. A system consisting of distributed computers, terminals, and communications circuits can operate as a single information processing facility.

There are four major communications functions required in a computer network. One of these functions is always resident in the host computer(s) in the network while the other three may be combined within various configurations of a single set of hardware. Each set of hardware and communications function(s) is generally known as a communications "node." ("Node" is generally defined as an intermediate point within the network, between host computer and terminal, where specific functions are performed related to moving data between source and destination.)

The specific functions mentioned earlier are discussed below.

- **Network Access**

  The communication function which is always resident in the host processor is the software that interfaces between host resident user application programs and the external communications. It may be variously identified as: the operating system's data communications handler; or communications driver; or even as an access method. Within present Control Data CDC CYBER operating systems this function is handled by INTERCOM for the SCOPE Operating System and by TELecommunications EXecutive and E/I 200 for the KRONOS Operating System. Its job is to efficiently move data between user program and network communication system while using the minimum amount of host processor resources. A second function allows the user programs to operate with various types of terminals.

- **Host Interface**

  This function provides the interface between the host data processor and the communications network. On the network side, it must be capable of interfacing with the other elements of the communication system. On the host side, it must present and accept data in the form required for the most effective operation of the host data processor.

- **Network Switching**

  In complex networks, it may be necessary to provide points at which a decision may be made as to transmission paths between data source and destination and even possibly to determine what the destination is to be. This function performs no data processing, but merely acts as a relay and distribution point between other network elements.

- **Terminal Interface**

  This function provides the interface between the terminal devices and the remainder of the network elements. The terminals may have widely varying characteristics with respect to transmission mode, data rates, supervisory protocol, buffer sizes, code structure, etc.

Associated with the items above is another important function. This is the function of formatting the data so that it can be efficiently and reliably moved between communications nodes within the network.

CDC Network Objectives

The design of a computer network will vary according to the requirements of each user. A primary objective of Control Data's Network Communication System is to provide a system of such modularity that the basic functions described above may be performed singly or in any desired combination, by selection of appropriate elements of a single multi-purpose hardware and software system. Since communications networks are never static, flexibility, ease of expansion, and adaptability to new terminal types are mandatory features. Provision must be made for the enhancement of the basic system through addition of data handling and processing functions peculiar to each specific application, and such enhancement must be attainable within the capabilities of the user.

The accomplishment of this objective will provide the users of CDC systems and services with the capacity to implement a truly versatile resource-sharing network. Some of the benefits and advantages that will accrue to the user will include the following:

- Network interfaces to CDC and non-CDC computers
- Network monitor and control software
- More effective computer utilization
- Increased system availability
- Lower cost per connected terminal and computer
- Better utilization of communication facilities
- Greater freedom in selection of terminals
- Greater freedom in design and enhancement of networks
- Capacity to totally manage the network and all of its elements.

The implementation of a Network Communication System is no small task. It must encompass the meeting of the long range objectives and at the same time meet the user needs of today. Control Data will implement its Network Communication System in two distinct phases associated with the new CDC Network Operating System. The Network Operating System will initially incorporate communications handling using already developed and field-proven software modules which will provide a basis for the evolution of the Network Communication System.

The first phase, covered in this document, will include the following elements:

- Develop the common hardware modules to be employed throughout the Network Communication System (NCS).
- Develop the modular software that will perform the Host computer and terminal interface functions.

Included in the second phase implementation will be the following:

- Physically segregate the Host interface and terminal interface functions.
- Develop software for inter-node communications.
- Develop the Network Switching Function software.
- Develop final version of common Host computer access method software for Network Operating System.
- Add redundancy capability.

The Network Communication System which includes the 255X series Host Communications Processor and its associated Communications Control Program is the visible result of the first phase of this implementation plan. The remainder of this document will relate primarily to these products. However, it must not be forgotten that this is not CDC's final solution but rather the first step in the development of a truly viable computer network capability.

255X Series Host Communication Processor

The 255X series Host Communications Processor (HCP) is a combination of hardware modules sized to meet varying customer requirements — from the very small to the very large.
The HCP interfaces to CONTROL DATA CYBER 170, CYBER 70 or 6000 series computer systems. The Communication Control Program (CCP), resident in the Host Communication Processor, includes variants that allow intercommunication with either CONTROL DATA NETWORK OPERATING SYSTEM or SCOPE 3.4/INTERCOM 4.2.

A primary element of each system is a communications processor. It provides buffer storage for input and output data, and includes hardware interfaces to both the host processor and the communications multiplexing subsystem. The HCP also performs, under the control of software and firmware, many of the functions which are dependent upon the protocol (supervisory) characteristics of terminal devices and of other network elements. This, in turn, reduces the cost and variation requirements of hardware circuit adapters, and facilitates incorporation of new terminal types into the network.

The basic element of the multiplexing subsystem consists of two very high speed digital data multiplex loop circuits, one dedicated to the handling of input from terminals to the communications processor, the other to output from processor to terminals. This basic element is combined with other elements, such as processor interface, circuit interface and control, to form the total communication system.

The design of most multiplexing subsystems is based on a scanning technique, in which each line interface is periodically interrogated to determine whether action is required, such as removal of an input character from a buffer, or providing another character for output. This scan function must be performed for each circuit at a rate which cannot be less than once during each character time. Since this operation consumes time, the number of connected circuits and their speed has a direct bearing upon system capacity, regardless of the amount of traffic handled.

By contrast, the Control Data system is demand-driven. When action is required by a circuit interface, it will generate "demand service" request to the communications processor. Circuit connectability becomes a factor only in the specification of numbers of circuits that can be physically terminated and addressed. Consequently, system configuration is greatly simplified by the fact that it is largely governed by throughput volume.

Associated with the 255X series hardware is a software package identified as the Communication Control Program. This software accomplishes the overall functions of interfacing to the host computer and to a variety of terminals. The Communications Control Program is modularly structured to provide the capacity to meet future phases of Network Communication System implementation, yet it also takes maximum advantage of the hardware features of the Host Communications Processor. The software and hardware are designed to truly complement one another.

The initially announced configurations of the Host Communication Processor incorporate the functions of Host Interface and Terminal Interface as defined previously. These combined functions provide many of the features of a "classical" communications front-end system.

Subsequent planned announcements will include the implementation of the second phase of NCS and various other enhancements in such areas as: enhanced overall performance, handling of a broader range of terminals and speeds, and better utilization of redundancy capabilities.

It should be emphasized, however, that with the user programmability incorporated into this product set, the user has the capability to enhance his system very substantially while at the same time assuring his ability to use future versions of CDC software.

HOST COMMUNICATION PROCESSOR

The Host Communication Processor (HCP) is a new line of communications products to serve the data communications handling functions for the CDC CYBER 170, CYBER 70 and 6000 Series Computer Systems. The HCP design integrates state-of-the-art hardware with new software concepts to provide a cost effective, modular and highly flexible solution to the data communication requirement. The HCP Series is composed of a series of common hardware and software elements. This section of the Product Description will discuss each of these elements and how they are combined into the various hardware configurations.

Communications Processor

The Communication Processor (CP) is a 16-bit micro processor which operates in a highly parallel mode. The program is stored in main memory, a very high speed storage medium, while the micro code which dictates how the program instructions are to be executed is stored in micro memory. The micro code available with the CP implements an instruction set offering an expanded repertoire of instructions providing additional power for character and field manipulation, indexing and other communication-oriented processes.

Major Features

- Micro-programmed
- 24K - 128K of 16-bit main memory
- Main memory cycle time of 600 nanoseconds
- Micro instructions cycle time of 168 nanoseconds
- Powerful instruction repertoire
- Eight memory addressing modes
- Memory word and region protection
- Main memory parity detection
- Direct storage access
- High-speed I/O data transfer
- Modularity (CPU and Controllers on PC Boards for ease of handling)
- High reliability and easy maintainability (through state-of-the-art technology and advanced diagnostic capability)

Addressing Modes

The Communications Processor features eight addressing modes:

- Absolute
- Indirect
- Relative
- Relative Indirect
- Constant
- Storage
- Storage Indirect
- Field

This multiplicity of addressing modes permits extended flexibility of CP usage.

Macro Instruction Repertoire

The HCP incorporates the 1700 Series instruction set and some extended instructions not available before. The repertoire includes one, two and three word instructions and is flexible enough to increase programming efficiency. Instruction groups include the following:

- Transfer
- Logical
- Stop
- Shift
- Interrupt
- Generate Parity
- Character/Field Manipulation
- Execute Micro Code Sequence
- Arithmetic
- Jump
- Decision
- Input/Output
- Program Protect
- Loop Control

Some instructions are immediate (literal), resulting in a saving of operand storage space and execution time. Multi-word instructions, like Indirect Addressing, are a means of addressing locations which cannot be accessed directly.

Registers

The CP emulates a total of 12 registers. The eight traditional registers are used in the execution of the normal CDC 1700 instruction set while four general-purpose registers have been added to support the enhanced instruction set. There also exist three special-purpose registers used exclusively for machine control.

Register	Type	Function
A (16 bit)	Accumulator	Principal arithmetic register; interface during I/O.
Q (16 bit)	General	Auxiliary arithmetic register; peripheral address register during I/O operations.
P (15 or 16 bits)	Special Purpose	Program Address Register.
X (16 bit)	Special Purpose	Storage data register.
Y (16 bit)	Special Purpose	Address register; holds temporary results during address computation.
[illegible] (16 bit)	Special Purpose	Interrupt mask register.
B (16 bit)	Special Purpose	Breakpoint address register.
I (16 bit)	Index Register	
1, 2, 3, 4 (16 bit)	General	Indexing, accumulation and loop control registers.
LB, UB (16 bits)	Special Purpose	Lower and Upper bound registers for unprotected area.
MP (4 bits)	Special Purpose	Memory page register.

Program Protection

The CP offers two modes of protection from the damage which may be done by programs accessing memory outside of their own region. The traditional word level protection of the 1700 Series is featured. This allows individual words to be declared protected by setting a bit in memory associated with that word. A second means of protection is also employed using upper and lower bounds to define an unprotected region. This has the same effect as word protection, except that a large unprotected area can be defined more quickly.

Interrupt System

The HCP firmware emulates the 16 levels of vectored interrupt featured on the 1700 Series Computers. This system consists of 15 levels of external interrupt and one internal interrupt.

Certain conditions such as an incorrect instruction, a memory parity error, or a power failure will generate an internal interrupt. External interrupts occur when a computer peripheral device has finished an I/O operation or requires attention.

The strength of the interrupt scheme of the CP is its ability to handle a significant number of interrupts in a flexible and efficient manner. Many computers do not provide for as many discrete interrupts as the CP.

Auxiliary Operations

Power Failure/Auto Restart

The CP makes recovery from power failure possible through the use of power failure detection hardware and software. When power fails, an interrupt will allow the programmed transfer of key program data from registers to main memory for non-volatile preservation until power is re-applied.

Breakpoint

The breakpoint facility, implemented in micro code, insures the termination of a program at a predetermined location in memory. Breakpoint is a most useful aid to program debugging. This feature is activated via the CP console device.

Autoload

The re-loading of programs is provided by this feature, implemented in micro code sequence. The operator can down-line load the CP from associated host computer. Alternatively, any readable device could be used as a loading media with the appropriate firmware control sequence. The system will support autoload via three schemes:

- Down-line via Coupler
- Down-line via Trunk
- Via Cassette (maintenance)

Maintenance Console

The CP features a maintenance console panel which is also useable by programmers and provides for:

- Display of Register Contents
- Display of Memory Locations
- Operating Switches
- Indicators

The above functions can also be implemented via the Communication Console.

Communication Processor Main Memory Systems

The HCP features high-speed memory with proven reliability of design. The CP memory options all feature the use of 18-bit storage words comprised of:

- 16 data bits
- 1 parity bit
- 1 program protect bit

The memory features a 600-nanosecond full cycle time in each of three configurations:

- 32K MEMORY
  Storage capacity is available in 8K increments up to 32K.
- 65K MEMORY
  Storage capacity is available in 8K increments up to 65K. The micro code in this version of the CP allows full direct access of 65K memory, but limits the use of indirect addressing to one level only.
- LARGE MEMORY
  Storage capacity is available to 128K words in 8K increments. This memory is used with the micro code which allows full use of the total storage area. Features of this memory are:
  - Virtual memory space is 128K at any instant of time. 65K for instructions; 65K for data.
  - 18-bit addressing by use of 16-bit word address and page selection register (2 bits).
  - Instruction/Data register (1 bit) determines which half of virtual space is accessed.

