CONTENTS

PAGE

INTRODUCTION 1

COMPUTER NETWORK DEFINITION 2

COMPUTER NETWORK DESCRIPTION AND ITS ELEMENTS 3

TYPES OF COMPUTER NETWORKS 6

WHERE WE ARE TODAY IN COMMON DATA COMMUNICATIONS PROCEDURES 11

WHERE WE ARE TODAY WITH CHANNELS 16

WHERE WE ARE TODAY WITH SWITCHES 16

WHERE WE ARE TODAY WITH ADAPTERS 16

WHERE WE ARE TODAY WITH COMPUTERS 20

WHERE WE ARE TODAY WITH DEVICES 21

WHERE WE ARE TODAY WITH THE USER 21

PROBLEMS TO SOLVE 25

TRENDS 27

CONTROL DATA'S NETWORK ARCHITECTURE 27

AN APPLICATION OF THE NETWORK ARCHITECTURE 30

SUMMARY 30

A LOOK AT COMPUTER NETWORKS
AND
WHERE WE ARE GOING

F. Ken Morioka
Control Data Corporation

INTRODUCTION

The evolution of Computer Networks was primarily caused by the need to access and to output from a computer-based system and employ its resources from geographically separate origins or destinations of information or data. Hopefully, if the network was properly designed, economic justification was derived by effective use of not only the computing resource but the interconnecting communications channels as well.

As these "private" computer networks grew from the early 1960's, it became evident that a better match between the computing resource and available communications channels was needed to fruitfully employ the ever improving and expanding computing resource. Overlapping communications channels, caused by the overlay of one computer network over the other, did not support sound economics and it also became evident that if one could share his resource with others, more leverage of available computer power and its data base could be realized.

Experiments started in the mid-1960's to apply a technology called Packet Switching which appeared to offer a better match between the characteristics of a computer's I/O and communications channels. Based on a concept for digitized voice transmission for Military communications — called the "Hot Potatoe" technique — resource-sharing networks using Packet Switch technology began to evolve. The United States ARPA Net and the United Kingdom NDL Net are examples of networks which employ Packet Switching technology and they are continuing to develop and expand its network capabilities. Today, Value Added Carriers — or VAN's — offer similar technologies for the movement of resource-sharing information.

Bandwidth, or the capacity of the channels used to transport information, is — even yet today — highly restricted. This restriction is caused by the need to employ transport channels which were largely designed for voice communications. We are tempted to use these channels because of their availability over a wide geographical area.

However, these channels are provided by governmentally-regulated common carriers and can only be employed in a disciplined manner. It wasn't until June of 1968, when the famous Carterfone Decision was made, that others outside of the regulated carrier business were allowed to attach "foreign" equipments to these regulated channels. Subsequent to that Decision, we have seen the steady improvement, at reduced cost, in the bandwidth capacity of these highly-dispersed channels. However, even today, the digital-to-voice channel "adapters", or modems, are largely supplied by the Bell System domestically. The ratio for private, leased circuits is about 45% to 65% outside-vendor-to-Bell and the switched network is about 2% to 98% for "foreign" equipment attachments.

Bandwidth has recently undergone further increases with the advent of not only common carrier, but specialized carrier microwave channels. Since these "Radios" operate at very high frequencies in the gigacycle range, one is able to derive much more transport bandwidth. These channels are channelized to derive voice, data and video (wide-band) channels. Digital data transport channels, with the ability to transmit up to 1.544 megabits per second, are available today.

Satellites extend the microwave channels over longer paths. Synchronous Satellites of today can supply up to twelve channels at 36 mhz bandwidth each, or 100,000 voice bandwidth channels. When compared to Terrestrial Channels, the Satellite Channels offer better binary error rate performance. However, because of their distance above the earth, we must pay a propagation delay penalty of about 500 to 700 milliseconds round-trip delay. Also, it would require three synchronous Satellites to provide worldwide coverage, thereby extending the propagation delay problem.

If we consider the available communications channels, including Satellite extensions and how we might best utilize them, we must review technology and see what's happening and how these changes might affect the Evolution of Computer Networks.

It is the purpose of this paper to review Computer Network Technology today, the problems we must solve in the future to realize effective use of Computer Networks, Control Data's Network Architecture towards addressing these problems, and then, briefly review an application of this Architecture.

COMPUTER NETWORK DEFINITION

It's often nice to define what it is you are going to discuss. A search of the technical library and other common reference material has revealed that no real definition exists for what we might call a "Computer Network".

Figure 1 illustrates our American National Standard definition and a revision to it suggested by the author. The point expressed here is not only are we constrained by technology, we must continuously assess change and economics in the Computer Network Evolution we face.

COMPUTER NETWORK DEFINITION

THIS IS A COMPLEX CONSISTING OF TWO OR MORE INTERCONNECTED COMPUTERS

—AMERICAN NATIONAL STANDARD
VOCABULARY FOR INFORMATION
PROCESSING

PERHAPS

A SOMEWHAT STRUCTURED CONGLOMERATION OF DIGITAL COMPUTER-BASED SYSTEMS, HUMAN OR SUBSCRIBER INTERFACE DEVICES AND INTERCOMMUNICATIONS CIRCUITS WHICH PERFORMS INFORMATION STORAGE AND RETRIEVAL, PROCESSING, TRANSMISSION AND/OR EXCHANGE TO ACHIEVE A DESIRED SET OF RESULTS WITHIN A DYNAMIC ENVIRONMENT CONSTRAINED BY GEOGRAPHY, SUPPLY AND DEMAND, LAWS, AND RESOURCES INCLUDING MONIES.

F. K. MORIOKA, 1975

FIGURE 1.

COMPUTER NETWORK DESCRIPTION AND ITS ELEMENTS

Simplicity often has its virtue in that it allows us to restrict our communications to a small set. A small set is usually easier to comprehend than a large, complex set. Figure 2 is an attempt to simply illustrate the elements which make-up or are used in a Computer Network. We might call the illustration a Circular Quad Gram, consisting of elements called CHANNELS and/or SWITCHES, ADAPTERS, COMPUTERS and DEVICES, and the USERS.

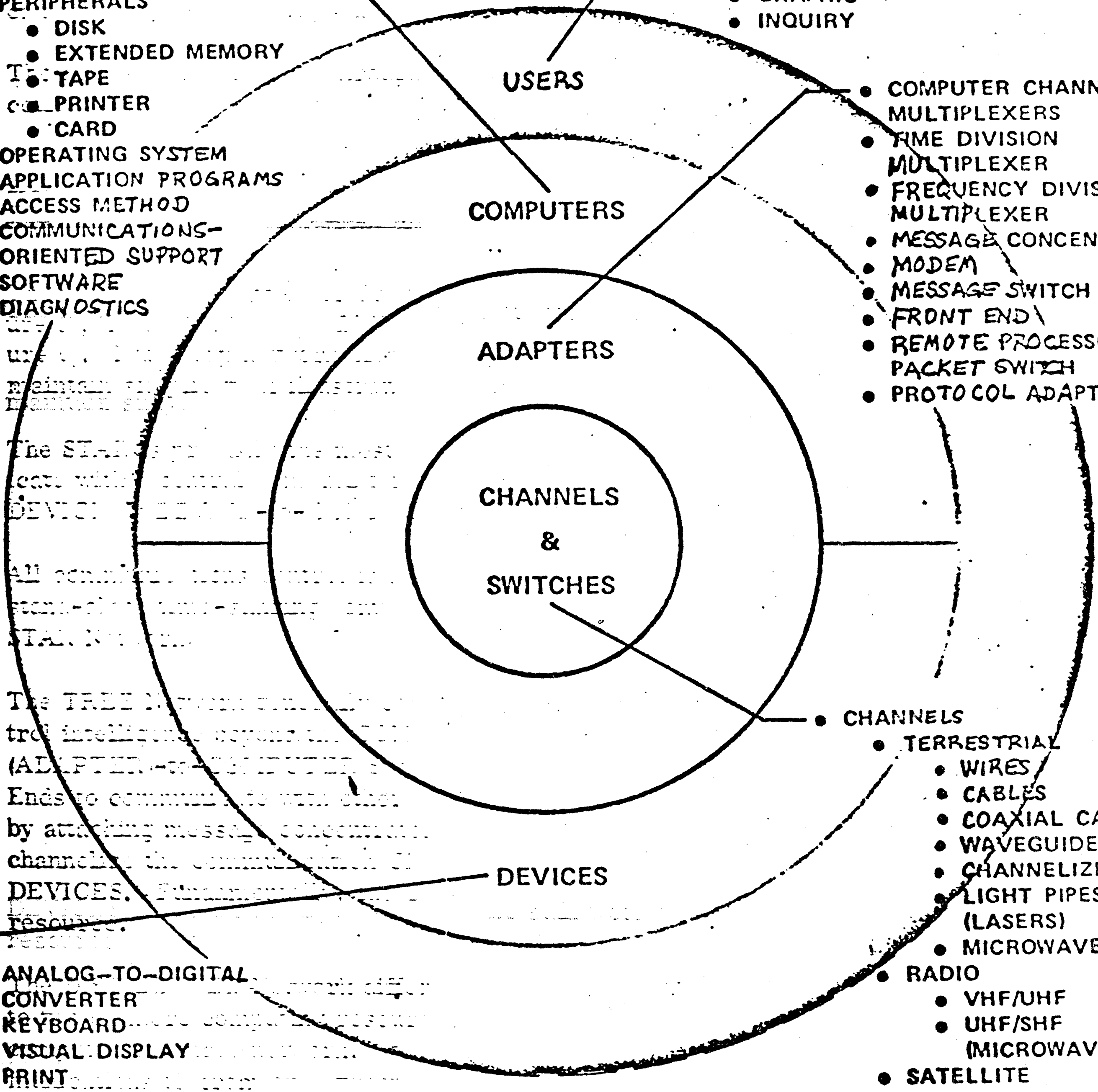
Although not intended to be a comprehensive and complete listing of all possible elements, please note the following definitions of those illustrated:

- CHANNELS are Terrestrial (earthbound) or Radio (Satellite being a Radio repeater).
- SWITCHES are basically electromechanical or electronic connectors of channels.
- ADAPTERS are those elements which adapt the characteristics of computer and device channels to communication channels.
- COMPUTERS are conventional micro, macro, or maxi computers, including their peripherals, software, and applicational programs.
- DEVICES interface the USER within the Computer Network. They can be fixed-wire or programmable (equal to or less than a computer).
- USER applies the Computer Network via the DEVICES which interface with ADAPTERS. CHANNELS and/or SWITCHES and/or the COMPUTERS or other DEVICES interface with ADAPTERS.

If one carefully analyzes the probable combinations, ADAPTERS could cross boundaries. As an example, if the modem is supplied by the CHANNELS and SWITCHES vendor, the modem could be identified as part of the Inner Circle. On the other hand, if the modem is included in the DEVICE vendor's hardware, the ADAPTER crosses into the DEVICE World. However, logically, the modem belongs in the ADAPTER World.

It is assumed that most readers fundamentally understand the characteristics of the identified network elements. However, an analysis of the listed ADAPTERS will show that they are truly "things" that adapt the COMPUTERS and DEVICE communications characteristics to the CHANNEL and SWITCH characteristics. An example here could

COMPUTER NETWORK DESCRIPTION AND ITS ELEMENTS



- CPU
- MEMORY
- CHANNEL(S)
- PERIPHERAL CONTROLLER
- PERIPHERALS
  - DISK
  - EXTENDED MEMORY
  - TAPE
  - PRINTER
  - CARD
- OPERATING SYSTEM
- APPLICATION PROGRAMS
- ACCESS METHOD
- COMMUNICATIONS-ORIENTED SUPPORT SOFTWARE
- DIAGNOSTICS

- RECORD (MESSAGE)
- INTERACTIVE
- TRANSACTION
- BATCH
- GRAPHIC
- INQUIRY

- COMPUTER CHANNEL MULTIPLEXERS
- TIME DIVISION MULTIPLEXER
- FREQUENCY DIVISION MULTIPLEXER
- MESSAGE CONCENTRATOR
- MODEM
- MESSAGE SWITCH
- FRONT END
- REMOTE PROCESSOR – PACKET SWITCH
- PROTOCOL ADAPTERS

- CHANNELS
  - TERRESTRIAL
    - WIRES
    - CABLES
    - COAXIAL CABLES
    - WAVEGUIDES
    - CHANNELIZERS
    - LIGHT PIPES (LASERS)
    - MICROWAVE
  - RADIO
    - VHF/UHF
    - UHF/SHF (MICROWAVE)
  - SATELLITE
    - UHF/SHF (MICROWAVE)
- SWITCHES
  - ELECTROMECHANICAL – CIRCUIT
  - ELECTRONIC – CIRCUIT
  - MESSAGE SWITCH – RECORD

- ANALOG-TO-DIGITAL CONVERTER
- KEYBOARD
- VISUAL DISPLAY
- PRINT
- READ
  - PUNCHED CARD
  - MAGNETIC TAPE
  - OPTICAL
  - CREDIT CARD
- GRAPHIC
- REMOTE COMPUTER SYSTEM

FIGURE 2.

be a modem (modulator/demodulator) which basically adapts digital signals to analog signals (and vice-versa) to adapt digital COMPUTER and DEVICE communications channels to analog CHANNELS and/or SWITCHES transport media.

The next set of Figures 3 through 6 illustrate some basic types of networks we encounter today.

TYPES OF COMPUTER NETWORKS

The more common types of Computer Networks today are generally called STAR (Figure 3), TREE (Figure 4), DISTRIBUTED (Figure 5), and FULLY-CONNECTED (Figure 6). Not every network element used in each configuration is shown in order to maintain simplicity of illustration.

The STAR is probably the most common type of network. DEVICES basically communicate with a central computing resource. Communications are generally COMPUTER-to-DEVICE or DEVICE-to-COMPUTER with little or no DEVICE-to-DEVICE communications.

All communications control is handled by the central computing resource. A simple, stand-alone time-sharing computer system with remote terminals is an example of a STAR Network.

The TREE Network generally consists of one or more levels of communications control intelligence beyond the COMPUTERS channel. An example here is a Front-End-(ADAPTER)-to-COMPUTER System which controls communications via other Front-Ends to communicate with other COMPUTERS or DEVICES. The TREE grows larger by attaching message concentrator ADAPTERS to the Front-End ADAPTERS to further channelize the communication CHANNELS and/or SWITCHES for connecting additional DEVICES. Fundamentally, the DEVICES still work with only one, central computing resource.