Input/Output Capability

The CP features both a Direct Memory Access (DMA) Interface to peripherals and a program interface to peripherals. Using the Interrupt Data Channel (IDC) Interface, the program controls the data transfer, employing the Q Register to address the desired peripheral and the A Register to transfer Data, Commands, or Status between CPU and peripheral.

The DMA Channel permits direct transfer of data between the peripherals and main memory, bypassing the CPU main registers. The DMA Channel can handle far more data than the IDC with less CPU supervision.

DIRECT MEMORY ACCESS	Direct access of peripherals to main memory
Maximum transfer rate	1,600,000 words/second

NON-BUFFERED INTERRUPT DATA CHANNEL

A Register	Transfers data in and out; transmits function codes (commands); receives status information
Q Register	Transmits address information to peripheral; transmits control signals
Transfer rate	160,000 words/second

CHANNEL COUPLER

The Channel Coupler provides a low-cost communications link between a CDC CYBER host computer and the Communications Processor (CP) in a Host Communications Processor.

The primary function of the coupler is to pass 8-bit data characters directly between the computer memories with minimum software supervision.

The coupler provides the means for transfers between PPU and CP at PPU Channel transfer rate. Supervision is via both PPU and CP Software Commands and control words in the CP buffer. Buffer chaining in the CP core is provided.

All coupler operations are initiated by PPU function commands and/or CP I/O commands. The transfer of data between computers also requires additional PPU I/O instructions and control words in CP memory.

The expanded HCP Systems (2550-2, 2552-1) allow the optional addition of a second 2558-1 Channel Coupler to the configuration, thus allowing various inter-connection options, such as 1 HCP with two hosts, 2 connections with one host, etc.

MULTIPLEX SUBSYSTEM

The feature setting the Host Communication Processor apart from traditional communication processors is the manner in which the communication processing takes place. The communication processing of the HCP is demand-driven.

Communication equipment has traditionally featured scanning techniques of some type to provide adequate servicing of the communication lines. These scanning techniques required frequent intervention by the processor, even when no activity was taking place on the communication channels. This overhead function meant that available processing time could not be optimized.

Control Data has designed the HCP to bypass this traditional problem by use of an advanced concept called "demand driven" multiplexing. Utilizing this concept, processor intervention is required only when data "blocks" require servicing. The hardware and firmware associated with this concept, within the HCP, is known as the "Multiplex Loop" subsystem. Detailed operation of the Multiplex Loop is discussed in the Multiplex Subsystem Operation section.

The component units which make up the Multiplex Subsystem are:

- Multiplex Loop Interface Adapter
- Loop Multiplexer
- Communication Line Adapters
- Programmable Cyclic Encoder

Multiplex Loop Interface Adapter (MLIA)

The Multiplex Loop Interface Adapter (MLIA) is an integral part of the Multiplex Subsystem. It inter-connects the Communications Processor and the Loop Multiplexers. It provides serial-to-parallel, and parallel-to-serial data conversions, loop control and error monitoring plus buffering and smoothing of communication demand peaks in both directions.

An MLIA will accommodate up to 256 communication channels at speeds up to 56KB. The unit is packaged on three (3) 11"x14" circuit boards plugged into and operating as peripheral equipment of a Communication Processor.

Performance

Quantitative parameters associated with the MLIA are as follows:

- Input Loop Rate = 20,000,000 BPS ± 1%
- Output Loop Rate = 20,000,000 BPS ± 1%
- Number of Circuits = 256
- Three separate levels of data loopback for diagnostic use
- Maintenance is at the module level, defective cards are swapped out, returned to repair station

Loop Multiplexer (LM)

The Loop Multiplexer (LM) provides the electrical and mechanical interface between the Multiplex Loop and Communication Line Adapters (CLA) which reside within the Loop Multiplexer.

The Loop Multiplexer consists of one loop interface logic card and a wired card enclosure with connectors suitable for rack mounting.

Functional Features

- Communication Line Adapter (CLA) demand-driven information flow
- 20 x 10^6 BPS high-frequency link speed
- 8 LM's per Multiplex Loop
- 32 maximum number of circuits serviced per Loop Multiplexer
- Cyclic check character generation on high-frequency input loop
- Cyclic check character verification on high-frequency output loop
- Transparent data handling
- Error response on input loop
- Error response on output loop

Structural Features

- Maximum number of CLA cards accepted by Loop Multiplexer = 16
- Card size = 11" x 14"
- Any CLA card can be inserted into any Loop Multiplexer CLA slot
- Rack mount (Retma 19" std.)
- Cage size = 19 x 16 x 16 inches

Configuration

The maximum number of Loop Multiplexers in a given subsystem is limited to 8 and the group of multiplexers may contain up to 256 circuits.

Operational Characteristics

- CLA bit rate maximum = 56K bps
- CLA demand-driven input and output
- Error rate of Loop Multiplexer = 10^{-11}
- Error detection and reaction on high-frequency loop

Interface Requirements

The Loop Multiplexer connects to other Loop Multiplexers and the Multiplex Loop Interface Adapter (MLIA) via two coaxial cables (the Input and Output Loops). The maximum distance between LM's or the MLIA and LM's is 100 feet, however, as configured, they are included in the same cabinet(s).

Communication Line Adapters (CLA)

Communication Line Adapters (CLA) are used to interface to communications facilities and provide level conversion and isolation, control, and interim character buffering.

There are three classes of CLAs. They are for:

- Asynchronous Transmission
- Synchronous Transmission
- TDM Line Interface

Asynchronous CLA

- *Half-duplex operation
- *Full-duplex operation
- *Echoplex operation
- *Variable code length (5 to 8 bits plus parity per character)
- *Variable baud rate (45 to 9600 baud — all standard speeds)
- *Input and output speeds may be different
- *1, 1.5, or 2 stop bits
- *Even, odd, or no parity check on input
- *Even, odd, or no parity generation on output
- *Loop-back self test mode
- *Break detection and generation
- *Reverse channel detection and control
- *Data transfer over-run detection
- Variable 8-bit CLA address
- Full EIA RS 232C/CCITT V.24 interface
- 2 CLA's per CLA card
- Modem interface connector on card handle
- Modem indicator lamps on card handle
- Circuit sophistication (MSI and LSI)
- Card size = 11" x 14"

*Program controlled

Synchronous CLA

- *Half-duplex operation
- *Full-duplex operation
- *Variable code length (5 to 8 bits plus parity per character)
- *Variable baud rate to 56K BPS (determined by modem)
- Transparent data handling
- *Variable synchronization character
- *Loopback self test mode
- Provision for external modem clock source
- Variable 8-bit CLA address
- EIA RS 232/CCITT V.24 interface
- AT&T 301/303 interface
- CCITT V.35 interface
- 2 synchronous CLA's per card
- Modem interface connector on card handle
- Circuit sophistication (MSI and LSI)
- Modem indicator lamps on card handle
- Card size = 11" x 14"
- Local terminal connection optional

*Program Controlled

Configuration

A minimum Communication Line Adapter configuration consists of one card to be accepted by one Loop Multiplexer. Each card contains logic to terminate two circuits. The maximum configuration is limited by the 8-bit Adapter addresses to 256 circuits on 128 CLA cards.

TDM Line Adapter

TDM Line Adapter provides for the inter-connection of remotely located Time Division Multiplexers (TDM) without the necessity of local TDM counterparts. This is accomplished by a hardware and software combination which allows the system to input the TDM data directly into memory where the de-multiplexing can take place. This allows users to obtain further economy by not requiring fan-out to a number of access ports.

Features of the TDM Line Adapter are:

- Replaces multiple local ports and local TDM equipment
- Compatible with Timeplexer T-16, T-20 and ADS 660 TDM units
- Byte multiplexer
- Synchronous operation
- Variable CLA address
- Modem interface connector on card handle
- Modem indicator lamp on card handle
- Card size 11" x 14"

PROGRAMMABLE CYCLIC ENCODER

The Programmable Cyclic Encoder is used to generate/verify cyclic redundancy checksums for data being received or transmitted over communication lines. This device is extremely flexible, allowing the implementation of many polynomial error detecting codes including that used for Binary Synchronous (BISYNC).

After computing a message checksum the results are available for program access until destroyed by the issuance of a new character to the encoder.

This feature reduces line interface adapter hardware complexity and saves significant processor time by accomplishing the message checksum generation and validation in hardware instead of software.

MULTIPLEX SUBSYSTEM OPERATION

The Multiplex Subsystem is configured as seen in Figure 1. The MLIA interfaces to the CP via the Interrupt Data Channel (IDC) and to memory via the Direct Memory Access (DMA) Channel for the transfer of input data directly to memory. Output data is handled by the IDC data transfer path.

The key to the ability of the Multiplex Subsystem to operate in a demand-oriented fashion is the Multiplex Loop connecting the MLIA with the Loop Multiplexers (LM). The mux loop is a very high speed serial channel which can be compared to a railroad and the MLIA and LM(s) can be compared to stations on the railroad. Data on the mux loop is analogous to "trains" on the railroad track.

Input

Data from remote terminals is received in serial fashion over the communication lines and placed into single character buffers in the Communication Line Adapter (CLA). When a character is received, the CLA notifies the Loop Multiplexer by signalling on a special line associated with each CLA on an LM. At this time the CLA buffer is acting as a waiting room for the passengers (data) wishing to board the "train" to the CP.

The "trains" of data originate as empty trains at the MLIA on the 20-MHZ serial input mux loop. The "trains," called input loop batches, are composed of loop cells which are the cars of the train in the analogy. A loop cell is composed of 12 bits comprised of:

- 1 bit frame marker (start of cell)
- 3 bits cell identification
- 8 bits cell information

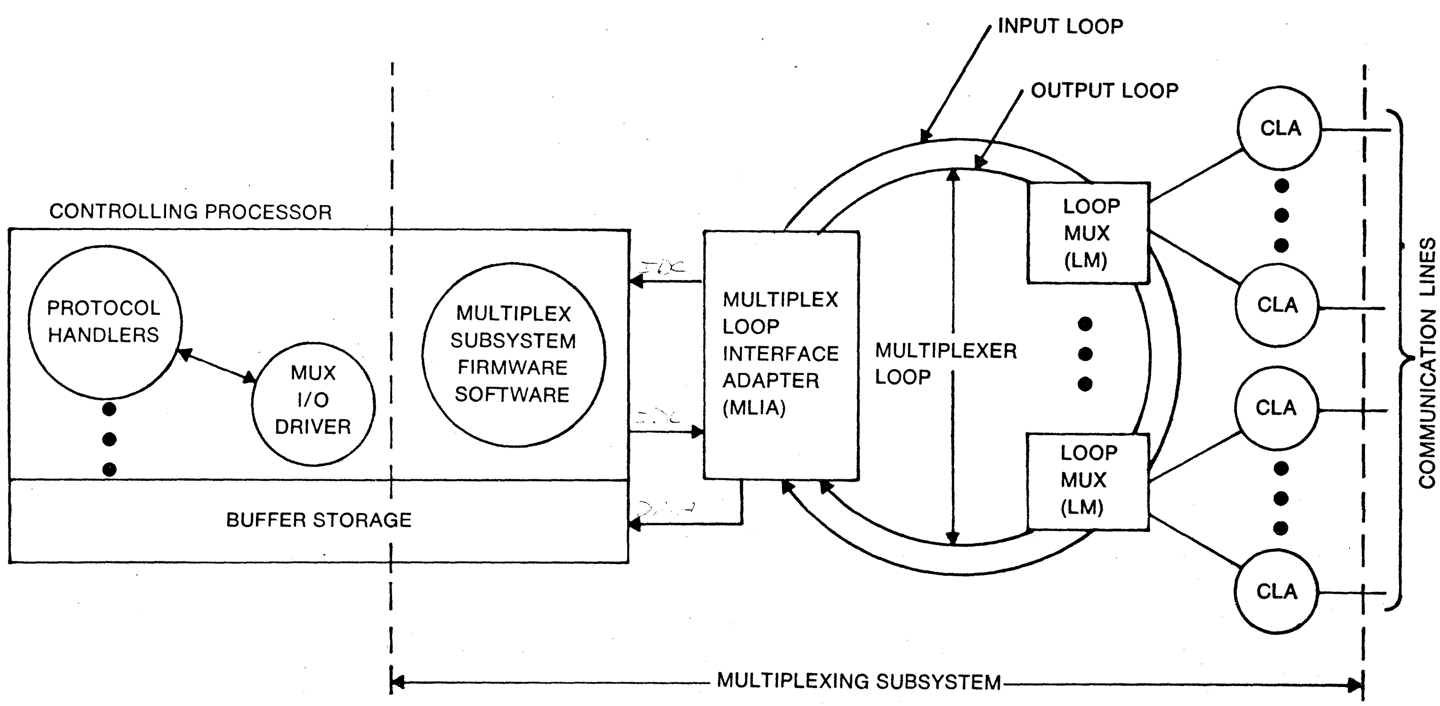


Figure 1. Functional Elements of the Multiplex Subsystem

As an input train passes a Loop Multiplexer which has input data to be forwarded to the CP, the LM marks the first empty frame as a CLA address by setting the appropriate cell identification and places the CLA address in the 8 information bit locations. The subsequent empty cell is marked as data and the data from the CLA is placed in the information bit locations. If a CLA has supervisory information it is placed in subsequent cells. The next empty cell is labeled as a "CRC" cell and the information bit locations are filled with a special checking code called the Cyclic Redundancy Checksum.

The above procedure is repeated for other CLA's on that Loop Mux and then for all CLA's on the adjacent Loop Muxes until there are no data bits left in the CLA's or no empty cells left in the train.

The train returns to the MLIA on its receive side. The MLIA, using the CRC, verifies the incoming data from each CLA. If the data is incorrect the next loop batch will identify that data was incorrectly received and the MLIA initiates error recovery procedures. If the input data is found to be good the MLIA, using the DMA Channel, places the data into a memory buffer as defined by memory resident control tables. The MLIA performs housekeeping on the control tables and if necessary will generate an interrupt to the CP to signal input buffer full for a given communication line. Upon arrival at the MLIA receive side, the MLIA initiates action to start the next train on the Loop.

Output

Output of data is accomplished in a method functionally similar to input.

In this case, "trains" of data originated by the MLIA on the output mux loop are not empty. In the outputting of data the output loop batches are composed of loop cells which specify the where, what and how of data transmission.

The "where" of the data transmission is in the form of a loop cell tagged as an address and containing, in the information field, the address of the CLA which is to send the data.

The "what" of the data transmission is a loop cell tagged as data which immediately follows the CLA address cell. The information field of this cell contains the data to be output.

The "how" of the data transmission consists of one or more cells (usually two) tagged as supervisory cells. The information contained in the cells is used for CLA command functions.

Finally, there is a CRC cell signifying the end of the activity for that CLA. The output loop batch may contain output data or commands for many CLA's.

Once the data flow from the CLA to the remote terminal has begun, the CLA signals the MLIA when more data is required. This is done by supervisory command on the input.

When the CLA demands more data, it signals the Loop Mux in a way similar to the manner in which input data is announced. The LM then uses an empty loop cell in the next input train to signal the fact that the CLA needs more data. The input train contains only two cells for an Output Data Demand (ODD). The first cell is a combination of ODD/Address and the CLA. The second cell is a CRC for error checking.

When more data is requested via an ODD the MLIA informs the CP which outputs the next character via a register I/O program sequence.

MULTIPLEX LOOP CONTROLLER

The Multiplex Loop Controller (MLC) is a micro programmed controller for the high-performance HCP.

The MLC serves the function of controlling completely the bi-directional data flow between the Mux Loop, where it interfaces with a version of the MLIA with DMA output, and main memory where it may have its own access port.

Using the MLC, the CP is no longer required to output using program register sequences. This and the freedom from memory interference allows the high-performance configuration to achieve approximately three times the throughput of the low-performance configuration.

2550-1 HOST COMMUNICATION PROCESSOR

The 2550-1 is the entry-level system for small configuration requirements. This system includes a Communication Processor, memory, Multiplexer Loop Interface Adapter (MLIA), CDC CYBER Channel Coupler, and Loop Multiplexer as well as a maintenance panel, maintenance tape cassette, cyclic encoder and power supplies. It provides a capability to support optional line printers and card readers. It requires a communication console, which is not provided.

Memory

The 2550-1 includes 24K of 600-nanosecond 16-bit main memory with the ability to expand to 32K words of memory with the addition of a 2554-8 Memory Expansion Module (8K words/module).

Throughput Capacity

The 2550-1 is conservatively rated at 10,000 characters/second continuous throughput with considerable reserve capacity for peak loading conditions.

Circuit Connectability

The 2550-1 includes multiplexer capacity to interface up to 32 lines using 256X Communication Line Adapter cards. With the addition of a 2556-2 Loop Multiplexer up to 64 total lines may be supported.

Expansion

The following configuration parameters apply to the 2550-1:

- 64 lines maximum connectability
- 32K words maximum main memory
- One local line printer
- One local card reader

2550-2 HOST COMMUNICATION PROCESSOR

The 2550-2 is the system to be supplied for requirements which exceed or which will grow to exceed the capacity of the 2550-1 entry level system. This system includes a Communication Processor, memory, CDC CYBER Channel Coupler, Multiplexer Loop Interface Adapter (MLIA), and Loop Multiplexer as well as a maintenance panel, maintenance tape cassette, cyclic encoder, expansion cabinet and power supplies. It provides the capability to support optional line printers, card readers and an additional communication channel coupler. It requires a communication console, which is not provided.

Memory

The 2550-2 includes 32K of 600-nanosecond 16-bit main memory. The 2550-2 may be expanded to 65K words of memory by the addition of 2554-8 Memory Modules (8K each).

Throughput Capacity

The 2550-2 is conservatively rated at 10,000 characters/second continuous throughput with considerable reserve capacity for peak loading conditions.

Circuit Connectability

The 2550-2 includes multiplexer capacity to interface up to 32 lines using 256X Communication Line Adapters. With additional 2556 Loop Multiplexers up to 128 lines may be supported.

Expansion

The following configuration parameters apply to the 2550-2:

- 128 lines maximum connectability
- 65K words of main memory
- Multiple local card reader (2572-1 or 2)
- Multiple local line printer (2570-1 or 2)
- Additional channel coupler (2558-1)

2552-1 HOST COMMUNICATION PROCESSOR

The 2552-1 is the system offering a high volume of throughput and circuit connectability. The 2552-1 includes a Communication Processor, memory, a processor-based Multiplex Loop Controller (MLC), Multiplex Loop Interface Adapter (MLIA), CDC CYBER 70/170 Channel Coupler. Also included are a maintenance panel, maintenance tape cassette, cyclic encoder, expansion cabinet and power supplies. The 2552-1 requires a communication console, which is not included.

Multiplex Loop Controller

The Multiplex Loop Controller is a processor-based micro-programmed peripheral controller for the Multiplex Loop comprising this communication subsystem. The MLC, as it is called, handles all communication tasks for the HCP. With its own direct memory interface, the MLC can assist in achieving significant throughput. MLIA bi-directional Direct Memory Accesses are also featured.

Memory

The 2552-1 includes 32K of 600 nanosecond 16-bit main memory. The 2552-1 has the ability to expand to 128K words of memory by using additional 2554-8 Memory Modules (8K words per module).

Throughput Capacity

The 2552-1 is conservatively rated at 30,000 characters per second continuous throughput with considerable reserve capacity for peak loading conditions.

Circuit Connectability

The 2552-1 includes multiplexer capacity to interface up to 32 lines using 256X Communication Line Adapter (CLA) cards. With the addition of up to seven 2556 Loop Multiplexers and the required main memory, the 2552-1 can support up to 256 lines.

Expansion

The following configuration parameters apply to the 2552-1:

- 256 lines maximum connectability
- 128K words of main memory
- Multiple local card readers
- Multiple local line printers
- Additional channel coupler

COMMUNICATION LINE ADAPTERS

The 256X Series of Communication Line Adapters are used to interface the Host Communication Processor to various types of communication lines at the modem or data set.

There are three types of CLA's whose details are described elsewhere in this publication. They are:

2560 Synchronous Communication Line Adapter

The 2560 Series provides for the connection of two synchronous communication lines at speeds up to 56,000 bits/second. They feature software selection of full duplex/half duplex, 5 to 8 bit code length (plus parity), frame synchronization, and loop testing facilities.

2560-1 Synchronous CLA

Provides for the connection via modems (data sets) of lines conforming to the EIA RS232C or CCITT V24 Interface Standard at speeds up to 10,800 bits per second. This CLA is compatible with AT&T 201/208 Data Sets and provides interface for two lines. Local terminal connection without modem is possible by strapping options and cables.

2560-2 Synchronous CLA

This CLA has identical features with 2560-1 except the inter-connection is, in this case, with two lines compatible with AT&T Data Sets 301/303 at speeds up to 50,000 bits per second.

2560-3 Synchronous CLA

This CLA has identical features with 2560-1 and 2560-2 except the interconnection of two lines conforming to CCITT Rec. V35 (AT&T Digital Data Network) at speeds up to 56,000 bits per second.

2561 Asynchronous Communication Line Adapter

The 2561 provides for the connection of two asynchronous communication lines at all standard speeds up to 9600 bits per second. The 2561 features software selection and control of half duplex/full duplex/echoplex operation; code lengths of 5-, 6-, 7-, 8-bits plus parity; variable speed; even, odd, or no parity operation; stop bit duration of 1.0, 1.5, or 2.0 units; and loop testing capability.

The 2561-1 provides for the interconnection of two lines conforming to EIA RS 232C or CCITT Rec. V24 standards. It is compatible with AT&T 103/113/202 Data Sets or their equivalents. Local connection without modem is available via CDC supplied cables.

2562 Time Division Multiplexer Line Adapter

The 2562-1 provides for the connection of a single, local byte-oriented Time Division Multiplexer. It provides multiplexing/de-multiplexing capability compatible with Timeplexer T-16, T-20 and ADS 660 Time Division Multiplexers.

PERIPHERAL DEVICES

Communication Console

The HCP Systems (2550-1, 2550-2, or 2552-1) all require Communication Consoles for their operation. The interface provided for the Communication Console is EIA RS232C at signalling rates of up to 9600 bits per second asynchronous using ASCII TTY protocol. Control Data Corporation will supply any of the following Standard Products for the Communication Console:

- 1711-4 Teletype, 10 characters/second page printer.
- 713-10 CRT, variable speed up to 30 characters/second. Has various options including hard copy output.

Card Readers

The HCP may, at the user's option, be configured with one of two Card Readers. The readers feature 80-column photoelectric reader mechanisms with light/dark checking. The readers are desk-top models with controller. The versions available are:

- 2572-1 300 cards per minute
- 2572-2 600 cards per minute

Both of the above readers feature 1500 card stacker and 1000 card input hopper.

Line Printers

The HCP may, at the user's option, be configured with one of two Line Printers. These printers feature 136 columns, full line buffering, 12 VFU Channels for forms control, 64-character print set, and choice of 6 or 8 line per inch vertical spacing under program control. The printers are provided in quietized cabinets with controller.

The versions available are:

- 2570-1 — 300 line per minute printer employing a drum printing mechanism for low cost.
- 2570-2 — 1200 line per minute printer employing a train printing mechanism for high quality, high volume print-outs. The print train is selectable from several optional sets and is not included in the basic 2570-2. Available print trains include CDC Models 595-1 through 595-6, however the 595-4 is fully supported by HCP software.

SYSTEM REDUNDANCY

The Network Communication System hardware and software design incorporates a high degree of redundancy. This capability will be fully supported upon completion of the second phase of the NCS implementation (Figure 3). A discussion of the redundancy capability is provided here in order to provide the user with a better understanding of the approach being taken by Control Data. An exception to this future redundancy is the 2558-1 Communication Coupler (Channel Coupler). It will be available with the first delivery of the HCP interfaced to the Network Operating System. (Figure 2)

The CDC CYBER 170 and its attendant Network Communication System will have an unusually high degree of redundancy configurable within the system as seen in Figure 3.

This figure does not address redundancy of any CDC CYBER 170 peripherals except the HCP. Note that in this case there exists fully redundant data paths from Loop Multiplexer to the CDC CYBER 170.

If either CDC CYBER 170 were to fail, there exists a data path from both HCP's to the remaining CDC CYBER 170 through the Channel Couplers to the PPU of that machine.