The DISTRIBUTED Network differs from the TREE in that we are able to communicate to two or more computing resources which are geographically separated. In a multi-computer resource network, we may restrict communications channels to allow communications to/from or between a selected set, or the network is not FULLY-CONNECTED, but DISTRIBUTED.

The FULLY-CONNECTED Network is a DISTRIBUTED Network with a full set of interconnections; i.e., all DEVICES and COMPUTERS could independently communicate with each other if desired without traversing the other's private connections.

TYPES OF NETWORKS

STAR

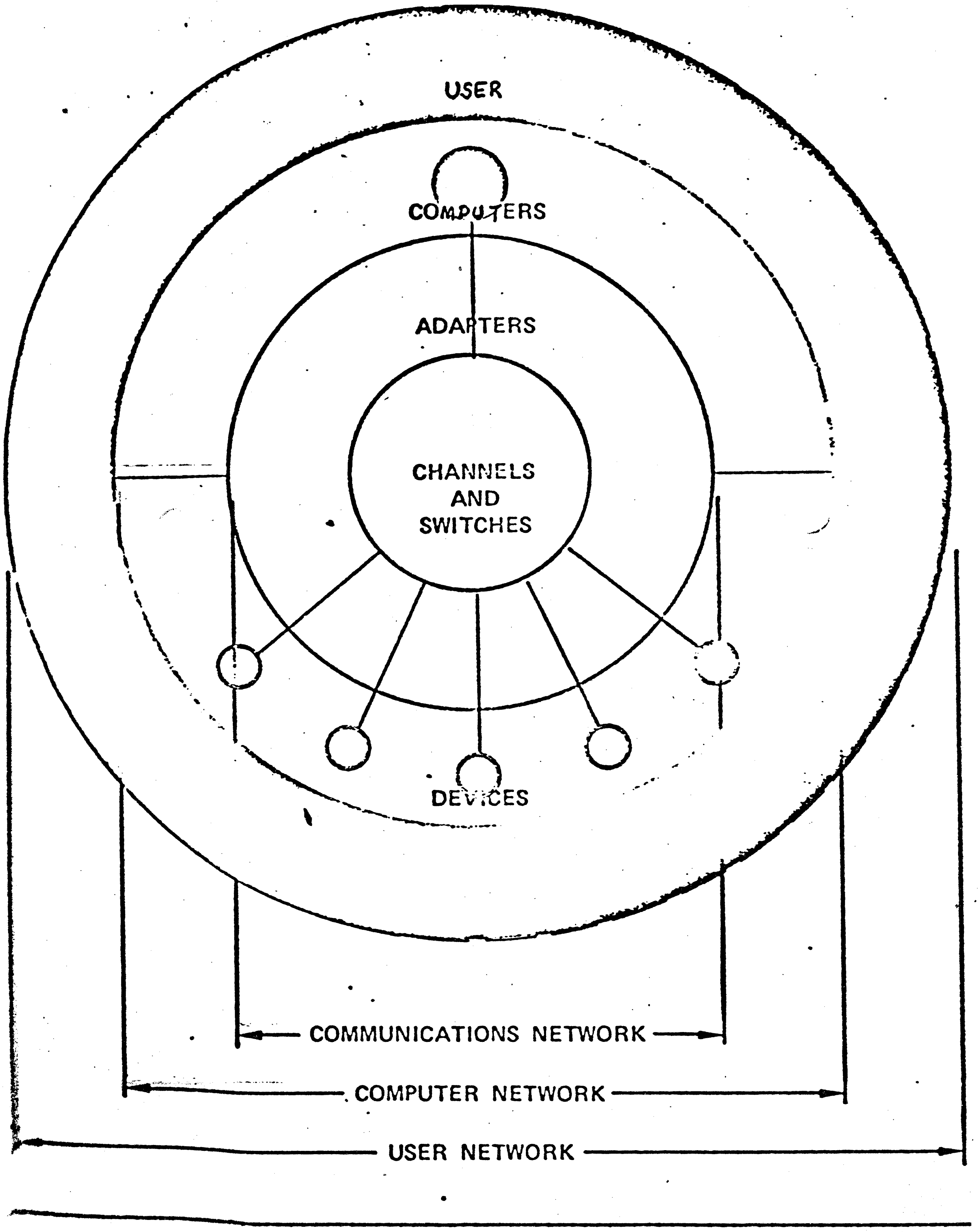
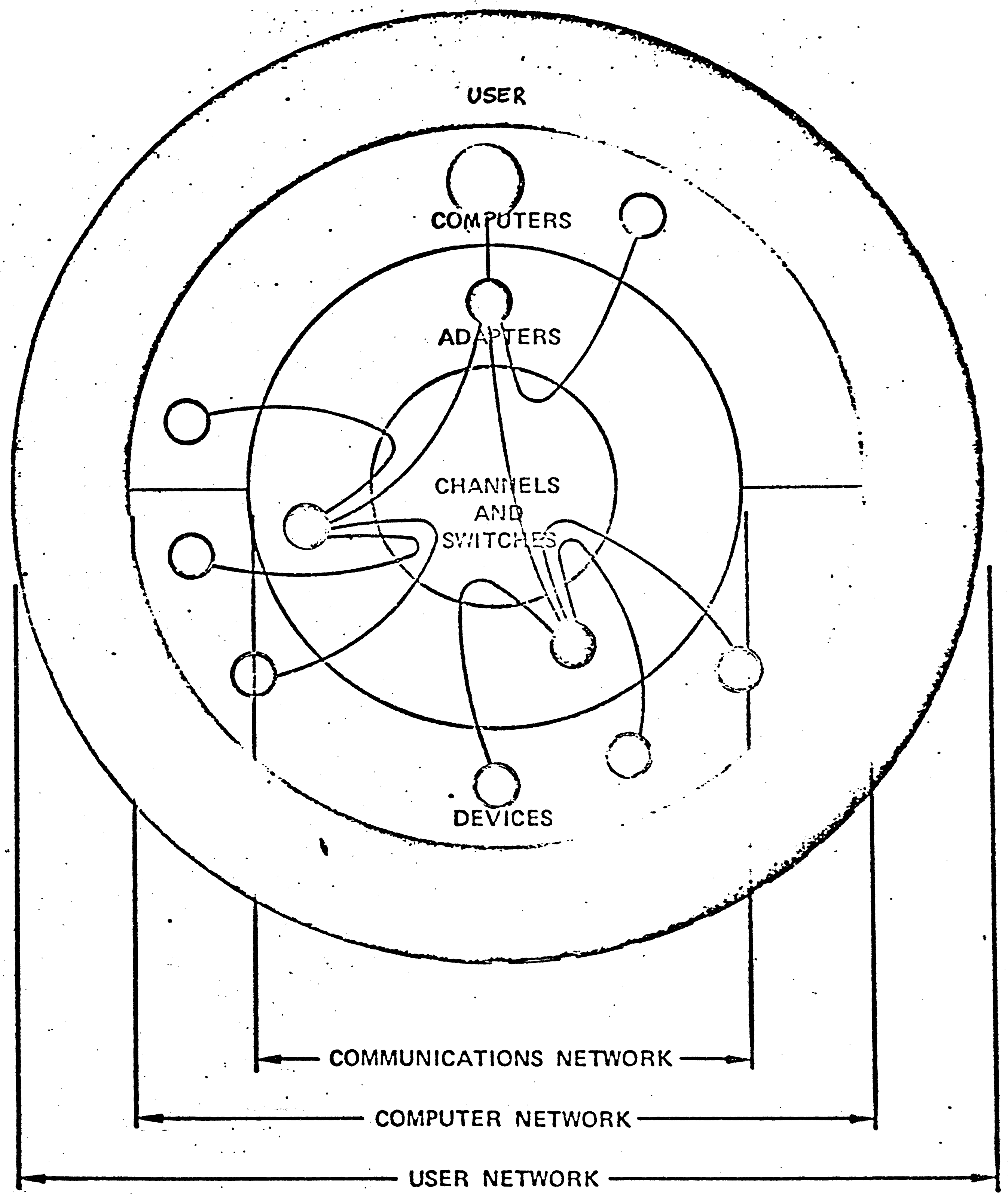


FIGURE 3.



TREE

FIGURE 4.

TYPES OF NETWORKS

DISTRIBUTED

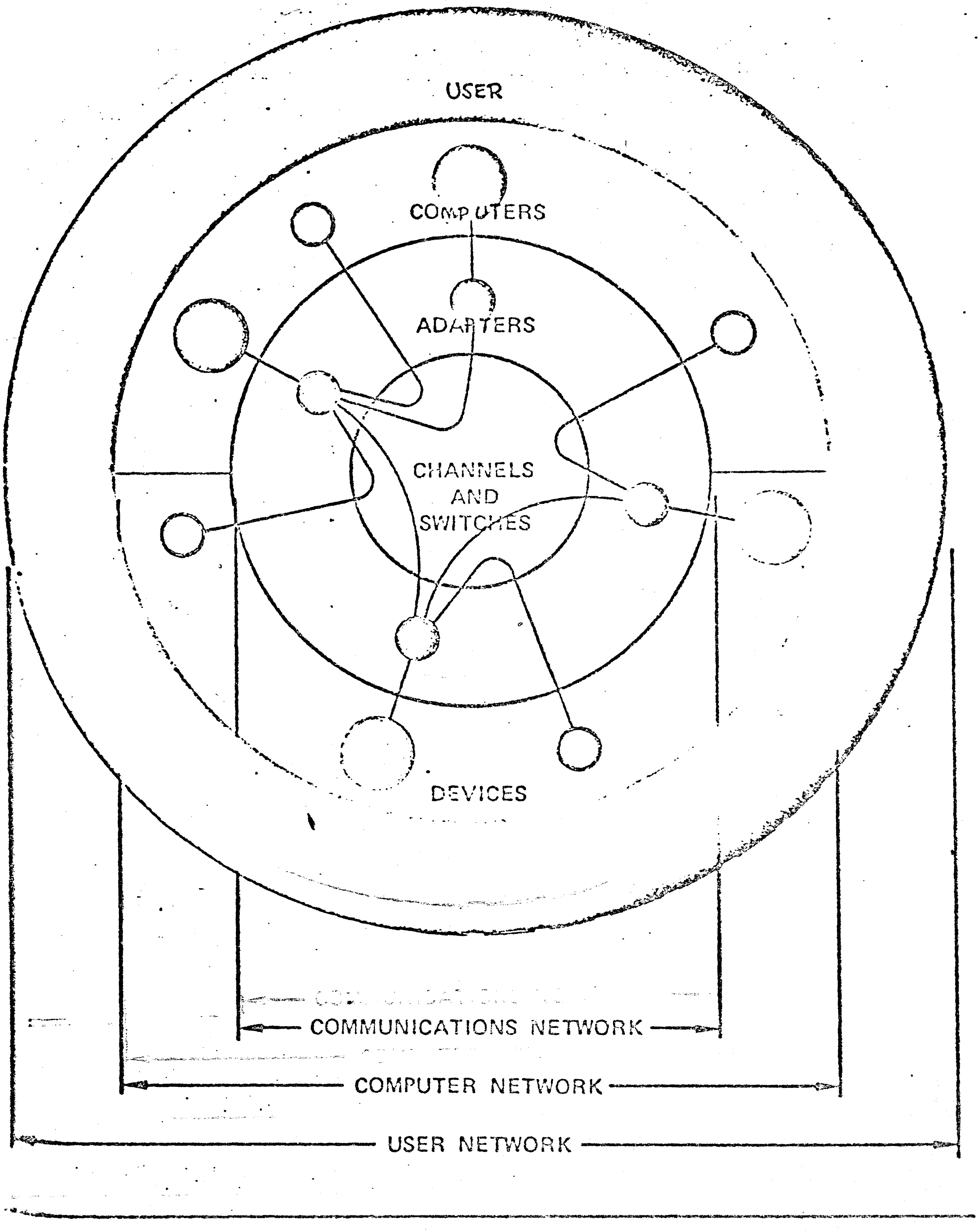


FIGURE 5.

TYPES OF NETWORKS

FULLY-CONNECTED

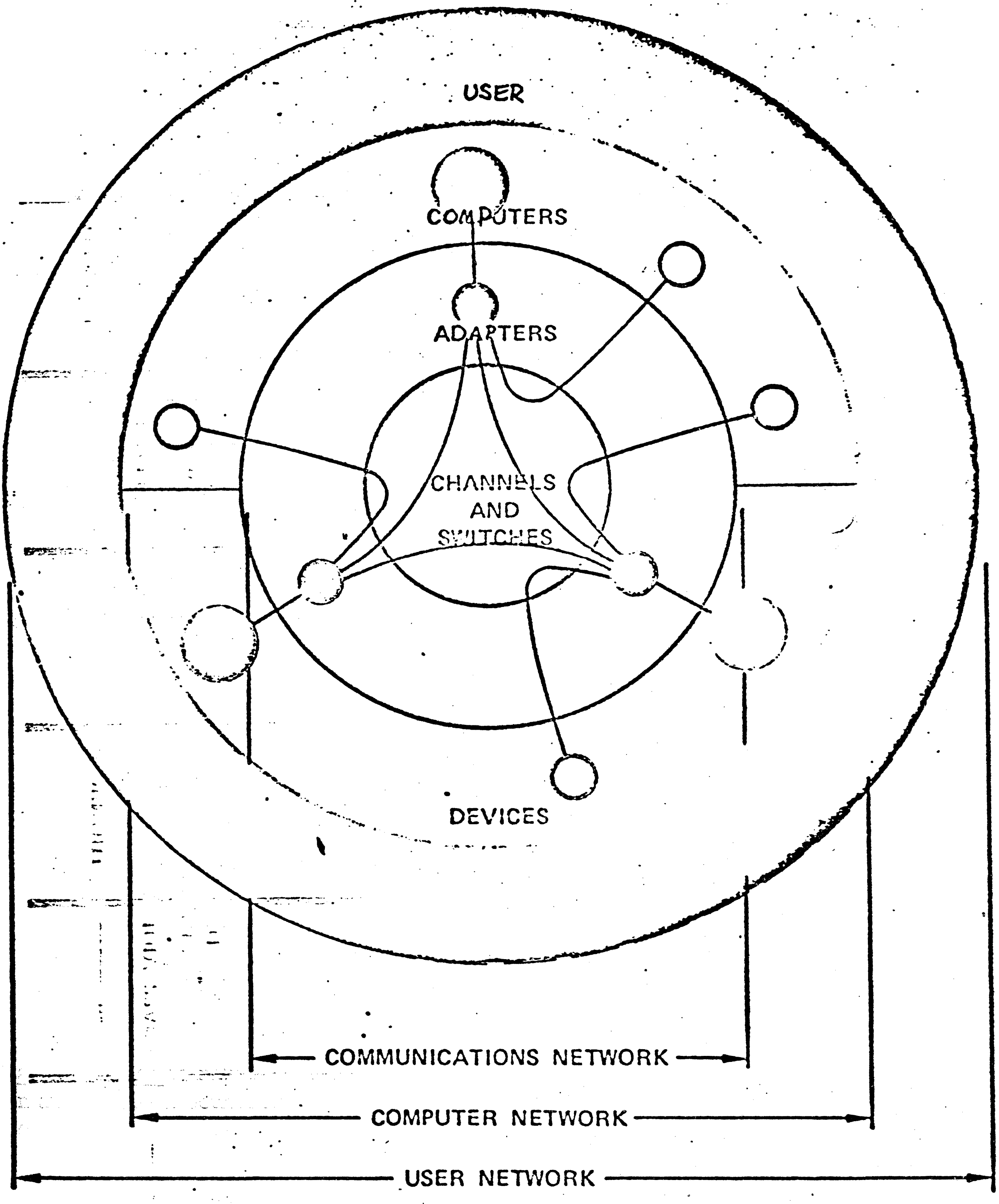


FIGURE 6.

A LOOK AT COMPUTER NETWORKS AND WHERE WE ARE GOING

The ARPA Net is an example of a DISTRIBUTED Network. Our Airlines Reservations Systems, which connect to each other on a selected basis, is another example.

The ideal network configuration for any community of interest or a set of communities of interest depends on the USER needs and economics. To assess where we are today and then understand what it will require to achieve effective computing resource-sharing, we will now briefly review where we are today and the problems we need to solve. It is becoming evident that if we are going to approach effective use of our computing resource base, we must address computer networking.

WHERE WE ARE TODAY IN COMMON DATA COMMUNICATIONS PROCEDURES

One of the basic reasons we are able to effectively apply our telephone network for voice communication today is that we follow common procedures with a highly-disciplined network. If we don't dial a number correctly, we will either receive a "not in service" message, or get a "wrong party", and we hang up and perhaps redial. If the network cannot accept our call because of a peak load, we will receive the famous "busy" signal. These procedures have evolved over a four-decade period. Any change to these procedures takes time and education.

Voice communication is also narrowband, redundant, and highly tolerant to distortion. Crosstalk, dial clicks, fades, and echos are tolerated by our redundancy and instant repeatability.

Figure 7 and Figure 8 illustrate the bandwidth needed for speech and how much frequency spectrum we employ for voice.

Digital computers require wideband channels because of their wideband signals (rectangular waves) and their speeds. They are also very sensitive to binary or bit errors, especially if we want to transport programs.

To achieve communications via CHANNELS and/or SWITCHES media, we basically add overhead to raw information to check for and correct errors and identify information stream boundaries. We also employ ADAPTERS to match these wideband computer channels to narrow or narrower than ideal CHANNELS and/or SWITCHES.

The methodology used to transport data over the CHANNELS and/or SWITCHES media is usually called Procedures and a complete Procedure includes many levels of control and the interface requirements.

NETWORK ELEMENTS
CHANNELS

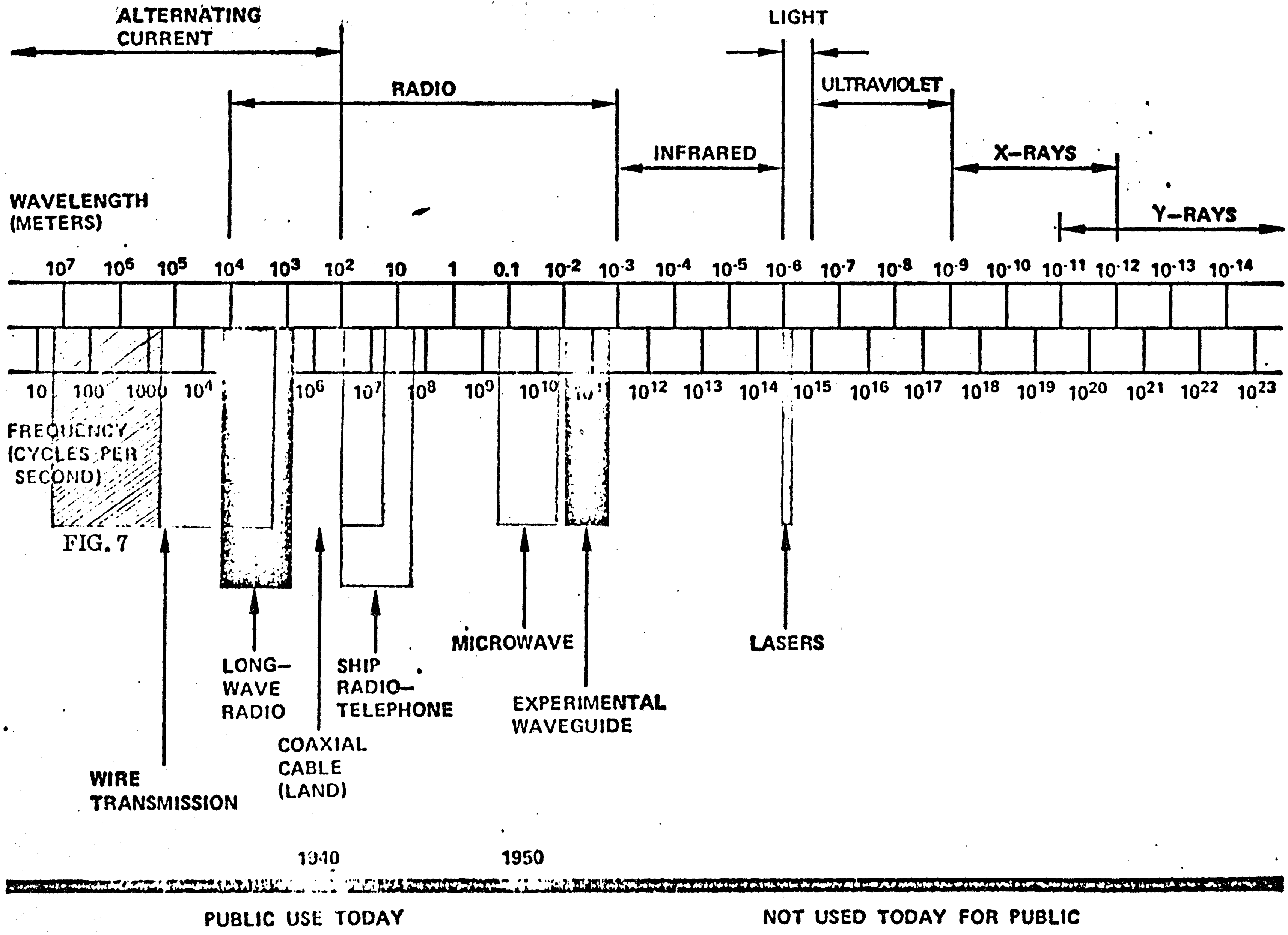


FIGURE 8.

Figure 9 shows the hierarchial Levels of Control which make up a complete Procedure. If we examine where we are today, in terms of Industry Standard or Common Data Communications Procedures, we are just beginning to achieve commonality. Some of this evolution is forced by the need to better match CHANNELS and/or SWITCHES characteristics such as the ability to tolerate propagation delays caused by Satellite Channels while maintaining high channel utilization to our COMPUTER and DEVICE channel characteristics.

Line (CHANNEL) Control and Communication Channel Interface Procedures are becoming quite standard today. Hence, we are able to buy ADAPTERS from many sources with common interface and line control for connection to EIA RS232, MIL STD 188, and CCITT V which are fairly well accepted throughout industry. The evolution period is roughly 1963 to 1969 — or six-plus years. We must recognize that these "Standards" will evolve in time to accommodate new technology. Link (one or more channels) Control, Device Control, Message Control and End-to-End Procedures have a ways to go. Binary Synchronous Control, or BSC, was announced by IBM in about the 1969 time-frame. BSC is fundamentally a character-oriented Procedure and operates in what we might call a "Halt and Wait" sequence; i.e., send something and wait for an answer before you send the next thing. We also call it a half-duplex communication, or two-way control procedure, with only one-way communication at a time over a two-way channel. This Procedure works adequately for Terrestrial Channels, but is questionable if we add more propagation delay caused by Satellite Channels.

In 1973, IBM announced SDLC or Synchronous Data Link Control Procedure. This Procedure is called a bit-oriented control procedure and improves the usage of a communications channel. It offers more flexibility to the bit-level as opposed to the character-level, is full-duplex, i.e., two-way simultaneously, and almost ideal if we include propagation delay times. It should be pointed out that this type of Procedure needed to evolve to progress in the effective use of available bandwidth. Other examples of similar Procedures are the Burroughs BDLC, announced in 1975; the ANSI ADCCP (Advanced Data Communication Control Procedure); ISO HDLC (High-level Data Link Control); NCR's BOLD (Bit Oriented Link Discipline); and CDC's CD/CCP (Communications Control Procedure).

Device Control Common Procedures with bit-oriented Link Control Procedures will probably become fairly common in about the 1977-1978 time-frame. Hence, we will all be faced with ADAPTION to many different types of LINK and DEVICE Control Procedures for a few years yet. Message and End-to-End Procedures will probably occupy a five-year period, if not more, to achieve full commonality. A few "Standards" exist by Industry for Message and/or End-to-End Control. Federal Reserve System's BOPEAP (Bank Oriented Processor Ender Accountability Procedure) and the Airlines Industry ATA/IATA Message Handling Procedures are examples. An early agreement for Common Data Communication Procedures will certainly advance our abilities to construct effective resource-sharing Computer Networks.

We will now briefly review our situation with other network elements.

WHERE WE ARE TODAY WITH CHANNELS

CHANNELS for transport of data or information in a Computer Network can be basically called "Terrestrial" or "Radio". Terrestrial represents the channels which are earthbound (includes ground microwave radios). Radios transmit data or information via electromagnetic waves which are modulated by various technologies to transport data or information. Transmission may be via "hops" (ionospheric, tropospheric, etc.) or line-of-sight. Figure 10 highlights their characteristics. Essential things to remember are: (1) bandwidth, (2) error characteristics, cost, and propagation delay. Note that Satellite Channels cause a delay time ten times that of typical Terrestrial Channels.

WHERE WE ARE TODAY WITH SWITCHES

SWITCHES are those elements which can manually, semiautomatically, or automatically connect CHANNELS to each other to achieve communications. Figure 11 illustrates a few of the basic types used today or beginning to increase in population.

Electronics has extended the number of terminations capability to microwave and waveguide type channels and it has also improved connect times or the time required to connect called to caller and disconnect when through. We are beginning to find greater use of the Public Switched Network because of its economics and access-point availability. However, as a good data communications media, the Switched Channel is generally noisier, narrower in bandwidth, and less reliable when compared to a privately-leased channel.

Circuit SWITCHES are included with the CHANNEL elements as almost 99% is provided by common carriers.

WHERE WE ARE TODAY WITH ADAPTERS

ADAPTERS, identified in Figure 12, are probably undergoing the most rapid change. Microelectronics has allowed the hardware designer to significantly reduce ADAPTER costs. As an example, 2400- to 4800-bit-per-second data modems have decreased in cost approximately $1/bit-per-second to $0.125/bit-per-second in the past few years.

WHERE ARE WE TODAY

CHANNELS

- TERRESTRIAL
  - BELL SYSTEM – VOICE NETWORK
    - SHANNON'S THEORY, 18,000 BITS/SEC
    - PRACTICAL LIMIT, 9,600 BITS/SEC
    - LOCAL DISTRIBUTION
  - OTHERS
    - MICROWAVE LINKS
    - MORE BANDWIDTH
    - LOWER LONG–HAUL COSTS
  - PROPAGATION DELAY, LESS THAN 70 MILLISECONDS COAST–TO–COAST
- RADIO
  - MILITARY, GOVERNMENT, LAW ENFORCEMENT, EMERGENCY ONLY
  - NARROW BANDWIDTH, EXPENSIVE, NOISY
- SATELLITE
  - EXTENSION OF LONG–HAUL TERRESTRIAL MICROWAVE
  - ADDS MORE CHANNELS, LOWER COST/ CHANNEL
  - INTELSAT IV – 12 CHANNELS OF 36 MHZ BANDWIDTH
  - BINARY ERROR RATE LOWER COMPARED TO TERRESTRIAL CHANNELS
  - REQUIRES ROUTING TO AN EARTH STATION
  - USE BELL LOCAL DISTRIBUTION
  - PROPAGATION DELAY APPROXIMATELY 10 TIMES TERRESTRIAL CHANNELS $\approx$ 700 MS

FIGURE 10.

WHERE ARE WE TODAY

SWITCHES

- ELECTROMECHANICAL
  - OLD
  - SLOW CONNECT TIME
  - MANY, MANY INSTALLED
  - ADEQUATE FOR PHONES
  - NOISE GENERATORS
- ELECTRONIC
  - NEW
  - FAST CONNECT TIME
  - FEW INSTALLED
  - MORE CHANNELS, LOWER COST/CHANNEL
  - TANDEM LINKS CAN INCREASE CONNECT TIMES
- ELECTRONIC PABX
  - NEWEST
  - LOCAL DISTRIBUTION
  - VOICE/DATA CHANNEL SHARING
  - EXTENDS BELL SWITCH RANGE
  - VERY FEW INSTALLED

FIGURE 11.

WHERE ARE WE TODAY

ADAPTERS

- MODEMS, TIME DIVISION AND FREQUENCY DIVISION MULTIPLEXERS, FRONT ENDS, MESSAGE CONCENTRATORS, PROTOCOL ADAPTERS, COMPUTER CHANNEL MULTIPLEXERS, PACKET SWITCHERS, AND MESSAGE SWITCHERS
  - GREATER USE OF MICROELECTRONICS TECHNOLOGY
    - IMPROVED RELIABILITY
    - LOWER COST
  - GREATER USE OF MINI– AND MICROCOMPUTER TECHNOLOGY
    - PROTOCOL HANDLERS
    - INTERFACE DEVICES
    - RELIABILITY
    - FLEXIBILITY BY FIRMWARE
- HARDWARE IMPLEMENTATION TECHNOLOGY FAR AHEAD OF SOFTWARE TECHNOLOGY
  - MICROPROGRAMMING
  - ASSEMBLY IMPLEMENTATION LANGUAGE
  - MACRO IMPLEMENTATION LANGUAGE

FIGURE 12.