If either of the HCP's or attendant Multiplexer Loops fail, the remaining HCP is able to service the Loop Multiplexers of the failed machine by access to the CLA's of the failed HCP via the secondary Loop Multiplexer board of the Loop Multiplexers on the failed HCP. The secondary and primary Loop Multiplexers share a common access to the CLA's making such back-up possible.

The only constraint about such redundancy of hardware is that all CLA's have unique addresses.

When a CLA fails, the communication line is switched to a back-up (spare) CLA. Communication Line Adapters are not duplicated. Because of the ease of configuring a CLA to a given channel under program control and because of the CLA's inherent reliability, this should prove sufficient back-up in most instances.

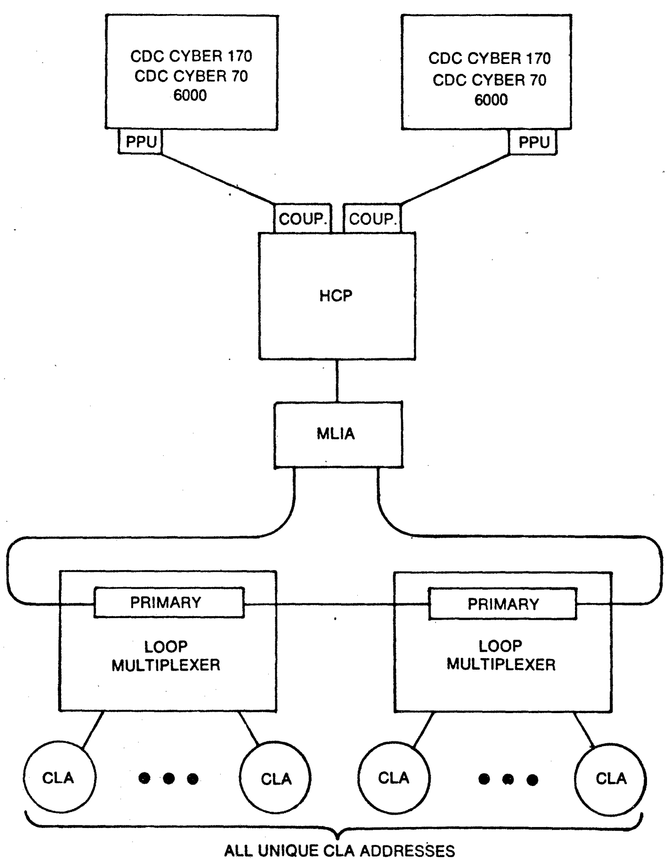


Figure 2. NCS Phase-1 Redundancy

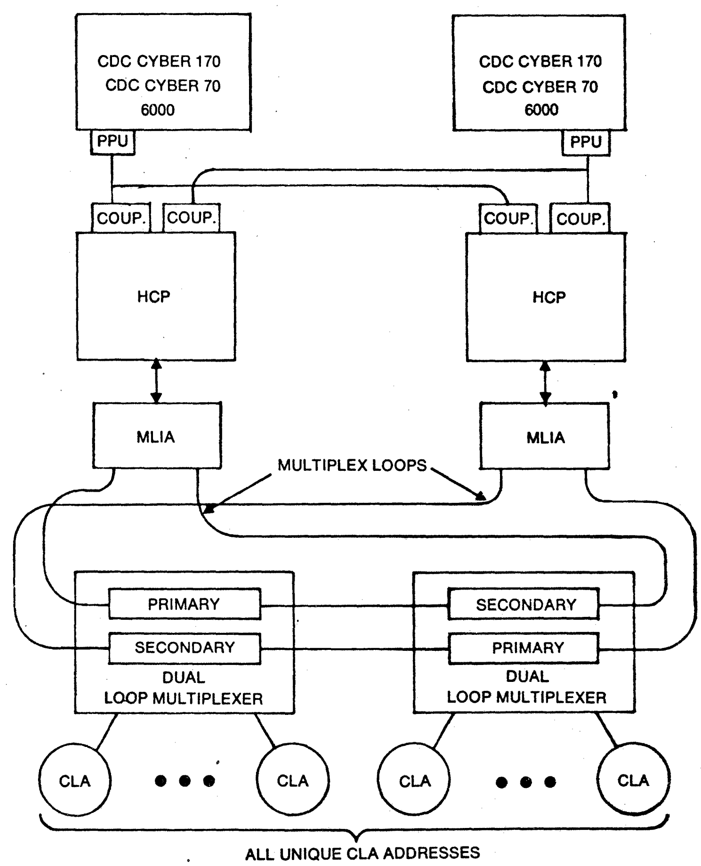


Figure 3. NCS Phase-2 Redundancy

SOFTWARE

The software associated with the hardware is equally as important and provides the user with the capacity to use the hardware and with the ability to inter-connect the communication system with user applications programs that are resident in the host computer. This Section of the Product Description will provide an overview of the standard software to be provided by Control Data. This will include the Communication Control Program (the HCP resident software); the Support Software Package that enables user programming of the HCP; software interfaces within the host computer operating system; and an overview of the software to be provided in conjunction with the fully implemented Network Communication System.

Communications Control Program

The Communications Control Program, or CCP, is the composite software that is executed by the Host Communications Processor. It is composed of several elements which are more fully described below.

Base System Software

The Base System Software is the "kernel" upon which the Communications Control Program is based. The Base System will perform the functions of an operating system, but will also provide control for the flow of information between the network nodes and the host processor. Included in this Base System is the Multiplex Subsystem Software/ Firmware.

Base System Software Functions

MONITOR

- Provides program scheduling in a multiprogramming environment.

TEST UTILITY PACKAGE

- Provides console interface for debugging (Enter/display core, breakpoint, trace, enter/display registers, dump to printer, program start/stop).

SYSTEM LOADER

- Provides for micro memory and core initialization from tape cassette, magnetic tape, or down line from the host computer.

INITIALIZATION

- Provides table initialization following system load or restart.

SPACE MANAGEMENT

- Provides dynamic allocation of memory in multiple buffer sizes.
- Provides for collection of small buffers into larger buffers.

QUEUE MANAGER

- Provides a common point for all queuing and dequeuing of messages.
- The user programs will be transparent to queue structures or storage resource.

LIST SERVICE

- Provides logic to permit a flexible scheme of inter-program communication through the use of worklists.
- Entries may be entered into and removed from worklists. Worklists may be searched for active entries.
- The List Service Program logic may be used to manipulate data structures other than worklists where applicable.

FIRST LEVEL INTERRUPT HANDLERS

- Determines interrupting device and transfers control to the appropriate interrupt handler program.

GENERAL TIMING

- Maintains current time-of-day.
- Distributes timing to all required software tables. Upon detection of any timer expiration, the appropriate worklist is constructed.

CONSOLE TERMINAL INTERFACE PACKAGE

- Provides the logic to interface to the communication console via the I/O Channel. The device must be teletype compatible.

ALARMS DRIVER

- Provides for basic statistics recording and unsolicited reporting of abnormal conditions for all internal programs.

CONFIGURATOR DRIVER

- Provides for system and line oriented table modifications during on-line operation.

ON-LINE DIAGNOSTICS

- Provides for isolation of a communication line problem to the vendor level (CLA, transmission line, local modem, remote modem, etc.). It is initiated by supervisory control.

MEASUREMENT

- Provides on-line statistics recording of predefined internal events to be output to an external device.
- Used for system performance analysis and is initiated by supervisory control.

PROCESS DISTRIBUTION

- Provides common method for internal routing messages to processes that service those messages.

CONSOLE INPUT INTERPRETER

- Provides the command processing from the local console for a minimum set of supervisory commands.
- Builds worklist entries for the appropriate command processors.

REPORTS DRIVER

- Provides for basic solicited report data gathering and distribution to the requestor.
- Basic report formatter for the local console will be provided as an optional feature.

PERIPHERAL DRIVERS

- Provides for software control of card reader(s) and line printer(s) attached locally to the Host Communications Processor.

Multiplex Subsystem Software/ Firmware Functions

- Firmware processes each input character and distributes to by-line buffers.
- Firmware provides a point-of-interface for special processing of each input character depending on the state of the line and the terminal type being serviced, e.g., special character sequence recognition, character counting, CRC generation and code translation.
- Firmware distributes characters to line adapters from by-line output buffers.
- Software provides a logical interface to user programs for all communication commands and status.
- Logical interface allows user programs to be transparent to method of hardware connection, i.e., a line connected via a TDM will appear the same as a line having a direct port connection.
- Dynamic allocation of table space provides capability to connect a large number of lines.

Host Interface Program

The second segment of the Communication Control Program (CCP) is identified as the Host Interface Program (HIP). The function of interfacing the HCP to the Host CYBER Computer is delegated to the HIP. It is the duty of the HIP to be able to communicate with the host in a protocol correct for the particular version of the host operating system, whether it be NOS or SCOPE 3.4.

The HIP receives data from the other elements of CCP, formats it according to host protocol, and, with the appropriate address information about which terminal sent the data, passes it to the CDC host processor. On the output of data from the host, the HIP formats the data according to the internal protocol of the HCP and passes it to the other elements of CCP for proper disposition to the correct terminal.

Terminal Support Software

The third segment of the Communications Control Program is the terminal handlers, providing interface to various terminals. Terminals vary widely in the protocols which they use to communicate. The CCP software will support a wide array of terminal systems through the use of the terminal support software, called Terminal Interface Programs or TIP's.

Functions

The function of a TIP is to provide the interface between a given terminal type, following a specific protocol, and the other elements of the CCP which follow a standard protocol for all data flow, whether it be destined for terminal or host.

The TIP must make allowances for those terminal features which comprise the terminal protocol, such as code length and type, line and message framing conventions, equipment timing restrictions and the like. The TIP does not provide the intelligence to drive the Multiplexer Loop or its attendant hardware. That equipment does not depend upon protocol once the software has established the basic line activity to be performed via commands to the individual CLA's (such as full/half duplex, code length, line speed, etc.).

Terminal Types Supported

The TIP's will support a variety of terminal types including the following:

- *Teletype Protocol*
  - CDC 713
  - TTY M33, M35, M37, M38
  - MEMOREX 1240
  - TEKTRONIX 4010
- *IBM 2741 Protocol*
- *Binary Synchronous Protocol*
  - IBM 2780
  - IBM 3780
- *Mode 4 Protocol*
  - CDC 711
  - CDC 714
  - CDC 200 UT BCD/ASCII
  - CDC 241 GRID
  - CDC 731/732/734
- *Time Division Multiplexer*

Support Software Package

The HCP user programmability is provided by a series of program systems, most of which are handled on the CYBER host computer.

This system, sometimes referred to as the Cross System because it crosses support from one system to another, consists of the following programs:

- Pre-Compiler Program
- PASCAL Compiler
- Format Program
- CP Macro Assembler Program
- CP Micro Assembler Program
- Library Maintenance Programs
- Link Editor

The heart of the Cross System is the PASCAL Compiler which will be discussed at length. The following is a synopsis of the modules in the support system:

- Pre-Compiler — a program which creates a Job Control Statement file. This file is used to control the execution of other modules in the system.
- PASCAL — a high-level compiler which allows the implementation of program tasks in very short order.
- Formatter — a program which reformats PASCAL output into an object code format which is compatible with the CP Macro Assembler output.
- Macro Assembler — assembles CP Macro level code.
- Micro Assembler — assembles CP Micro level code.
- Library Maintenance Programs — to update source and object code libraries.
- Link Editor — locates relocatable programs at specified (or implicit) addresses and assigns addresses to symbols (entry point names) and resolves external references.

Overview of PASCAL

The programming language PASCAL is a high-level language patterned after ALGOL 60. Using PASCAL, the implementor defines the task in statements which are processed by the compiler to yield a variable number of actual program instructions.

The PASCAL Source Language consists of two essential parts:

- Description of actions to be performed.
- Description of data on which actions are performed.

Actions are defined by statements while data is described by declaratives and definitions.

Data

The data are represented by values of variables. Every variable occurring in a statement must be introduced by a variable declaration which associates an identifier and a data type with that variable. The data type defines the set of values which may be assumed by that variable.

The basic data types are the scalar types. There are three standard scalar types: integer, Boolean, and char. Values are denoted by numbers for type integer, by identifiers for type Boolean and by triple marks for type char. The set of values for type char is the CDC 63-character set. The programmer may define an enumeration type. The definition indicates an ordered set of values (i.e., introduces identifiers standing for each value in the set). A type may also be defined as a subrange of a scalar type by indicating the smallest and the largest value of the subrange.