A LOOK AT COMPUTER NETWORKS AND WHERE WE ARE GOING

With the introduction of micro- and mini-computers, we are now able to construct lower-cost Front-Ends, Message Concentrators, and Multiplexer ADAPTERS. Figure 12 summarizes ADAPTER Status today.

Message Switchers are identified as ADAPTERS because they fundamentally can convert transport speeds, language code sets, edit, route, queue, hold and retrieve data for information. They basically adapt DEVICE and/or COMPUTER traffic to channels when the outgoing channel is available and assume responsibility for delivery.

What restricts quick changes to Message Switchers is that they rely quite heavily on not only hardware reliability, but — more importantly — specialized software to perform much of the logic and tolerance to errors caused by CHANNELS, ADAPTERS, COMPUTERS and DEVICES.

A reliable Message Switcher will have approximately 20% Switch Logic and 80% Protective or Error Handling Logic. Hence, progress in this area will rely heavily on programming technology which will take time or emulation by the use of newer, faster hardware to leverage the software.

Advances in the hardware area clearly surpass our software technology. It is probably fair to assume that if we are going to achieve reasonable life-cycle costs, we must improve our software technology in the ADAPTER area.

WHERE WE ARE TODAY WITH COMPUTERS

Similar to the ADAPTER situation, microelectronics technology has improved the cost/performance of our COMPUTERS and their Peripherals including main Memory. However, most "fourth-generation" machines are basically "emulators" in a sense that a USER can progress to a better cost/performance COMPUTER without extensive changes to his software library. Hence, the functional limitations in the software architecture carries forward.

basically [illegible]

To achieve true COMPUTER networking, we must restructure the software in such a manner that we can interface and apply common Control Procedures. The capability to connect COMPUTER resources from a hardware sense exists today — we merely need to decide whether we insert some of the necessary procedural functions within a COMPUTER's software or its ADAPTER.

If the USER [illegible]

Each existing Operating System and its support software, such as "Access Methods", limits true, reliable networking. These limits occur in such areas as procedural incompatibilities, address range limit, queuing limit and processor time available limit.

argum[illegible].

Clear functional separation of network interface, supervision, and control is needed in the future to allow evolution of network technology. The Programmer/USER is then only concerned with an Address and Content for an inquiry, message, job, etc. The COMPUTER Network handles timely delivery to whatever computing resource will accept his task and his Address.

Figure 13 highlights our situation today with COMPUTERS.

WHERE WE ARE TODAY WITH DEVICES

Again, similar to the ADAPTERS and COMPUTERS situations, microelectronics applied to micro- and mini-computer architectures has opened up new DEVICE capabilities. The greatest advance we can make in this area is in the area of common Data Communications Procedures so that a DEVICE can effectively connect to any Computer Network.

Until we are able to fully agree on a fundamental Procedure for DEVICE and Link Control, we will probably be faced with a wide range of Procedures (Protocols) through the 1970's, which will certainly keep the ADAPTER Market active. Figure 14 summarizes the problems with DEVICES of today.

WHERE WE ARE TODAY WITH THE USER

The USERS of today generally are dealing with Centralized or TREE-structured COMPUTER Networks which, in some cases, are geographically distributed but not necessarily interconnected in a true resource-sharing manner. An ARPA-type Network approaches the COMPUTER Network concept.

The trend appears to be toward centralization of resources in many environments to basically reduce Management Staff, Support Staff, Channel, and Redundant Computing Resource costs. However, there are those situations where we might centralize particular communities of interest to gain economies, but we must interconnect with other communities of interest to achieve total economics. To answer the latter, the USER must address COMPUTER Networking.

If the USER community can solve the Security issue, we could then share our computer power, or have access to power, whenever we needed it. As an example, shift work overload to a Service Bureau at peak work-load situations rather than have excess margin (costly) on standby. Figure 15 highlights the Centralized versus the Decentralized arguments.

WHERE ARE WE TODAY

COMPUTERS

- SOFTWARE
  - MULTIPROGRAMMING CAPABILITY
  - NETWORK AND COMMUNICATIONS SUPERVISION TIGHTLY COUPLED WITH ACCESS METHOD AND APPLICATIONS PROGRAMMING
  - LIMITED NETWORK DEFINITION LANGUAGE
  - EXTENSIVE USE OF ASSEMBLY LANGUAGE
  - OFF–LINE–ORIENTED
    - AUTOMATIC RESTART
    - RECOVERY
  - LIMITED USE OF HIGH LEVEL IMPLEMENTATION LANGUAGE
  - EXPENSIVE TO IMPLEMENT/SUPPORT
  - REQUIRES CONTINUED EVOLUTION/MAINTENANCE
  - FLEXIBLE
  - LIMITED FEATURES FOR INCREASING AVAILABILITY
    - MTBF
    - MTTR
- HARDWARE
  - INCREASED SPEED
  - LOWER COST STORAGE
  - HIGHER RELIABILITY
  - LOWER LIFE CYCLE COST

FIGURE 13.

Where Are We Today

Devices

- Broad Range of Protocols
  - Character-oriented
  - inefficient use of channel capacity
  - inflexible to procedural change

REDUND

APACITY

ANGE

Figure 14

WHERE ARE WE TODAY

USER

- DECENTRALIZED COMPUTING RESOURCES
  - INHERITED BY BUSINESS
  - OVERLAPPING DATA COMMUNICATIONS NETWORKS
  - REDUNDANT HARDWARE
  - UNDERUTILIZED RESOURCES, NOT EASY TO SHARE
  - DIFFERENT SOFTWARE TO SUPPORT
  - DIFFICULT TO EFFECTIVELY MANAGE
  - REDUNDANT STAFF
  - NO UNIFORMITY
    - OPERATIONAL PROCEDURES
    - SECURITY PROCEDURES
  - DISASTER PROTECTION

- CENTRALIZATION OF COMPUTING RESOURCES
  - LESS STAFF
  - CENTRAL MANAGEMENT AND SUPPORT
  - UNIFORM SYSTEM BY EDICT
  - CAPACITY LIMITED PERHAPS
  - REQUIRES LONG-HAUL ACCESS CHANNELS
  - NO REDUNDANCY OF HARDWARE
  - IMPROVED UTILIZATION
  - MAY HAVE DIFFERENT SOFTWARE TO SUPPORT
  - LARGE DATA BASE TO MANAGE
  - NO DISASTER PROTECTION

FIGURE 15.

Hence, the USER today must strive for data communication commonality in the future to achieve the ability to truly resource-share. However, he must recognize the evolution period and the time required to achieve adequate commonality — or, the point is — "Standard Link Control" Procedures is only a small part of the overall problem we face.

PROBLEMS TO SOLVE

The fundamental problem we must solve — to achieve true Computer Networking, as presented in this paper — is to achieve commonality of Communications Procedures. To achieve this commonality, we are basically addressing software technology. The hardware to accomplish the task exists today.

Other problems which we must address are procedural flexibility, education, and separation of functions.

Procedural flexibility allows one to adjust to procedure evolution, which we know is going to occur over quite a period of time.

Education improves our ability to track the evolution in an orderly manner. "Unpleasant Surprises" are minimized.

Interrelated with the education process is the decision-making process. There are many failures — at great expense — we can look at in the networking process because somebody would not make a firm decision during the specification, planning and implementation phases. All too often we prepare "loose as a goose" requirements documents. However, will we penalize the implementor because he did not do the job?

Finally, it should be kept in mind that Computer Network flexibility and growth can be easily achieved if we can clearly identify and control independent functions, i.e., a change in one function does not alter a change in another. A careful delineation between storage and retrieval, processing, and communications functions will enhance our abilities to achieve the necessary independencies for orderly growth and adjustment to environmental (Procedural) changes. Figure 16 highlights these points.

PROBLEMS TO SOLVE

- COMMON COMMUNICATIONS PROCEDURES MUST EVOLVE
  - ALLOW BROADER CONNECTABILITY
  - MAKE MORE EFFECTIVE USE OF BANDWIDTH
    - BIT–ORIENTED PROTOCOL
    - EFFECTIVE DATA INTERMIX
    - IMPROVED CONNECT TIME
  - IMPROVE MESSAGE INTEGRITY
  - IMPROVE SECURITY MEASURES
  - NEEDED TO ALLOW RESOURCE SHARING
- MORE ATTENTION TO DEVICE SELECTION AND ITS PROTOCOL NEEDS
  - REDUCE QUANTITY – COSTLY TO SUPPORT
  - IMPROVE QUALITY – MORE EFFECTIVE USE
  - MAINTAIN FLEXIBILITY FOR PROCEDURAL CHANGES
- IMPROVED TRAINING AND DECISION MAKING PROCESS
  - USER NEEDS AND CHANGES
  - TECHNOLOGICAL CHANGES
- BETTER PLANNING FOR ACHIEVING FUNCTIONAL INDEPENDENCY
  - DATA BASE MANAGEMENT
  - DATA PROCESSING
  - COMMUNICATIONS MANAGEMENT
  - EACH CAN GROW, ADAPT WITHOUT AFFECTING OTHER

FIGURE 16.

A LOOK AT COMPUTER NETWORKS AND WHERE WE ARE GOING

TRENDS

Figure 17 summarizes some trends as envisioned today. The significant cost item is suggested to be in the software development area. Until we are able to reduce programming to a science — by addressing fundamental needs and changes to them — we will continue to spend monies at an ever-increasing rate, both in construction and maintenance of software, to achieve Computer Networking.

CONTROL DATA'S NETWORK ARCHITECTURE

The Network Architecture Control Data is employing is not revolutionary, but evolutionary. It was established to form a foundation to basically address independency of functions and the status and evolution of Common Data Communications Control Procedures today. We must also recognize and constrain our evolution with the Installed Base in mind while recognizing the opportunities in new Market areas.

The current strategies point to three Operating Systems for our CYBER Product Line, with two of these evolving into one. The objective here is to reduce the cost to implement and maintain multiple Operating Systems while achieving necessary performance for the USER. To assure that the migration is accomplished in an orderly manner, we have established a common, single Network Architecture at the outset. A separate paper describes this Architecture, but its characteristics address the fundamental problems that we must address — namely — separation of Data Communications or Network Management and Control Functions, the ability to add End-toEnd Assurance Procedures, accommodate a wide range of Protocols until we are able to achieve commonality, the ability to Resource-Share, and the ability to construct Distributive-Resource Networks. To assure that our implementation, support, and maintenance costs are minimized, we are employing high-level Implementation Languages. Experience thus far indicates that it works and significantly reduces code/debug/change times.

The application of microelectronics technology has shown us that we can improve reliability, reduce cost, and improve performance. A recent delivery illustrates progress. A Computer System, including Network ADAPTERS, was delivered and accepted within an 18-day time-span.

We also participate in the ADAPTER Marketplace with Message Switchers, Concentrators, and Front-End equipments. It is clear that if we are to lever the costly element called software, we will be "emulating" with hardware which employs newer, lower-cost, and reliable microcircuits.

TRENDS

- CHANNELS
  - MORE BANDWIDTH, LOWER ERROR RATE
  - LOWER COST
  - GREATER PROPAGATION DELAY THAN TERRESTRIAL

- SWITCHES
  - CONVERSION TO ELECTRONICS
  - SLOW EVOLUTION
  - IMPROVED CONNECT TIMES
  - MORE CHANNELS

- ADAPTERS
  - SIGNIFICANT REDUCTION IN HARDWARE COSTS
    - MODEMS (NOW LOW AS $0.125/BIT)
    - MULTIPLEXERS
    - CONCENTRATORS
    - FRONT ENDS
    - PACKET SWITCHERS
    - CHANNEL ADAPTERS
  - NO SIGNIFICANT REDUCTIONS IN SOFTWARE IMPLEMENTATION AND SUPPORT
    - MESSAGE SWITCHERS
    - PACKET SWITCHERS
    - NETWORK MANAGERS

FIGURE 17.

TRENDS (CONTINUED)

- COMPUTERS
  - SOFTWARE
    - USE OF HIGH LEVEL IMPLEMENTATION LANGUAGE
    - USE OF BETTER DISCIPLINES
  - HARDWARE
    - IMPROVED RELIABILITY
    - LONGER LIFE CYCLE
    - MORE WORK SPACE

- DEVICES
  - IMPROVED FLEXIBILITY FOR PROCEDURAL CHANGES
  - INTEGRAL MODEMS (POWER SHARING)
  - IMPROVED RELIABILITY

- PROCEDURES
  - INTEGRATION TO BIT–ORIENTED PROCEDURES FOR LINK CONTROL
  - FORCED BY NEED TO USE BANDWIDTH
  - TOLERATE PROPAGATION DELAYS
  - IMPROVE DEVICE CONTROL FLEXIBILITY

FIGURE 17. (Continued)

AN APPLICATION OF THE NETWORK ARCHITECTURE

An application of the Network Architecture and its viability in a geographically-dispersed, computer-resource configuration is illustrated in Figure 18. The Network Functions needed to implement this Computer Network are shown in Figure 19.

The Network consists of multivendor "Host" COMPUTERS which must interconnect with multivendor DEVICES via a Common Communications Network.

To adapt to the multivendor COMPUTERS, ADAPTERS (called Couplers and MPCC) are employed to interface with common, local Switch ADAPTERS called Local Network Processors (LNP's). The LNP's connect to other LNP's or RNP's (Remote Network Processors) to transport data with a Common Link Control Procedure throughout the Communications Network channels. All LNP's and RNP's are managed by the NETWORK MANAGER which resides in a Host Computer dedicated to Network Management and Support in this special case.

The functions which logically vary by application are the Coupler ADAPTERS and the "Line/Terminal Interface" functions residing in the LNP's and RNP's. Note that all RNP's and LNP's are logically identical except when they are connected to a Communications CHANNEL, DEVICE, or a COMPUTER,or the RNP's and LNP's only vary in space needs (Memory) and Line/Terminal (DEVICE) Interface functions. Space needs are dictated by size of the overall Computer Network and Throughput needs.

Bit-oriented Link Control Procedures are employed between the RNP's and LNP's to effectively employ bandwidth and adjust to CHANNEL characteristic changes, and Packet Switch technology is used to better match Computer Channel to Communication Channel error characteristics. This technology is common for all applications.

Analysis and Design thus far indicate that the Architecture is sound and that we will be continually adjusting Procedures to match Communications Control Procedure evolution.

SUMMARY

Computer Networking with the ability to share Computing Resources can be achieved in an economical manner by the establishment of Common Data Communications Control Procedures. Channel Interface, Channel Control, and Link Control Procedures are not sufficient. We must complete Device Control, Message Control and End-to-End Control Procedures to achieve effective utilization.