Structured types are defined by describing the types of their components and by indicating a structuring method. In PASCAL, there are three structuring methods available: array structure, record structure, and set structure.

In an array structure, all components are of the same type. A component is selected by an array selector whose type is indicated in the array type definition and which must be scalar. It is usually of the type integer.

In a record structure, the components (called fields) are not necessarily of the same type. A record selector is an identifier uniquely denoting the component to be selected. These component identifiers are declared in the record type definition.

A record type may be specified as consisting of several variants. This implies that different variables, although said to be of the same record type, may assume structures which differ in a certain manner. The differences may consist of a different number and different types of components. The variant which is assumed by the current value of a record variable is indicated by a component field which is common to all variants and is called the tag field. The part common to all variants may consist of one or more components, including the tag field.

A set structure defines the set of all subsets of values of its base type. The base type must be a scalar type, and will usually be a programmer-defined scalar type or a subrange of the type integer.

In contrast to explicitly declared variables, some variables may be generated by an executable statement. Such a generation yields a pointer which subsequently serves to refer to the variable. The pointer may be assigned to other variables of type pointer. Every pointer variable may obtain pointers pointing to variables of the same type T only, and it is said to be bound to this type T. It may, however, also obtain the value "nil," which points to no variable.

Actions

The most fundamental statement is the assignment statement. It specifies that a newly computed value be assigned to a variable (or component of a variable). The value is obtained by evaluating an expression. Expressions consist of variables, constants, sets, operators and functions operating on the denoted quantities and producing new values. Variables, constants, and functions are either declared in the program or are standard entities. PASCAL defines a fixed set of operators. The set of operators is subdivided into groups of:

- Arithmetic operators of addition, subtraction, multiplication, division, sign inversion, and computing the remainder.
- Boolean operators of union, conjunction, and negation.
- Set operators of union, intersection, and set difference.
- Relational operators of equality, inequality, ordering, set membership, and set inclusion. The results of relational operations are of type Boolean. The ordering relations apply only to scalar types.

The procedure statement causes the execution of the designated procedure. There are statements which specify sequential, selective, or repeated execution of their components. Sequential execution of statements is specified by the compound statement, conditional or selective execution by the IF statement and the CASE statement, and repeated execution by the FOR statement, the REPEAT statement, and the WHILE statement. The IF statement serves to make the execution of a statement depend on the value of a Boolean expression, and the CASE statement allows for the selection among many statements according to the value of a selector. The FOR statement is used when the number of iterations is known beforehand, and the REPEAT and WHILE statements are used otherwise.

A compound statement can be given a name, and be referenced through that name. The statement is then called a procedure, and its declaration, a procedure declaration. Such a declaration may additionally contain a set of variable declarations, type definitions and further procedure declarations. The variables, types, and procedures thus declared can be referenced only within the procedure itself, and are called local to the procedure. Their identifiers have significance only within the program text which constitutes the procedure declaration and which is called the scope of these identifiers. Since procedures may be declared local to other procedures, scopes may be nested. Entities which are declared in the main program (i.e., not local to some procedure) are called global.

A procedure has a fixed number of parameters, each of which is denoted within the procedure by an identifier called the formal parameter. Upon an activation of the procedure statement, an actual quantity has to be indicated for each parameter which can be referenced from within the procedure through the formal parameter. This quantity is called the actual parameter.

There are three kinds of parameters: value parameters, variable parameters, and procedure or function parameters. In the first case, the actual parameter is an expression which is evaluated once. The formal parameter represents a local variable to which the results of this evaluation is assigned before the execution of the procedure (or function). In the case of a variable parameter, the actual parameter is a variable and the formal parameter stands for this variable. Possible indices are evaluated before execution of the procedure (or function). In the case of procedure or function parameter, the actual parameter is a procedure or function identifier.

Functions are declared analogously to procedures. The only difference lies in the fact that a function yields a result which is confined to a scalar type and must be specified in the function declaration. Functions may therefore be used as constituents of expression. In order to eliminate side-effects, assignments to non-local variables should be avoided within function declarations.

Cross System Performance

Operational Characteristics

The Cross System requires a CDC CYBER 170/70 System capable of running NOS or SCOPE 3.4. The system is designed to run in less than 100K (octal) words of central memory.

PASCAL Performance

Experience indicates that PASCAL programs can be implemented in about 40% less time than Macro Assembler programs when the programming takes place in a structured environment with experienced programmers working from functional program specifications.

Network Communication System/NOS Software Support

This section relates primarily to the fully implemented Network Communication System working in conjunction with the Network Operating System (NOS). Since the Network Communications System will be implemented in an incremental fashion, it is necessary that this subject be viewed also from the aspect of the Network Operating System. The initial version of the NOS will use the field-proven software TELecommunications EXecutive, TRAnsaction EXecutive and Export/Import 200 to handle interactive, transaction and batch terminals respectively. From this basis, through a series of incremental enhancements, will evolve the completed Network Communication System. The following discussion is provided in order to more completely understand the Network Communication System software structure and resultant user features at the point in time when NCS is fully implemented.

Network Concepts

Control Data will provide a standard capability to establish generalized computing networks, ranging upward from one host with a simple set of terminals. Heterogeneous and geographically separate hosts may be linked with a variety of geographically distributed terminals. The interconnection means is a Communication Network consisting of communication lines and intermediate communications handling equipment (nodes) plus communication software in the hosts and nodes.

These networks will be capable of simultaneously handling wide varieties of data streams associated with differing applications including transaction processing, time sharing, remote batch processing and graphic science.

Capability will also exist for host-to-host communication to support features such as load leveling, multi-task execution, etc.

Network Features Overview

The primary functions of the Network Communications System software are to provide:

- Highly reliable transmission of messages between a terminal and a particular program (process) existing within a host, or between two processes executing in separate hosts.
- Flexible network configurations accommodating the addition or removal of hosts, nodes, or lines with no requirement to change the Communication Subsystem or user application programs.
- Acceptance of new terminal types and user applications without rework of the existing communication or other software.
- Uniform communications subsystem interface between operating system or user application support packages and the Communications Subsystem.
- High level language programming capability for node resident user application programs.

External Network Features

Applications Supported

The generalized computing network will be capable of supporting the entire spectrum of general-purpose computation including both terminal-host and host-host communications as necessary for remote batch job processing, interactive processing, file transmission, graphics work, multi-tasking, etc. Of ultimate importance is the support of applications where a non-programmer may access and manipulate items of data as a terminal user, regardless of where the data may reside physically.

The communications subsystem supports any type of applications message traffic where messages are to be passed unaltered from source to destination.

Configurations Supported

It will be possible to connect hosts and terminals to the network, regardless of their location, using either dedicated or public data communications equipment. Connectable terminals encompass a broad range of devices, including asynchronous terminals such as TTY, 2741 and polled terminals, as well as synchronous terminals such as the 200 UT, 2780, 711/714.

The communication network, itself, is the collection of software, communications equipment and node processors which support all the various possible terminal-host and host-host data flows. It will be easily reconfigurable to allow for expansion, load shifts, etc., in the most effective manner.

Any CDC CYBER 170 mainframe may serve as a host. A single PPU will be capable of handling at least two HCP node processors.

Clean separation of function is maintained within the software so that a user need not experience significant overhead for unused capabilities. For example, the existence of batch capability does not significantly affect the performance of a system devoted exclusively to interactive work.

A process within a host may conduct simultaneous communications with a multiplicity of terminals and/or other host processes. A terminal may be logically connected with only one host process at any given time.

Terminal/Host Support

The total support of specific terminal types is the joint responsibility of the host computer and Terminal Interface Package (TIP) in the node connected directly to the terminal. Because of the possible existence of dissimilar hosts and the application dependence of some aspects of terminal support, the functions provided by the TIP are host-independent and standardized. A host application may communicate with a terminal in a device independent manner or may take full control of all terminal features. In the device dependent mode, the TIP handles only the line management and link control; leaving all device control to the host. In the device independent mode, the TIP handles any of a class of similar devices as a standard virtual terminal. This class may be extended in the future as new requirements dictate. A single virtual terminal (Interactive Virtual Terminal) is to be supported by the communications subsystem software. The class of terminals supported by this virtual terminal includes point-to-point or multi-drop connected interactive devices using asynchronous or Mode 4A or 4C synchronous protocols.

Host-Host Support

The communication network handles host-host and host-terminal traffic as compatible subsets of the network protocol.

Application Dependent Functions

As noted previously, the communications subsystem is independent of and transparent to the applications it supports. The development of specific applications in a network environment may involve any of several steps:

- Coding of host application programs.
- Development of an application executive.
- Development of node application subroutines.

Several aids to applications development will be provided:

- A communication control system in each host to provide static (initialize time) or dynamic (connect time) allocation of terminals to processes, establish terminal configuration, establish logical connections, gather statistics, run diagnostics, etc.
- A network control system operating in only one host within the network responsible for: loading all nodes, establishing network configuration (node identifiers, node linkages and routing tables), monitoring the network for failures, gathering statistics, etc.
- A communications subsystem interface process in each host which allows system and application support packages to communicate through the network.
- A program generation and loading system for the HCP, running on a CDC CYBER host. This system includes a high level language compiler (PASCAL) and a link editor to build a system from separately compiled or assembled code modules. This system will be the primary code generation tool for the standard node processor software and user written extensions to that software.

Internal Network Features

Interface Definition and Maintenance

In order to make possible an orderly evolution of increasing levels of network capability, a system of fundamental interfaces between major software components must be defined and preserved.

Among the most significant features are:

- Delivery Assurance Protocol. This is the highest level protocol for sending messages from any source to any destination within a computing network while guaranteeing correct delivery of messages even in the presence of non-debilitating network failures.
- Process Connect Protocol. This protocol establishes a logical connection to initiate communication via the Delivery Assurance Protocol.
- Standard Packet Format. A standard packet is the network unit of data for transmission over a logical connection. Each packet consists of a message segment plus its packet header and trailer which:
  - Determine routing through the network
  - Delimit the packet
  - Provide information required for delivery assurance
- Node-Node Link Protocols. These are physical level protocols which provide a communications and control envelope for one or more Standard Packets to allow them to be sent over communication lines.
- Host-Node Protocols. These provide a communications and control envelope for Data Blocks to allow them to travel over the Host-Node channel connection.
- Communication Subsystem Interface. This is the interface between system programs or applications support packages and the communications subsystem software. It provides the capability for both terminal dependent and terminal independent communications.
- Terminal Interfaces. Each supported terminal will have a specified interface covering line management, link control, and device control. As noted above, many terminal interfaces will differ only in specified optional aspects of the Interactive Virtual Terminal.

Communications Error Detection and Recovery

The Delivery Assurance Protocol provides a system of end-to-end packet sequencing. The packet sequencing system guarantees that no packets will be lost or duplicated without detection. Additionally, Node-Node Protocol includes a local sequence number scheme and a cyclic redundancy check.

Network Level Failure Isolation and Recovery

Each network component will continuously monitor its own operations and those of its adjacent components. When a failing component is identified, the network control system is notified; it reconfigures the network to minimize the affected network resources. Manual procedures such as dialing alternate lines, etc., may be involved.

Failure of the host running the network central system will result in a back-up host taking over the network control responsibilities. Also, in the event of a failed host — not the network control host — it may be possible to assign another host to take over the failed host's responsibilities.

Dynamic Reconfiguration

The network control host will be able to instruct any node to delete or reinstate any of its node trunk lines or change the line number on which it is to transmit packets for a given destination node (change routing). This is done via a system message sent to a node, without having to reload the node. Delivery assurance protocol procedures guarantee the integrity of message traffic will be maintained during a reconfiguration.

In addition to such global reconfiguration, a local form of reconfiguration is possible for back-up purposes. A pair of hosts in a common location may share two node processors in such a manner that any Communications Line Adapter (CLA) may be connected to either of the node processors. Each node processor may communicate with either host and each host may communicate with either node. In the event of a node or host failure in such a configuration, the remaining component can take over the function of its failed partner.

Scope 3.4/Intercom 4.2 Software Support

The SCOPE 3.4 Operating System, in conjunction with CYBER 70 and 6000 systems, offers an extension of the INTERCOM Communications Subsystem (Version 4.2) which allows users the advantages inherently available with programmable communication processors.

A new PPU driver permits the Host Communication Processor with its Communication Control Program to act as a communication controller for the users of SCOPE/INTERCOM.

In this environment, network control will remain in INTERCOM. In larger systems, line termination cost reductions will be significant. This will be especially important to users of Time Division Multiplexing, since only one port will be required for the multiplexed circuit.

Terminal support is the same as now included under SCOPE 3.4 and INTERCOM 4 with the addition of Time Division Multiplexers, and are:

- Teletype M33, 35, 37, 38
- 713-10 Conversational Display Terminal
- 217-XX Remote Display/Entry Controller
- 711-10 CRT Display Terminal
- 714-XX Terminal Controller
- 241-1 GRID Graphics Terminal
- TEKTRONIX 4010, 4014
- 731-12 Low Speed Batch Terminal
- 732-12 Medium Speed Batch Terminal
- 734-1 Multi-Speed Batch Terminal
- Remote Time Division Multiplexer

SYSTEM BENEFITS AND FEATURES

The 2550 series hardware and software offer the prospective user many benefits and include a significant number of features. This section compiles the major benefits and features into a specific discussion.

Benefits —

Costs — The user will be able to realize significant cost savings when evaluating on a per-circuit-connected basis, as opposed to other approaches to inter-connecting terminals to the host processor. When the costs are measured on cost/performance basis the savings are even greater.

Increased Performance — The 2550 series provides substantially increased throughput performance while at the same time requiring the same, or less, host computer system resources. With the HCP, only one Peripheral Processing Unit (PPU) is required regardless of the types of processes being handled (Batch, Interactive or Transaction). In fact, these processes can coexist within the 2550 series.

Expandability — The modularity of the HCP hardware and CCP software is such that there is considerable capacity for future expansion on a highly incremental basis. This benefits the user by providing the capacity for expanding his system at a rate equivalent to his requirements. Thus incremental expansion cost effectiveness is achieved.

Looked at it in another way, HCP and CCP provide the user with the basic elements necessary to expand to provide a viable computer network. This capacity is provided by the implementation plans for the Network Communication System.

Versatility — As a result of using a processor in the hardware complement of the HCP and the modularity and open-ended features of the Communications Control Program, the user is equipped with a highly versatile product. These features, when coupled with system programmability, allow the user versatility not known heretofore in a set of data communications equipment.

Management — Quite possibly the most significant benefit, and yet probably the most subjective, is the user's ability to manage his own data communications environment when using these new CDC products. The user can plan, monitor and modify his data communications system to most exactly match his total requirements. In addition, the user is assured that he can meet his anticipated, and often un-anticipated, future data communications and network communications needs.

Features —

Processor Based — With use of a state-of-the-art communication processor as the heart of the Host Communication Processor product, the advantages of flexibility, modularity, versatility normally associated with a computer are made available to accomplish the task of data communications between terminal and host computer.

User Programmable — With the optional software support package the 2550 user is not constrained to use of solely CDC provided software. The user, at his option, may add other tasks to be accomplished by the HCP, or may include the ability to connect with terminals not included in CDC's list of supported terminals. Other possible tasks assignable to the HCP might include data pre- and post-processing, etc. These additional programs can be added in a cost effective manner, without effect on the standard CDC provided software, or future releases of it.

Modular — Batch hardware and software are highly modular. The HCP hardware allows for modular expansion in the areas of: Channel Coupler; processor memory; maximum data throughput; loop multiplexer; and communication line adapters. The software is also organized modularly allowing for support of other terminals, performance of additional tasks, and even interface to other host computers without impact on the other elements of the software. In addition, the standard software is highly parameterized such that the user may optimize the C.C.P. software for his specific needs, and make use of only the module required for his operation.

Process Transparent — The software structure of the CCP and, in fact, the overall design of the Network Communication System allows data involved in various processes such as Remote Job Entry (Batch), File Update and Inquiry (Interactive), and Transaction to be handled simultaneously. This feature allows a single package of hardware and software to handle the total data communications task.

Demand Driven Multiplexing — With the multiplexing technique employed within the HCP the communication subsystem is truly load sensitive, with little dependency on the number and speed of circuits to which it is connected. In addition the technique allows for future growth such that still higher circuit speeds and data loads can be handled by the sub-system.

Open-Ended Performance — The hardware products are structured so as to allow selection of the capacity best suited to meet the user's needs of today, as well as almost unlimited growth. Growth capabilities include implementation of a geographically distributed Network Communication System and multiple HCP systems.

Terminal/Circuit Independence — Because of the modular terminal handlers built into the HCP resident programs, as well as the advanced concepts employed in the communication line adapters, the system is highly independent of unique terminal and circuit characteristics. This means that the user can, quite easily, add different terminals and circuits without the requirement for unique hardware, and without impacting already developed software. In fact, it is intended that some of the standard terminal handler submodules might be modified or complemented by user supplied subroutines to handle new terminal features.

Host Computer Independence — Because of the high degree of hardware and software isolation at the host computer interface, use of other computer systems is not expected to be a major undertaking.

Local Peripherals — With local peripherals connectable to the HCP, users have a possibility of off-loading current host computer unit record peripherals. The user may find it feasible to completely replace those peripherals in unique system configuration. In such a situation, a significant benefit is gained by release of a dedicated peripheral processor and deletion of software within the host computer. The impact on data communications throughput will be proportional to the increased unit record tasks and must be evaluated with care.

Communication Task Oriented — The overall Network Communication System, and specifically the HCP and CCP, are planned as a part of a total computer network. The NCS concentrates on the problem of handling the data communications task in the most efficient and effective manner possible.

CONFIGURATION INFORMATION

The HCP is arranged as a series of modular products specifically sized to meet various customer needs. This part of the Product Description provides information about: the specific configuration parameters of each of the 255X Series Products; how to determine the specific HCP required for a user's data communications requirements; and finally, information about the physical packaging of the products is also included.

Product Configurators

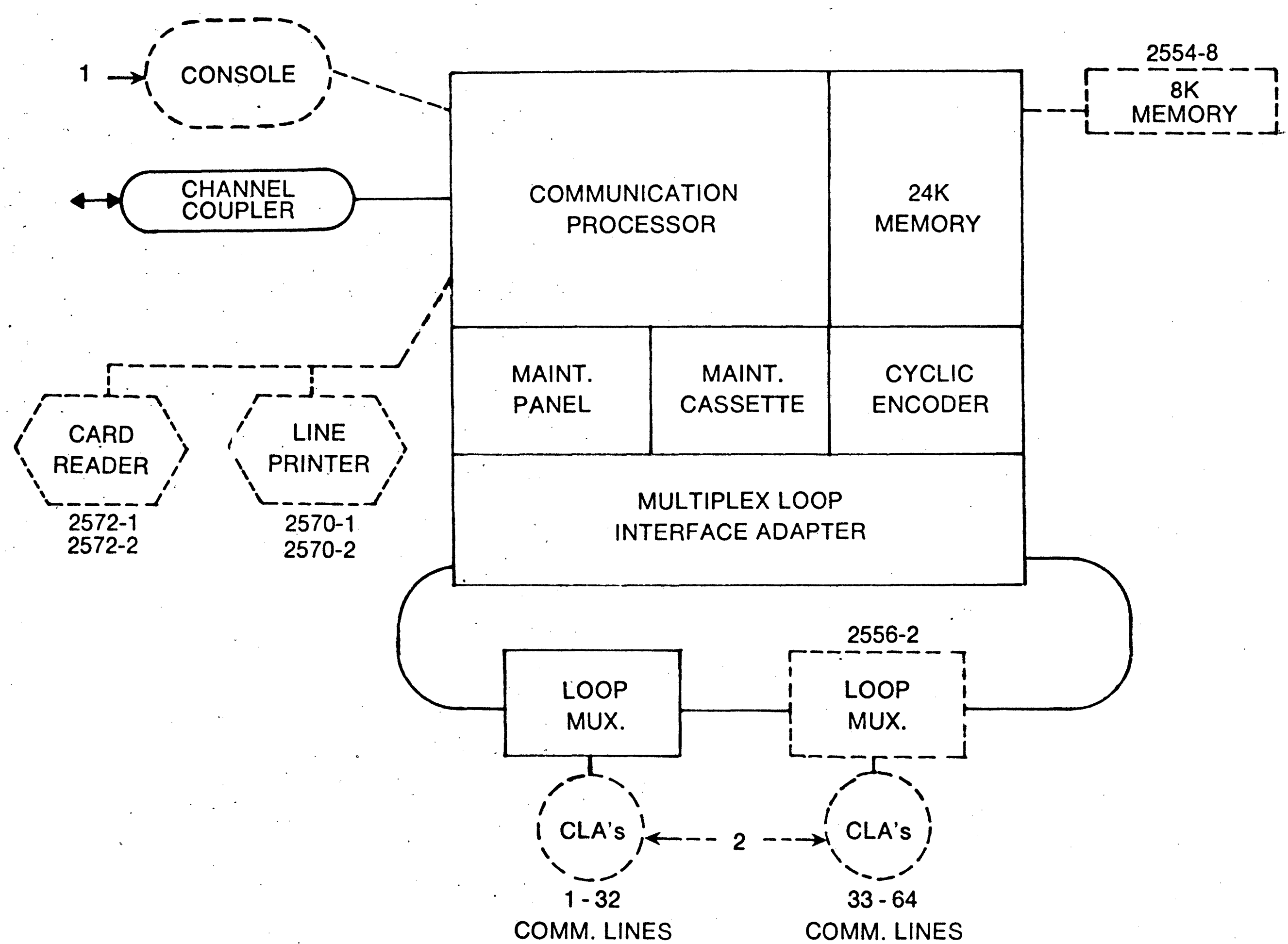
The following configurators for each model of the HCP include the other elements required for a completely operational system as well as indicating the various expansion and enhancement options available and their limits. Detailed information relating to the products is to be found in a preceding section of this product description.

In addition to the notes relevant to each specific configuration, following are some further considerations that are generally applicable:

- The products are not presently field upgradeable from one product to the next higher performance product.
- The 2550-1 may control only one line printer and card reader whereas the 2550-2 and 2552-1 may control multiple local peripheral devices. Communication control program provides support for one card reader and one line printer.
- When connecting local terminals, the interconnecting cable must be ordered separately through CDC. Terminals with an RS232C interface may be connected without modems (locally) within 50 feet of the HCP. Beyond this distance, the economic factors of cable costs versus modem costs should be considered. In special cases, longer cables can be supported on a QSE basis.
- Local synchronous terminals may operate at 2400, 4800, or 9600 bits per second. Timing signals are provided by the communication line adapter.
- As a special option, low-speed terminals (TTY) can be supported at distances up to 20 miles through the use of 362-1 and -2 Telegraphic Level Converters which connect to the Communication Line Adapters.