AN APPLICATION OF THE NETWORK STRUCTURE

A LOOK AT NETWORK PRODUCTS APPLICATION

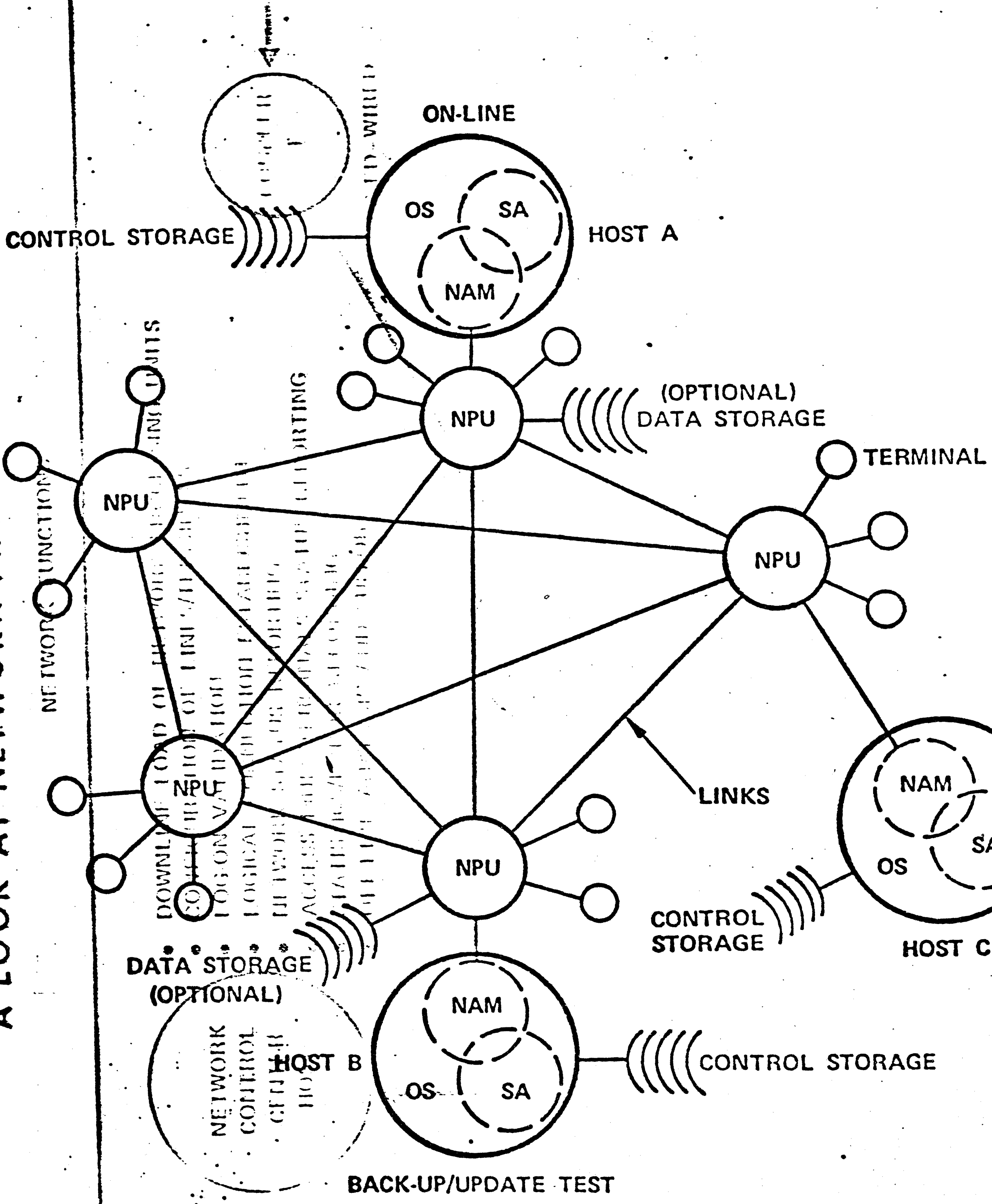
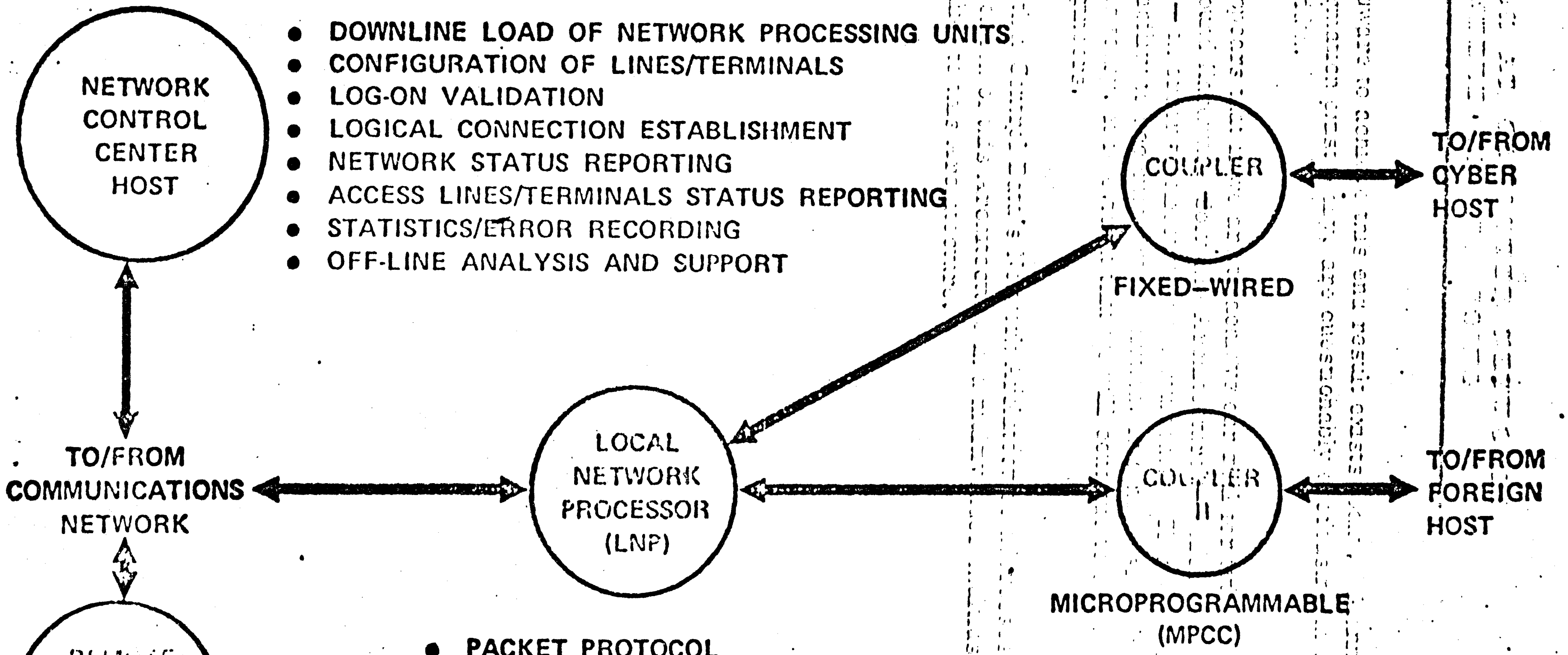


FIGURE 18.

A LOOK AT NETWORK PRODUCTS APPLICATION

NETWORK FUNCTIONS



- DOWNLINE LOAD OF NETWORK PROCESSING UNITS
- CONFIGURATION OF LINES/TERMINALS
- LOG-ON VALIDATION
- LOGICAL CONNECTION ESTABLISHMENT
- NETWORK STATUS REPORTING
- ACCESS LINES/TERMINALS STATUS REPORTING
- STATISTICS/ERROR RECORDING
- OFF-LINE ANALYSIS AND SUPPORT

- SAME AS LNP EXCEPT
- DELETE HOST COUPLER

- PACKET PROTOCOL
- BLOCK REASSEMBLY
- BLOCK PACKETIZING
- NETWORK LOGICAL ADDRESS TO PHYSICAL ADDRESS TRANSLATION
- HOST INTERFACE AT BLOCK LEVEL
- STATUS/STATISTICS REPORTING
- PACKET ROUTING
- TRUNK PROTOCOL
- LINE/TERMINAL INTERFACE

- HOST CHANNEL ELECTRICAL INTERFACE
- CHANNEL COMMAND PROCESSING
- MULTIPLEXING OF CHARACTERS FROM HOST AND ASSEMBLY TO BLOCKS
- MULTIPLEXING OF CHARACTERS TO HOST FROM BLOCKS

FIGURE 19.

The hardware to achieve this end result exists today. Channels to transport data and information exist today, are questionably decreasing in cost, and are expanding in bandwidth.

It then appears that if we are to achieve our goal for true "Computer Networking", we need a sound Hardware/Software Architecture — which can evolve with the necessary changes — in order to effectively communicate. Clear separation of functions, coupled with flexibility to accommodate evolution; is a necessity. Since the hardware technology exists today, it is evident that we will continue to see a significant future for Software Engineers.

It is felt that Control Data's Network Architecture addresses the Evolution envisioned — and application of this Architecture in a multivendor/multiuser environment appears to add viability to this Architecture.

BIT ORIENTED COMMUNICATION CONTROL PROTOCOLS

A USERS PERSPECTIVE

James W. Conard
Control Data Corporation

INTRODUCTION

The world of data communications is simmering with new activity. Trade magazines, seminars, and standards groups are humming with new acronyms: SDLC, ADCCP, HDLC, CDCCP. In the development labs of corporations, engineers and programmers are struggling with flags, bit-stuffing, commands, and responses. Among the users of data communications an internal and widespread interest is being generated. Users are asking: What is the cause of all this activity? What does it mean to me? How will it impact my requirements?

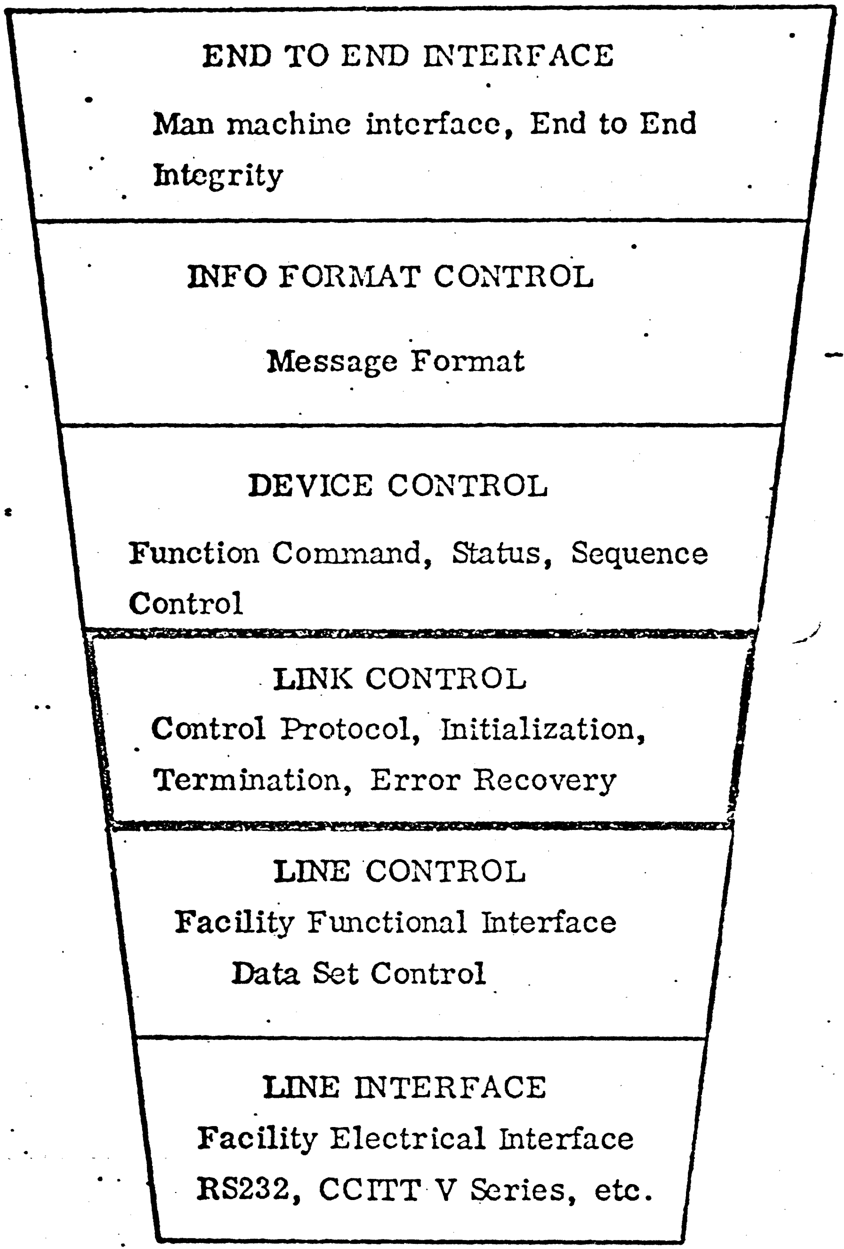
The cause of this activity is the rapid maturity of a different approach to data link control protocols. Setting aside all of the acronyms for the moment, we refer to this approach generically as bit oriented link control protocols.

We'll review this new approach beginning with an overview of bit oriented protocols. We'll continue with a discussion of the evolution of the new technique. From there we'll delve into the technical aspects and conclude with a summary of Control Data's activity and goals as they relate to the new protocol.

AN OVERVIEW OF BIT ORIENTED PROTOCOLS

A data link control protocol is a set of very specific rules under which data is exchanged between business machines via a communications circuit. The business machines may be terminals, concentrators, message switches, or computers, in any mix. A link protocol typically defines initialization of an established link, control of normal data interchange, termination of the link, and perhaps most important to the user, abnormal condition recovery techniques which serve to assure message integrity.

Strictly speaking, the term link control excludes other levels within the communications procedure hierarchy (Figure 1). One of the objectives of the new protocol was, in fact, to clearly delineate the interface between link control and higher levels such as device and message control. The characteristics of these levels, do however, impact on link level control. The prudent system designer keeps a wary eye on their requirements.



COMMUNICATION CONTROL HIERARCHY

FIGURE 1

BIT ORIENTED COMMUNICATION CONTROL PROTOCOLS A USERS PERSPECTIVE

Link control protocols have traditionally been character-oriented. They utilized, either singularly or in sequence, defined character structures from a given code set to convey supervisory information. Even though character oriented protocols represent the vast majority of protocols in use today, it has long been recognized that they suffer from many deficiencies. Among these are:

1. The necessity to distinguish between data and control characters within a code set places a burden on hardware and software implementation.
2. The assignment of characters for link control subtracts from the combinations otherwise available for information transfer.
3. The character orientation meant that they were not naturally transparent to the structure or encoding of the text.
4. Transparency could only be achieved by invoking complicated escape techniques and at the expense of incompatibility with non-transparent protocols.
5. The mixture of message control, device control, and link control forced a significant amount of processing at a low functional level and blurred the interface between these logically independent functions.
6. Error checking is usually done only on the text thus exposing supervisory sequences to undetected errors which complicate error recovery.
7. The inherent two way alternate nature of these protocols do not economically utilize full duplex facilities.
8. The rigid structure of character oriented protocols lack flexibility and expandability.

Bit oriented protocols are an outgrowth of attempts to overcome these deficiencies. The inherent characteristics of the new protocol, which when properly applied, overcome the disadvantages of character protocols include:

1. Bit orientation. They utilize positionally located control fields rather than code set combinations for link control.
2. Code independence. The use of framing flags and control fields divorces link control totally from the pattern or code structure of the information content. Thus bit oriented protocols are inherently transparent.
3. Reliability. The use of one standard format for all information and supervisory transmission permits error checking of control as well as text information.
4. Flexibility. Bit oriented protocols permit implementation in a variety of applications using a variety of communication facilities without modification of basic link control procedures.

5. **Efficiency.** The techniques applied are designed to take full advantage of full duplex facilities while retaining the ability to operate efficiently on half duplex facilities where desired.

6. **Hierarchical Independence.** Bit oriented protocols separate the functions of link control from those of device and message control.

Bit oriented protocols combine these characteristics to provide greater utilization of facilities than is possible with the older character oriented methods. Their application permits the user to more fully achieve the benefits of his data communication system.

While reviewing what the bit oriented protocols are, it is equally important to consider what they are not. The new protocols are not the total solution to the communications problem. They are only a link level control mechanism and thus are concerned solely with the transfer of data at that level. They are not a network protocol. They do not control the flow of information between users in a multi nodal network. They can, however, be applied between nodes or between a node and a user. Any necessary end-to-end controls must be imbedded in the bit oriented frame as information.

EVOLUTION OR REVOLUTION

It is natural for the user to ask: Do the new bit oriented protocols represent an evolutionary or revolutionary departure from the older approaches? The answer may be found in a brief review of the evolution of data communications protocols.

Data link control protocols are as old as data communications. Over the years these protocols have been evolving typically to fulfill the requirements of a particular application. Early systems, using Baudot Code, had no inherent link control capability. They relied totally on sequences of data characters to implement supervisory functions. The advent of other character sets led to protocols using controls derived from these sets. Each manufacturer developed protocols reflecting the needs of his product line and usually optimized for a specific implementation.

Control Data and IBM, to cite two examples, have each developed at least three protocols which achieved fairly widespread application. Control Data developed Mode II, Mode IV, and Export 0 each with different characteristics and areas of application. IBM did the same with GPD, STR, and BSC. Other manufacturers and user's groups also construced protocols to meet their unique requirements. All of these various protocols were character oriented in approach and generally incompatible with each other.

Standards organizations here and abroad, especially ANSI, ISO, and ECMA, recognized the problem and struggled to resolve the incompatibilities. For lack of standardization, the protocols developed by the larger dominant manufacturers tended to fill the vacuum by becoming, in effect, de facto standards. This has certainly been the case with IBM's BSC developed in the late 1960's.

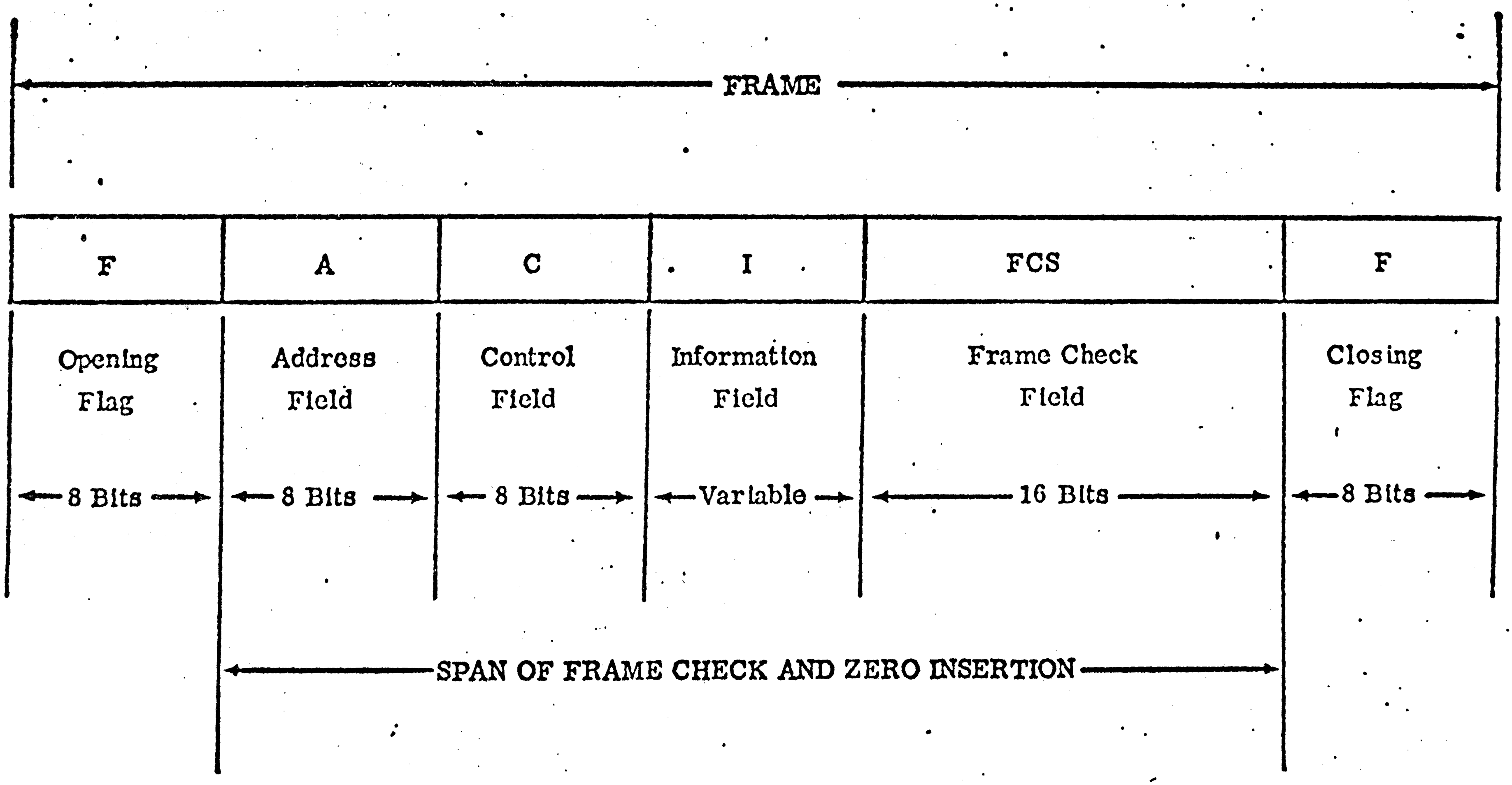


FIGURE 2. FRAME STRUCTURE

BIT ORIENTED COMMUNICATION CONTROL PROTOCOLS
A USERS PERSPECTIVE

The standards organizations finally reached agreement with the publication in 1971 of ANSI's X3.28 on the use of ASCII control characters for information interchange and ISO's R1745 Basic Mode Control Procedures. It is worthy of note, however, that even before publication of these standards both bodies were already at work on bit oriented protocols. This was the result of recognition of deficiencies in character oriented protocols that surfaced during the standardization process, and as a result of on-line experience with these protocols.

In late 1969, ANSI and ISO began formal work leading toward the development of bit oriented standards. Other groups such as IATA, ICAO, and ECMA also initiated study efforts as did the various manufacturers. These efforts reached fruition and caught the attention of users in mid 1973 with the announcement by IBM of their bit oriented protocol known as Synchronous Data Link Control (SDLC). ANSI followed in early 1974 with the first draft of Advanced Data Communication Control Procedures (ADCCP). ISO also began to formulate their High Level Data Link Control Procedures (HDLC).

This brief review demonstrates that the "new" approach to link control, the bit oriented protocol, represents no more than a natural and evolutionary milestone in the continued effort to improve data communications. It is, perhaps, revolutionary in the sense that a large degree of standardization is being achieved before widespread implementation.

TODAY AND TOMORROW

After having traced the evolution of bit oriented protocols, it is appropriate to review the present "state of the protocol" and to attempt to assess the probable future impact of this approach.

The present status of bit oriented protocols may be characterized as rapidly approaching maturity. Looking at the progress of the standards activity first we see that ANSI X3S34, which bears the responsibility for data communication protocol procedures, has completed the fourth draft of ADCCP. This group is now working on the definition of some new commands and responses which have recently been added, completing work on classes of procedures, firming up recovery procedures, and cleaning up open items and editorial changes. It is anticipated that the final draft should be ready for ballot within a year. CDC is a very active member of this task group as well as its parent body X3S3, which covers data communications.

ISO, the International Standards Organization and more specifically ISO/TC97/SC6, has chosen to divide the HDLC standard into three or more standards. The frame structure standard, DIS 3309.2 has been approved and, following some editorial changes, should be released soon. The elements of procedure standard, DP 4335 has been approved at the subcommittee level and has been sent to the next level for balloting. Approval is expected within a year. Some changes are being proposed by the US and most should be adopted. ISO has elected to standardize classes of procedure as separate documents. Five of these were formulated at a recent ISO meeting and were released for ballot at the subcommittee level. CDC also participates in this activity and was a member of the US delegation at the recent TC97/SC6 meeting in Washington.

While the standards are not binding, they will exert strong influence on the industry. They will provide guidelines that will point future software and hardware development in the same direction.

Manufacturers, meanwhile, are busy developing and announcing their own bit oriented protocols. IBM has their SDLC, Burroughs their BDLC, NCR has BOLD, and CDC has CDCCP. The best information available indicates that all of these protocols are close to complete subsets of ADCCP. Some manufacturers have also announced products including bit oriented protocol packages. Some of these are operational in limited applications.

The federal government is also in the process of preparing standards for publication as FIPS. These are also ADCCP compatible. ECMA and CCITT are expected to publish standards compatible with HDLC.

Although not yet fully mature, bit oriented protocols can be expected to have a major impact on data communications over the next five years. A primary reason for this is the impetus provided by IBM. SDLC is expected to be the only bit oriented protocol that IBM will support. This will require manufacturers of terminals and processors, as well as software suppliers, who hope to interface IBM equipment to adopt SDLC which is a subset of ADCCP.

Another impetus toward adoption is that for, perhaps the first time, a broad base of standardization exists before widespread implementation begins. This fact has been recognized by IC manufacturers who are now developing chips to handle perhaps 80% of the bit oriented protocol function.

TECHNICAL OVERVIEW

All of the bit oriented protocols being considered for implementation at this time may be characterized as being comprised of three major constituent parts. These are: the frame structure; the elements of procedure; and the classes of procedure.

Frame Structure

The frame structure provides a common structure for all supervisory and information transfers in the bit oriented protocols. The frame structure governs the structure, formatting, and significance of the various fields in the frame as well as the frame delimiting flags and frame check sequences. The following paragraphs provide a broad overview of the technical aspects of the frame structure.

A frame is a sequence of contiguous bits bounded by and including opening and closing flag sequences. A valid frame is a minimum of 48 bits in length and must conform to the following structure (Figure 2):

F, A, C, I, FCS, F

where

F = Flag Sequence
A = Address Field
C = Control Field
I = Information Field
FCS = Flag Check Sequence

Frames containing only link control sequences form a special case where no I field is present.

Flag Sequence (F)

All frames open and close with the flag sequence. This sequence has the binary configuration 01111110, that is, a zero bit followed by six one bits, followed by a zero bit.

The opening flag serves as a position reference for the address, and control fields and initiates transmission error checking. The closing flag serves as a position reference for the flag check sequence.

Transmitters must send only complete eight-bit flags. All receivers attached to the data link must search continuously, on a bit-by-bit basis, for the flag sequence. Thus, the flag sequence provides frame synchronization.

An F may be followed by a frame, another F, or an idle line. An F which closes a frame may also be used as the opening F on a following frame. Any number of F's may be transmitted between frames.

Since the F sequence brackets and synchronizes the frame, it must be prevented from occurring in any field of the frame. This is accomplished by the zero insertion technique described below.

Each transmitter must insert a zero bit following five contiguous one bits anywhere between the opening and closing flag sequences. The insertion of the zero bit thus applies to the address, control, information, and FCS fields and effectively prevents the fortuitous transmission of the F sequence 01111110.

Each receiver after detecting the opening flag (start of frame) continuously monitors the received bit stream and removes any zero bit which follows a succession of five contiguous one bits. Note that zero insertion at the transmitter follows the computation of FCS and that zero deletion at the receiver precedes the FCS check process.

Address Field (A)

The address field (A) immediately follows the opening flag of a frame and precedes the control field. This field always contains the address of the secondary station. The primary station is never identified. The address field is N octets in length where $N \geq 1$. The contents of the field may be a single, group, or global address.

Two addressing modes are defined for the secondary station link address field. These are the basic and extended modes described below. For a specific link the maximum number of octets must be explicitly defined.

In the basic mode, the secondary link address field contains one address, which may be a single, group, or global secondary address. In this mode, address extension is not permitted. All 256 combinations are available for addresses. This basic mode field consists of one eight bit octet with the format illustrated in Figure 3.

In the extended mode, the secondary link address field is a sequence of octets which comprise a single secondary address. The least significant bit is used as an extension indicator. When this bit is zero, the following octet is an extension of the address field. The address field is terminated by an octet having a one in bit position one (least significant bit). Thus the address field is recursively extendable. The format of the extended address field is also illustrated in Figure 3.