SYSTEM CONFIGURATION

2550-1 HOST COMMUNICATION PROCESSOR



LEGEND:

________ SYSTEM ELEMENTS INCLUDED IN BASIC 2550-1 PRODUCT.

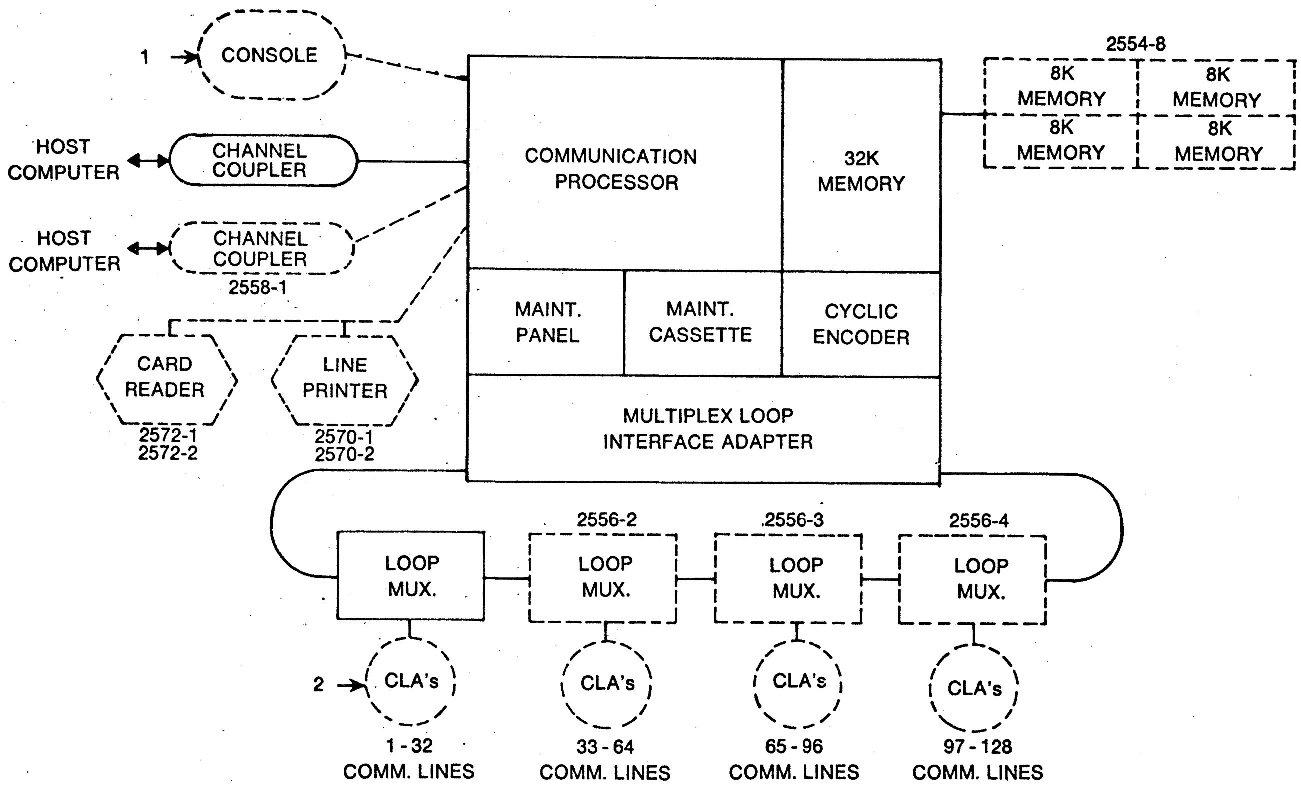
_ _ _ _ _ REQUIRED AND OPTIONAL UNITS WHICH MUST BE SEPARATELY SPECIFIED BY PRODUCT NUMBER.

NOTES:

1. CONSOLE IS REQUIRED, BUT MUST BE SPECIFIED. SELECT 1711-4 (M33 TTY), 713-10, OR INDICATE USE OF CUSTOMER PROVIDED EQUIVALENT DEVICE.
2. SELECT CLA (COMM. LINE ADAPTER) CARDS ACCORDING TO COMM. LINE TYPE TO BE TERMINATED FROM 2560-X, 2561-1, OR 2562-1. EACH 2560 OR 2561 TERMINATES **TWO** LIKE COMM. LINES. MAXIMUM IS 16 CLA CARDS PER LOOP MUX. (32 COMM. LINES). 2562-1 TERMINATES SINGLE COMM. LINE.

SYSTEM CONFIGURATION

2550-2 HOST COMMUNICATION PROCESSOR



LEGEND:

________ SYSTEM ELEMENTS INCLUDED IN BASIC 2550-2 PRODUCT.

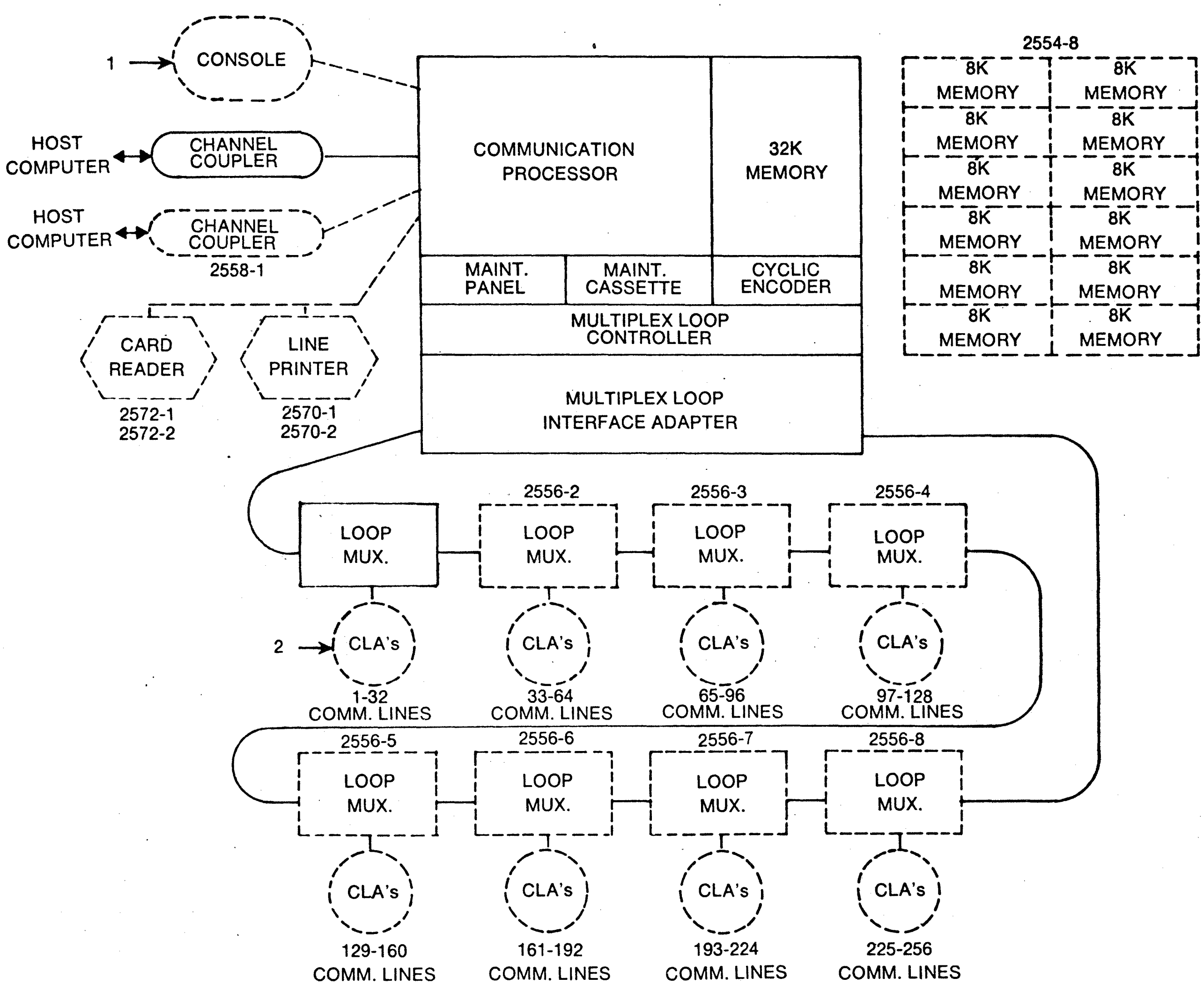
_ _ _ _ _ REQUIRED AND OPTIONAL UNITS WHICH MUST BE SEPARATELY SPECIFIED BY PRODUCT NUMBER.

NOTES:

1. CONSOLE IS REQUIRED, BUT MUST BE SPECIFIED. SELECT 1711-4 (M33 TTY), 713-10, OR INDICATE USE OF CUSTOMER PROVIDED EQUIVALENT DEVICE.
2. SELECT CLA (COMM. LINE ADAPTER) CARDS ACCORDING TO COMM. LINE TYPE TO BE TERMINATED FROM 2560-X, 2561-1, OR 2562-1. EACH 2560 OR 2561 TERMINATES **TWO** LIKE COMM. LINES. MAXIMUM IS 16 CLA CARDS PER LOOP MUX. (32 CIRCUITS). 2562-1 TERMINATES SINGLE COMM. LINE.

SYSTEM CONFIGURATION

2552-1 HOST COMMUNICATION PROCESSOR



LEGEND:

________ SYSTEM ELEMENTS INCLUDED IN BASIC 2552-1 PRODUCT.

\- - - - - REQUIRED AND OPTIONAL ELEMENTS WHICH MUST BE SEPARATELY SPECIFIED BY PRODUCT NUMBER.

NOTES:

1. CONSOLE IS REQUIRED, BUT MUST BE SPECIFIED. SELECT 1711-4 (M33 TTY), 713-10, OR INDICATE USE OF CUSTOMER PROVIDED EQUIVALENT DEVICE.
2. SELECT THE CLA (COMMUNICATION LINE ADAPTER) CARDS ACCORDING TO COMM. LINE TYPE TO BE TERMINATED FROM 2560-X, 2561-1, OR 2562-1. EACH 2560 OR 2561 TERMINATES **TWO** LIKE COMM. LINES. MAXIMUM IS 16 CLA CARDS PER LOOP MUX. (32 COMM. LINES). 2562-1 TERMINATES SINGLE COMM. LINE.

System Configuring

This section is provided to assist you in determining the proper 2550 series products needed to meet a user's specified requirements. The step-by-step procedure outlined in the next several pages, if followed, will result in the definition of:

1. The specific HCP required
2. The Communication line expansion units required
3. The Communication line adapter units required
4. The total amount of memory required

NOTE: The "system configurator" does not provide for such items as:

1. Terminals not supported by standard CDC software
2. Circuits not complying with the: speed; signal level; and byte-size characteristics of the standard line adapters
3. Expansion considerations
4. Special user programs that will reside in the HCP

The configurator is relatively simple to use. In order to properly size (or configure) an HCP to meet a customer's requirements, you need know only how many circuits will be connected to the HCP and some of the characteristics of the circuits.

Step 1. Throughput Definition

A. Worksheets 1 and 2 on the following pages will assist in defining the total throughput, in characters per second, required by a user to meet his specific requirements.

WORKSHEET NO. 1 — SYNCHRONOUS CIRCUITS

DATA RATE Bits/Sec. (1)*	Char./Sec. (2)	HDX=1 or FDX=2 (3)	*Circuit Data Rate (4)	Qty. of Circuits (5)	Extended Char. Rate (6)	% of Line Loading (7)	Throughput Char. Rate (8)
2,000 =	250						
2,400 =	300						
3,600 =	450						
4,800 =	600						
7,200 =	900						
9,600 =	1,200						
=							
=							
=							

(9) TOTAL QTY.
SYNCH. CKTS. __________

(10) TOTAL THROUGHPUT
RATE-SYNCH. __________
(CHAR./SEC.)

*Each item is fully explained on the following page.

WORKSHEET NO. 2 — ASYNCHRONOUS CIRCUITS

DATA RATE Bits/Sec. (1)*	Char./Sec. (2)	HDX=1 or FDX=2 (3)	Circuit Data Rate (4)	Qty. of Circuits (5)	Extended Char. Rate (6)	% of Line Loading (7)	Throughput Char. Rate (8)
50 =	6.6						
75 =	10						
110 =	10						
134.5 =	14.5						
150 =	15						
300 =	30						
600 =	60						
1,200 =	120						
1,800 =	180						
2,400 =	240						
4,800 =	480						
9,600 =	960						
=							
=							
=							

(9) TOTAL QTY.
ASYNCH. CKTS. __________

(10) TOTAL THROUGHPUT
RATE ASYNCH. __________
(CHAR./SEC.)

*Each item is fully explained on the following page.

Worksheets 1 and 2 — Explanatory Notes

(1) & (2) Standard data rates, in bits per second, and the equivalent in characters per second are listed. Blank lines are provided for calculation using other data rates or the same data rates where some lines of the same speed, handle full-duplex and others half-duplex. These charts assume the standard number of bits per character.

(3) HDX (Half-Duplex) or FDX (Full-Duplex) — In Col. (3) enter a "1" if the data flow will be HDX (2-way alternate) or a "2" if it will be FDX (2-way simultaneous). NOTE: This column relates to the actual flow of data — NOT to the capacity of the circuit to handle full-duplex. If both HDX and FDX are required for a specific line speed, treat each separately. Use one of the blank lines at the bottom of the page.

(4) Circuit Data Rate — Multiply cols. (2) & (3) and place result in col. (4). This represents the maximum character throughput rate of each circuit of this type.

(5) Quantity of Circuits — Enter in col. (5) the quantity of circuits of that type existing and/or projected, which the system must accommodate. If none are required enter "NONE" to avoid oversight.

(6) Extended Character Rate — Multiply cols. (4) & (5) and place result in col. (6). This represents the total maximum throughput rate for all circuits of this type.