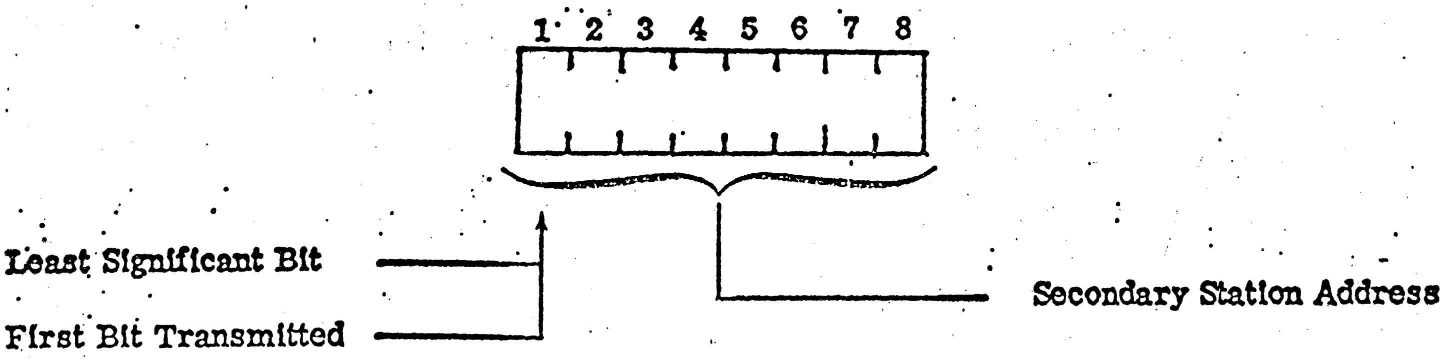
Each secondary station on a data link must be capable of recognizing a group or global address which is contained in one unextended octet even when extended mode is normally used.

Two or more secondaries may be required to recognize the same group or global address. Each secondary, however, responds with its individual address.

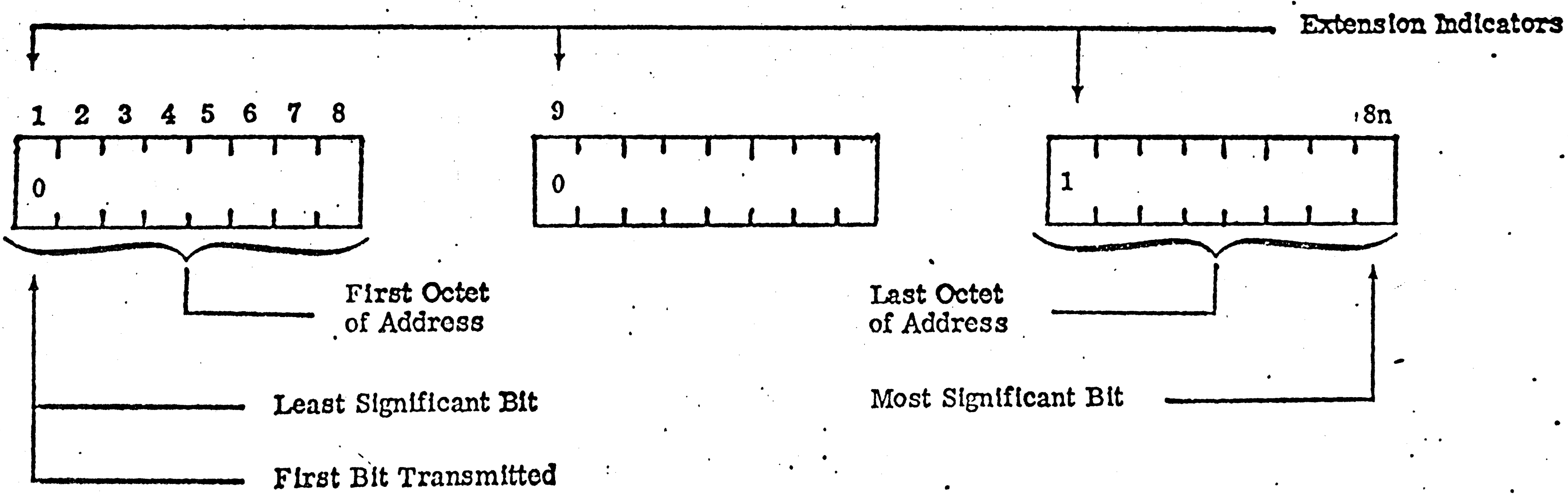
Control Field (C)

The control field (C) is located immediately following the address field and preceding the information field in the frame structure. The control field is used to convey commands, responses, and sequence numbers necessary to control the data link.

There are two modes defined for the control field. These are the basic and extended modes described in the following paragraphs. For a given link the mode must be specifically identified.



a) BASIC MODE ADDRESS FIELD



b) EXTENDED MODE ADDRESS FIELD

FIGURE 3. ADDRESS FIELD FORMAT

The basic control field consists of a single 8 bit octet. This field is structured into one of three formats. These are the information transfer format used by primary and secondary stations to transfer information, the supervisory format used to convey link supervisory data, and the unnumbered format used to provide additional primary and secondary link control functions.

In addition, each format includes a format identifier and a poll/final bit. The poll/final bit serves as the send/receive control. A poll (P) bit is sent only by a primary and is used to authorize secondary transmission. The final (F) bit is used only by a secondary in response to a P bit. Only one P bit is outstanding, i.e., unanswered by an F bit, on a data link.

Figure 4 illustrates the basic mode control field.

The basic mode control field provides for a modulus 8 sequence count. On long propagation delay links, e.g., satellite links, it may be necessary to extend the sequence number modulus. The extended mode control field provides this capability.

The control field is extended by the addition of a second contiguous octet immediately following the basic field. This extension increases the modulus count to 128. The three formats for an extended mode control field are also illustrated in Figure 4.

Information Field (I)

A frame exists as a vehicle for transporting the data contained in the information field (I). The data link control is completely transparent to the contents of the I field. The I field may, therefore, consist of any number of bits, in any code, related to character structure or not. The I field is unrestricted as to length but it should be recognized that typical length is contingent on system requirements and limitations beyond the link level. Factors limiting I field length may include channel error characteristics, station buffer sizes, and the logical properties of the data.

The occurrence of a flag or abort sequence within the I field is prevented by the zero insertion technique described previously.

I fields are normally included in every frame having a C field with an information transfer format. These information transfer frames are the only ones which are sequence numbered. An information field with a length of zero is specifically permitted.

Provisions are also made for an I field in an unnumbered C field format. Such frames are not protected by sequence checking.

Formats	1	2	3	4	5	6	7	8
Information Transfer	0	N (S)			PF	N (R)		
Supervisory	1	0	S	S	PF	N (R)		
Unnumbered	1	1	M	M	PF	M	M	M

Control Field Bits

First Bit Transmitted

where:

N (S) = Send Sequence Count

N (R) = Receive Sequence Count

S = Supervisory Function Bits

M = Modifier Bits

PF = Poll Final Bit

a) BASIC MODE CONTROL FIELD

	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Information Transfer	0	N (S)							PF	N (R)						
Supervisory	1	0	S	S	X	X	X	X	PF	N (R)						
Unnumbered	1	1	M	M	X	X	X	X	PF	M	M	M	X	X	X	X

First Bit Transmitted

where X bits are reserved and set to 0.

b) EXTENDED MODE CONTROL FIELD

FIGURE 4. CONTROL FIELD FORMAT

Frame Check Sequence (FCS)

Each frame includes a 16 bit frame check sequence (FCS) immediately following the I field (or the C field if there is no I field) and preceding the closing flag. The FCS field serves to detect errors induced by the transmission link and validate transmission accuracy. The 16 bits result from a mathematical computation on the digital value of all binary bits (excluding inserted zeros) in the frame including the address, control and information fields.

The process is known as cyclic redundancy checking using a generator polynomial of $X^{16} + X^{12} + X^5 + 1$. The transmitter's 16 bit remainder value is initialized to all ones before a frame is transmitted. The binary value of the transmission is premultiplied by X^{16} and then divided by the generator polynomial. Integer quotient values are ignored and the transmitter sends the complement of the resulting remainder value, high order bit first, as the FCS field.

The receiver will discard a frame in error and will not advance the receive sequence count thus causing a retransmission of the errored block.

Elements of Procedure

The elements of procedure comprise the building blocks of a bit oriented protocol. All elements employ the common frame structure discussed previously. Elements of procedure include operational modes, commands, and responses. Using these common elements, various classes of procedure which meet the requirements of various application situations can be constructed. The paragraphs which follow summarize the various elements and their characteristics.

Operational Modes

Bit oriented protocols define two primary operational modes. These are the Normal Response Mode (NRM) and the Asynchronous Response Mode (ARM).

NRM is an operational mode in which a Secondary station may initiate transmission only as the result of receiving explicit permission to do so from the Primary station. Explicit permission is defined as transmission by the Primary of a command frame with the Poll bit set to 1. After receiving permission, the Secondary initiates a response transmission. The response transmission may consist of one or more contiguous frames. The last frame of the transmission will be explicitly indicated by the Secondary by means of a Final bit set to 1. Following transmission of the last frame, the Secondary will stop transmitting until explicit permission is again received from the Primary.

ARM is an operational mode in which a Secondary may initiate transmission without receiving explicit permission from the Primary. Such an asynchronous transmission may contain single or multiple frames and is used for information field transfer and/or status changes in the Secondary. Examples of status changes are the number of the next expected frame, change from a ready to a busy condition or vice versa, or establishment of an exception condition.

In ARM, a Secondary will transmit a frame with a Final bit set to 1 only in response to a received command frame with the Poll bit set to 1. Additional response frames may be transmitted following the frame which has the Final bit set to 1.

Transmission Formats

Three control field formats are used perform information transfer, basic supervisory control functions, and special or infrequent control functions.

The Information (I) format is used to perform an information transfer. It is the only format which may contain an information field. The functions of sequence counts and poll/final bit are independent, that is, each frame has a transmit send sequence count, the receive sequence count may or may not acknowledge additional frames at the receiving station, and the P/F bit may or may not be set to 1.

The Supervisory (S) format is used to perform link supervisory control functions such as to acknowledge information frames, to request retransmission of information frames, or to indicate temporary interruption of receive capability.

The Unnumbered (U) format is used to provide additional Primary and Secondary link control functions. This format contains no sequence numbers. As a result, 5 modifier bit positions are available which allow definition of up to 32 additional supervisory functions.

Transmission Parameters

The parameters associated with the three transmission formats are described in the following paragraphs.

Each information frame is sequentially numbered and may have the value 0 through modulus minus 1 (where modulus is the modulus of the sequence numbers). Modulus equals 8 for the unextended control field, and the sequence numbers cycle through the entire range.

The maximum number of sequentially numbered information format frames that the Primary or Secondary may have outstanding (i.e., unacknowledged) at any given time may never exceed one less than the MODULUS of the sequence numbers. This restriction is to prevent any ambiguity in the association of transmission frames with sequence numbers during normal operation and/or error recovery action.

Each station maintains a separate (independent) Send Sequence Number N (S) and a Receive Sequence Number N (R) on the information frames it sends and receives. Each Secondary station then maintains an N (S) count on the information format frames it transmits to the Primary, and an N (R) count on the information format frames it has correctly received from the Primary. In the same manner, the Primary maintains separate N (S) and N (R) counts for information format frames sent to and received from each Secondary on the link.

Poll/Final (P/F) Bit

The Poll/Final (P/F) bit serves a function in both command and response frames. In command frames, it is referred to as the P bit. In response frames, it is referred to as the F bit. In both cases, the bit is set to 1.

The P bit is used by a Primary to solicit a response or sequence of responses from Secondaries.

In NRM, the P bit is set to 1 when the Primary desires to solicit information frames from a Secondary or solicit supervisory or unnumbered responses from a Secondary. In NRM, the Secondary cannot transmit until a command frame with a P bit is received. The Primary can solicit information frames by sending an information frame with a P bit or by sending certain supervisory frames with a P bit. The Primary can also restrict the Secondary from transmitting information frames by sending a "receive not ready" supervisory frame with a P bit.

In ARM, the P bit is not used to solicit information frames since these can be transmitted by the Secondary on an asynchronous basis. The P bit may, however, be used to solicit supervisory or unnumbered responses. For example, if the Primary wants to get positive acknowledgment that a particular command was received, it may set the P bit in the command. This will force a response from the Secondary.

The F bit is used only by a Secondary and only to respond to a P bit received from a Primary.

In NRM, the Secondary is required to set the F bit to 1 in the last frame of its response which may consist of one or more frames. Following the transmission of a frame with the F bit set to 1, the Secondary must halt transmission until a command frame with a P bit set to 1 is received.

In ARM, the Secondary is required to transmit a response frame with the F bit set to 1 in response to a P bit but is not required to halt transmission. The F bit shall be sent at the earliest opportunity as a function of link configuration, i.e., TWA or TWS. Since additional frames may be transmitted by a Secondary in ARM following an F bit response, the F bit is not to be interpreted by the Primary as the end of transmission. It simply serves to finalize the response to the Primary command frame with the P bit set.

Since P and F bits are exchanged on a one for one basis and only one P bit can be outstanding at a time, the N (R) count of a frame containing a P or F bit set to 1 can be used to detect I frame sequence errors. This capability is referred to as checkpointing and can be used not only to detect frame sequence errors but to indicate the frame sequence number to begin retransmission.

Commands and Responses

The following paragraphs briefly describe each of the set of commands and responses used in each of the three transmission formats. Figure 5 summarizes these commands and responses.

The function of the Information Transfer command and response is to transfer sequentially numbered frames containing an information field across a data link. The I command and response control field is illustrated in Figure 6.

Bit 1 of the I control field is always zero and identifies this frame as an I frame. Bit 5 is the Poll/Final bit described previously.

The Information format control field contains two sequence numbers. Bits 2, 3, and 4 comprise N (S), the send sequence count which indicates the sequence number associated with this information frame. Bits 6, 7, and 8 comprise N (R), the receive sequence count which indicates the sequence number of the next expected information format frame to be received. The N (R) implicitly acknowledges correct receipt of information frames numbered up to N (R) -1.

Supervisory format commands and responses are used to perform basic link supervisory control functions such as acknowledgment, polling, and error recovery. Frames with the supervisory format do not contain an information field and therefore do no increment the sequence counts at either the transmitter or the receiver. The Supervisory command and response control fields are illustrated in Figure 6.

Bits 1 and 2 of the S control field identify the frame as an S Frame. Bit 5 is the Poll/Final bit.

Bits 6, 7, and 8 comprise the N (R), receive sequence count, which indicates the sequence number of the next expected information format frame to be received. It also implicitly acknowledges correct receipt of information frames numbered up to and including N (R) -1.