(7) Percent of Line Loading — This is a very important factor. This is the representation of how much use is made of the circuits and the terminals to which they are terminated. This figure represents the amount of time the circuits are in use, moving data, relative to the total time the circuits are available for use. This should be represented as a decimal value (e.g. 75% = 0.75, 10% = 0.1). The date for this column are normally known if the circuits are already operational, or should be estimatable if the circuits are new requirements. If neither is available, the following rules-of-thumb may be used, with customer concurrence:

1. Point-to-point, dedicated circuit — 50%
2. Point-to-point, dial-up circuit — 65%
3. Multi-point, dedicated circuit — 75%

If different line-load factors exist for different lines of same speed, use blank lines at bottom of appropriate worksheet.

(8) Throughput Character Rate — Multiply cols. (6) & (7) and place result in col. (8). This represents the actual or anticipated average total throughput for all circuits of this type.

(9) Add all entries in col. (5) to determine total quantity of all circuits.

(10) Add all entries in col. (8) to determine total required throughput of all circuits.

Step 2. Circuit Connection Definition

This step will determine if additional Line Expansion units are required, and if so which of the 2556-X series are to be used.

A. Add item (9) from worksheets 1 and 2

B.

If total ckts. is ___ or less	These units must be added to config.
32	none
64	2556-2
96	2556-2 & 2556-3
128	2556-2, -3, & -4
160	2556-2 thru -5
192	2556-2 thru -6
224	2556-2 thru -7
256	2556-2 thru -8
more than 256	requires add'l HCP(s)

Step 3. HCP Definition

This step will determine which of the HCP products is required.

A. Add item (10) from worksheets 1 and 2

B. If total from Step 3A is ______ or less	and	Total ckts. is ______ or less	Then HCP Model ______ is required.
10,000 CPS		64	2550-1
10,000 CPS		128	2550-2
30,000 CPS		256	2552-1
more than 30,000 CPS		N/A	Add'l HCP(s)

Step 4. Communication Line Adapter Definition

This step will determine the number and type of CLA cards required.

A. Synchronous

Divide item (4) worksheet #1 by 2 and round-off upwards. This yields the total quantity of 2560-X CLA cards required. The specific model of the 2560 will depend upon the specific interface criteria and type of modem used on the circuit.

B. Asynchronous

Divide item (9) worksheet #2 by 2 and round-off upwards. This yields the total quantity of 2561-1 CLA cards required.

Step 5. Memory Requirement

(At time of printing the specific program module sizes were not available. This information will be provided in the near future.)

Configuration Flexibilities and Constraints

Preceding system configuration guidelines will produce an appropriate hardware list, based on throughput and circuit count. However, other factors should be taken into account in reaching a final configuration definition.

(1) Consideration must be given to the fact that the 2550 System will not initially be field upgraded beyond the specified limits for each product number. Growth is an almost universal characteristic of communications systems. Where calculations indicate that a network will fit into a specific product set, but only marginal expansion capability is provided, serious consideration should be given to the next higher performance product in the 255X series.

(2) User-generated programs for processing of data within the Network Communication System will consume a portion of the available resources, hence will reduce the stated maximum throughput capability for each product set. No specific rules can be established for evaluation of the effect, since each requirement will be different. Provision must be made on an individual basis, by analysis of the processes to be added.

(3) Communications Line Adapters may be used to terminate local devices without use of modems. This requires special cables, which are not included in the basic system and which must be included as separate items in the equipment list.

Hardware Packaging

The packaging of the Host Communication Processors is simplified in that, with the exception of configuration-oriented items such as memory modules, multiplexers and Communication Line Adapters, no additional equipment is required once the HCP is selected.

The following is a list of HCP packaging features:

- Minimal floor space required
- Expansion room internal to cabinet
  - Memory expansion room
  - CLA expansion room
  - Channel Coupler expansion room, where applicable
- No additional external power required

HARDWARE PRODUCTS SPECIFICATIONS

Physical Dimensions

2550-1 Host Communication Processor
- Height 75 inches
- Width 24 inches
- Depth 34 inches

2550-2 Host Communication Processor
- Height 75 inches
- Width 48 inches
- Depth 34 inches

2552-1 Host Communication Processor
- Height 75 inches
- Width 48 inches
- Depth 34 inches
- Expansion beyond 128 circuits requires one 75" H x 24" W x 34" D cabinet for each 64 additional circuit termination capacity added

Environmental Requirements

Non-operating Environmental Requirements

The HCP system will be able to withstand the following environmental conditions in a non-operating mode, such as encountered during storage and transit.

- Altitude — will withstand pressure range of from 1000 ft. below sea level to 15,000 ft. above sea level.
- Temperature — will withstand range −30°F to +150°F (−35°C to +65°C). Will withstand thermal shock conditions of either +80°F to −30°F or or +80°F to +150°F if allowed to return to stable temperature at a rate not exceeding 20°F per hour.
- Humidity — relative humidity range of 5% to 95% (no condensation) with special packaging to provide additional moisture protection as required.
- Shock — will not suffer damage and fail to perform when subjected to 18 impacts of 5g ± 10% for a duration of 11 ± 1 msec with max "g" force occurring at approximately 5½ msec. The impacts specified shall consist of 3 impacts in each direction along the 3 major axes.
- Vibration — peak displacement of ± .005 in. at the frequency range of 5 to 60 C.P.S. and 2G in the range of 60 to 500 C.P.S. when packed for shipment.

Operating Environment

The HCP systems will be able to tolerate the following conditions in an operational environment.

- Altitude — will operate at any altitude from −1000 ft. to +6000 ft. relative to sea level.
- Temperature — will operate at any ambient inlet temperature in the range of +40°F to +120°F (recommended temperature 75°F) with maximum temperature rate of change of .2° per minute.
- Humidity — will operate at any relative humidity from 10% to 90%. Note: media handled (such as cards) may dictate a more limited range.

Power Requirements

The following are power requirements for the Host Communications Processors, including power for memory expansion, multiplexor expansion and line expansion to the limit of the configuration.

- 2550-1 HCP — 2.6 KVA
- 2550-2 HCP — 4.8 KVA
- 2552-1 HCP — 5.0 KVA

60 Hz and 50 Hz power sources, either 115V or 230V, can be accommodated in accordance with CDC standards.

GLOSSARY OF TERMS

The terms used within this Product Description are listed below. The definition of these terms complies with their use within this specific document.

A

Access Method: Term applied to software within the host computer that facilitates data transfer between the communication system and the host operating system and application programs.

ASCII: American Standard Code for Information Interchange — An 8-bit-per-byte code set that is widely accepted as a standard character and control code set.

Asynchronous: A technique of data transmission that encloses a data byte within bits that perform the task of framing the byte such that the receiving point can detect and understand each bit in its proper relationship to other bits of the byte.

B

Baud Rate: The speed at which a circuit operates. Generally synonymous with bits-per-second (BPS).

BCD: Binary Coded Decimal — Another character code set that specifies 63 characters plus blank.

Bit: The basic element of binary intelligence. May be either a "one" or a "zero."

Byte: A series of bits in combination that either: form a basic unit of human intelligence (character); or is the basic set of bits forming an entity for processing purposes.

C

CCITT: An international standards definition that provides common interfaces between communication facilities and terminals and computer systems.

CCP: Abbr.: Communication Control Program

Channel Coupler: See pg. 7

Circuit: The composition of physical elements that comprises an electrical path over which data or information is moved between points. May be telephone lines, microwave, radio frequencies, or combinations of these.

Circuit Adapters: A device that interfaces between a communication circuit and set of hardware that manipulates the data arriving over the circuit. Adapts the data for use within the context of the hardware.

CLA: See Communication Line Adapter

Clock: Relates to frequency of timing signals used in maintaining bit/byte relationships on synchronous circuits.

Ckts.: Abbr. Circuits

Code Length: Specifies the number of bits per byte

Communication Channel: See circuit

Communication Control Program: See pp. 14-15

Communication facility: See circuit

Communication line: See circuit

Communication Line Adapter: The name applied to CDC's circuit adapter. See pg. 8

Communication Multiplexing Subsystem: The technique of collecting and disseminating data from and to a group of circuits and providing composite handling by a processor/processing system.

Communication Node: See pg. 2

Communication Processor: See pp. 4-5

Computer Network: See pp. 1-2

CP: Abbr.: Communication Processor

CRC: Abbr.: Cyclic Redundancy Check(sum)

Cross System: See pp. 15-16

CRT: Abbr.: Cathode Ray Tube

Cyclic Encoder: See pg. 9
Hardware and software device to generate and verify cyclic redundancy checksum.

D

Demand Driven Multiplexing: The concept used in CDC's new multiplexing subsystem. See pp. 7-8

Destination: The place of final reception for a segment of data.

DMA: Abbr.: Direct Memory Access. See pg. 6

E

Error Rate: The frequency at which a generated error is not detected by the next point in data flow having responsibility for error detection.

F

FDX: Full Duplex

Full Duplex: (1) The technique used to move data over a communication line in both directions at the same time. Always requires a full-duplex communication line. (2) A communication line so arranged to allow simultaneous 2-way data flow.

H

Half Duplex: The technique used to move data over a communication line in both directions, one way at a time.

HCF: Abbr.: Host Communication Function

HCP: Abbr.: Host Communication Processor

HDX: Abbr.: Half-Duplex

HIP: Abbr.: Host Interface Program. See pg. 15

Host Communication Function: See pg. 2

Host Communication Processor: See pp. 4-10

Host Computer: A point in a network where the data processing power is located to perform user application tasks. May act as a data source or destination.

Host Processor: Same as Host Computer

I

IDC: Abbr.: Interrupt Data Channel. See pg. 6

Input data: Generally defined as data coming from terminals and being input to a host computer.

Input Loop: See pp. 9-10

L

Lines: Same as Communication Line

LM: Abbr.: Loop Multiplexer

Loopback: A technique of checking various segments of hardware and communication lines, in a data path, for purpose of isolating a segment causing errors.

Loop Multiplexer: See pg. 7

Loop Multiplexing Subsystem: The name given to CDC's communications multiplexing subsystem. See pp. 7-9

LSI: Large Scale Integration

M

MLC: Abbr.: Multiplex Loop Controller

MLIA: Abbr.: Multiplex Loop Interface Adapter

Modem: Modulator/Demodulator — A device that converts data from a digital form to a form compatible with the communication facility being used as the data path.

MSI: Abbr.: Medium Scale Integration

Multiplex Loop: See pg. 7

Multiplex Loop Controller: See pg. 11

Multiplex Subsystem: Same as Communication Multiplexing Subsystem

Mux.: Abbr.: (1) Multiplex; (2) Multiplexer; (3) Multiplexing

N

NAM: Abbr.: Network Access Method

NCS: Abbr.: Network Communication System

Network Access Method: See pg. 2

Network Communication System: See pp. 1-3

Network Switching: See pg. 2

Node: Same as Communications Node

O

Output Data: Generally defined as data being output from the host computer and destined to a terminal or another host computer.

Output Loop: See pp. 9-10

P

PASCAL: A high-level language. See pp. 16-17

PPU: Peripheral Processing Unit

Protocol: The sequence of control and interaction between devices at each end of a communication line that facilitates the movement of the data with efficiency, accuracy and economy.

R

Redundancy: The hardware and software within a computer network whose primary purpose is to provide back-up protection to other hardware and software elements.

RS232C: A standard specifying the signal level interfaces used widely within the United States.

S

Scan Driven Multiplexing: See pg. 7

Source: The point at which data originates for movement to a destination.

Stop bits: The ending framing bits used in asynchronous operation of a circuit. Varies depending the type of terminal used.

Synchronization: The technique used to maintain continuous, proper bit relationships within bytes using a synchronous transmission technique.

Synchronous: A technique of data transmission that provides transmission of data without framing bits for each byte.

T

TDM: Abbr.: Time Division Multiplexing

Terminal: A composite of devices and a control element that provides human access to computer processing capability.

Terminal Interface: See pg. 2

Throughput: The composite, Input data plus Output data, rate at which data moves between the host computer and the communication system interface.

Time Division Multiplexer: See pg. 8

Timing Signals: Same as synchronization signals.

TIP: Abbr.: Terminal Interface Program. See pg. 15

Transparent: The constraint within hardware and software that insures information will remain unchanged unless specified by the user.

TTY: Abbr.: Teletypewriter