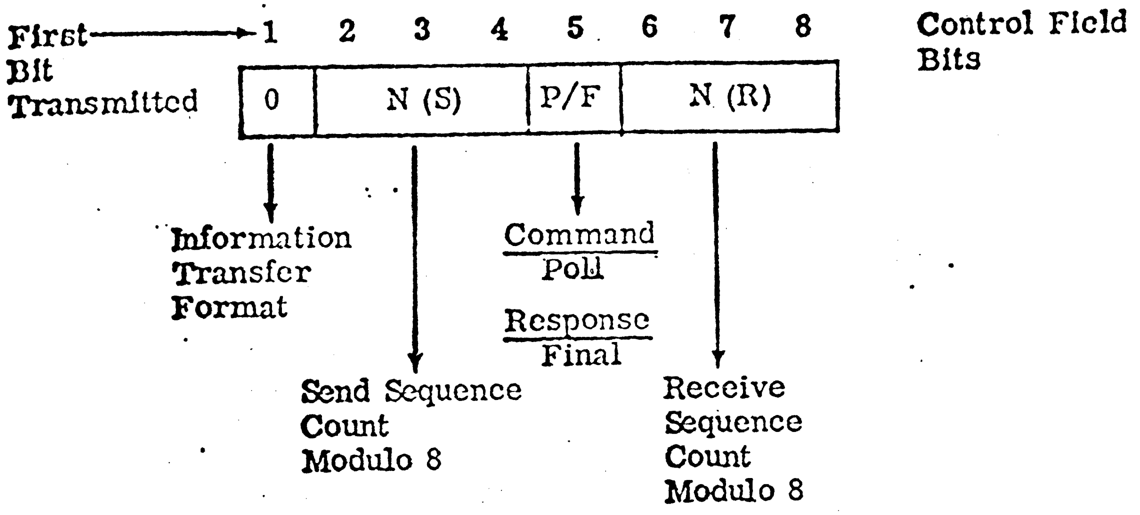
Bits 3 and 4 of the S control field define the supervisory function and are encoded as follows for both command and response frames:

Bit	3	4	Command/Response	
	0	0	RR	– Receive Ready
	0	1	REJ	– Reject
	1	0	RNR	– Receive Not Ready
	1	1	SREJ	– Selective Reject

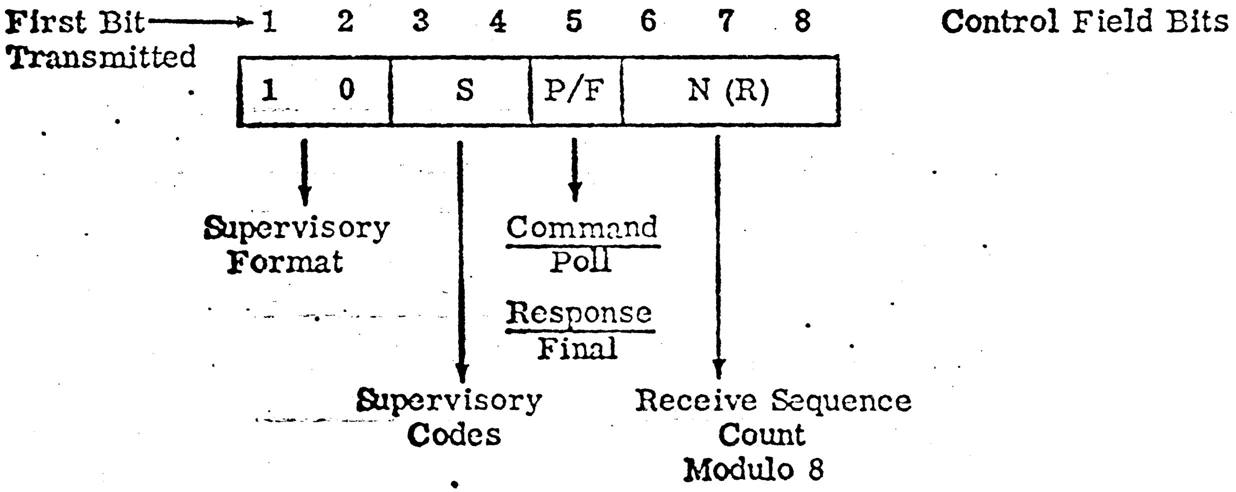
FORMAT	COMMANDS	RESPONSES
Information	I - Information	I - Information
Supervisory	RR - Receive Ready	RR - Receive Ready
	RNR - Receive Not Ready	RNR - Receive Not Ready
	REJ - Reject	REJ - Reject
	SREJ - Selective Reject	SREJ - Selective Reject
Unnumbered	SNRM - Set Normal Response Mode	
	SARM - Set Asynchronous Response Mode	
	DISC - Disconnect	
	RSPR - Response Reject	CMDR - Command Reject
	SNRME - Set Normal Response Mode Extended	
	SARME - Set Asynchronous Response Mode Extended	
	UI - Unnumbered Information	UI - Unnumbered Information
	SIM - Set Initialization Mode	RIM - Request Initialization Mode
	UP - Unnumbered Poll (Optional Res Poll)	
		UA - Unnumbered Acknowledge
		DM - Disconnect Mode (Request On Line)

FIGURE 6. RAND'S CONTROL FIEL

FIGURE 5 COMMAND/RESPONSE SUMMARY



I FORMAT CONTROL FIELD



S FORMAT CONTROL FIELD

FIGURE 6. I AND S CONTROL FIELD FORMATS

The following paragraphs delineate each of these commands and responses.

The Receive Ready (RR) supervisory frame is used by the Primary or Secondary to indicate that it is ready to receive an information frame and to acknowledge previously received information frames numbered up to and including N (R) -1. A Primary may use the RR command with the Poll bit set to 1 to solicit responses from, i.e., "poll", Secondary stations.

The Reject (REJ) supervisory frame is used by the Primary or Secondary to request retransmission of information format frames starting with the frame numbered N (R). Information format frames numbered N (R) -1 and below are acknowledged. Additional I frames pending initial transmission may be transmitted following the retransmitted I frame(s).

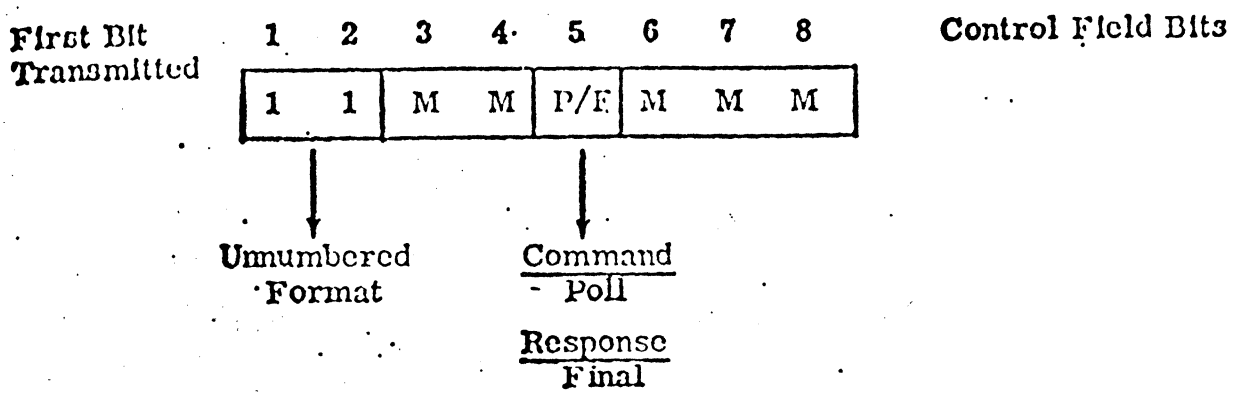
The Receive Not Ready (RNR) supervisory frame is used by the Primary or Secondary to indicate temporary inability to accept additional incoming information format frames. Information format frames numbered up to and including N (R) -1 are acknowledged; information frame N (R) and any subsequent information format frames received, if any, are not acknowledged. A station receiving an RNR frame when in the process of transmitting (i.e., a FDX station) is to stop transmitting at the earliest possible time by completing or aborting the frame in process.

The Selective Reject (SREJ) supervisory frame is used by the Primary or Secondary to request retransmission of the single information numbered N (R). Information format frames numbered through N (R) -1 and below are acknowledged. Once a SREJ has been transmitted the only I frames accepted are those which are numbered contiguously and in sequence following the I frame requested and the specific retransmitted I frame indicated by the N (R) in the SREJ command/response.

The Unnumbered (U) format commands and responses are used by the Primary and Secondary to extend the number of link supervisory functions. Frames transmitted with the unnumbered format do not increment the Send Sequence counts N (S) at either the transmitting or receiving station. Five modifier bits are defined which allow up to 32 additional supervisory functions. Of these 10 are defined. The remaining combinations are reserved for future assignment. The Unnumbered command and response control field is illustrated in Figure 7.

Bits 1 and 2 of the U format control field identify the frame as a U frame. Bit 5 is the Poll/Final bit. Bits 3, 4, 6, 7 and 8 are the modifier bits and are encoded as illustrated in Figure 7. Each of these commands and responses is described in the following paragraphs.

An Unnumbered Poll command is used to solicit transmission from the addressed Secondary station. An I field is not permitted in an UP frame.



BITS					DEFINITION	USED AS	
3	4	6	7	8		Command	Respons
0	0	0	0	0	UI – Unnumbered Information Frame	X	X
0	0	0	0	1	SNRM – Set Normal Response Mode	X	
0	0	0	1	0	DISC – Disconnect	X	
0	0	1	0	0	UP – Unnumbered Poll	X	
0	0	1	1	0	UA – Unnumbered Acknowledge		X
1	0	0	0	0	SIM – Set Initialization Mode	X	
					RIM – Request Initialization Mode		X
1	0	0	0	1	RSPR – Response Reject	X	
					CMDR – Command Reject		X
1	1	0	0	0	SARM – Set Asynchronous Response Mode	X	
					DM – Disconnect Mode		X
1	1	0	1	0	SARME – Set ARM Extended	X	
1	1	0	1	1	SNRME – Set NRM Extended	X	
All Others					Reserved For Future Assignment		

The Unnumbered Acknowledge response is used by a Secondary to acknowledge receipt and acceptance of an unnumbered command. The UA response is transmitted in the normal or extended control field format as directed by the received unnumbered command. No information (I) field is permitted with the UA response.

The SIM command is used to initiate system specified link level initialization procedures at the Secondary station. The expected response is UA. Both Primary and Secondary N (R) and N (S) counts are reset to zero.

An RIM is transmitted by a Secondary to notify the Primary of the need for a SIM Command. The receipt of commands except a SIM will cause the Secondary to repeat the RIM.

The RSPR command is used by the Primary station to report that an exception condition resulted from the receipt of an error free frame from the Secondary station. A status field is returned with this command to provide the reason for issuance of the command.

The CMDR is used by a Secondary to report that an exception condition resulted from the receipt of an error free frame from the Primary. A status field is returned with a CMDR to provide the reason for issuance of the CMDR.

The SARM command is used to place the addressed Secondary station in an Asynchronous Response Mode (ARM) where all control fields are one octet in length. No information field is permitted with the SARM command. The Secondary confirms acceptance of SARM by the transmission of an Unnumbered Acknowledge (UA). Upon acceptance of this command, the Secondary station send and receive sequence counts are set to zero.

DM is transmitted by a Secondary to indicate that it is disconnected.

The SNRME command is used to place the addressed Secondary station in the Normal Response Mode Extended (NRME) where all control fields will be two octets in length. The Secondary station confirms acceptance of SNRME by transmission of a UA response. Upon acceptance of this command the Secondary send and receive sequence counts are set to zero.

The SARME command is used to place the addressed Secondary station in the Asynchronous Response Mode Extended (ARME).

Classes of Procedure

Procedural differences among applications, based on overall system considerations such as network configuration, recovery procedures, terminal sophistication, etc., will be accommodated by defining various classes of procedure. These classes combine the modes of operation (ARM and NRM), a subset of commands and responses, and exception recovery procedures. Each class forms an implementation subset of the procedures. A class is thus characterized as the ability at the primary to receive and action all responses in the prescribed subset and the ability at the secondary to receive and action all commands in the prescribed subset.

All classes of procedure use the standard frame structure. All procedures assume that the links include primary and secondary link controllers. The primary link controller is responsible for control of the link by determining, within the constraints of this standard, which commands to send. Primary link controllers transmit only commands, in frames (with or without data). Secondary link controllers receive the command frames and transmit responses in frames (with or without data).

Since classes of procedure are now in their formative stage, the standardization picture is somewhat cloudy. ANSI has currently defined six classes covering normal mode, asynchronous mode, and primary to primary modes. ISO has very recently documented five classes covering basically the same applications. It is expected that most or all of these classes will ultimately be adopted although probably not in their present form. Discussion is now under way on methods of codifying these and other classes which will inevitably be constructed.

Implementation

The subject of compatibility between the various bit oriented protocols was mentioned earlier. Since this subject is especially important to the user, the chart in Figure 8 has been prepared. This illustrates the complete set of commands and responses now defined and, for each protocol, lists the ones being implemented. The information presented is, of course, subject to change but represents the best data available to the author. This chart indicates a high degree of basic compatibility.

Given this basic compatibility, it remains for the user to carefully determine his requirements in terms of a "class of procedure" to be used. This will define the operational modes and elements of procedure to be used. Following this it will be necessary to generate a system specification for the specific application. This document will identify and quantify many variables necessary to achieve successful on-line operation. It is here that the impact of lower and higher levels of the communications hierarchy will be reviewed and specified.

ADCCP		HDLC		SDLC		CDCCP		BOLD		BDLC		SNAP	
CMD	RES	CMD	RES	CMD	RES	CMD	RES	CMD	RES	CMD	RES	CMD	RES
I	I	I	I	I	I	I	I	I	I	I	I	I	I
RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR	RR
REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ	REJ
RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR	RNR
SREJ	SREJ	SREJ	SREJ			SREJ	SREJ	SREJ	SREJ	SREJ	SREJ	SREJ	SREJ
UI	UI	*	*	NSI	NSI	UI	UI	NSI	NSI				
SNRM		SNRM		SNRM		SNRM		SNRM		SNRM			
DISC		DISC		DISC		DISC		DIS		DISC		DISC	RDIS
UP		*		ORP		UP		ORP					
	UA		UA		NSA		UA		NSA		UA		UA
USR 0	USR 0												
USR 1	USR 1												
USR 2	USR 2												
USR 3	USR 3												
SIM	RIM	*	*	SIM	RIM	SIM	RIM	SIM	RIM				
RSPR	CMDR	*	CMDR		CMDR	RSPR	CMDR	RSPR	CMDR	RSPR	CMDR		CMDR
SARM	DM	SARM	*		ROL	SARM	DM	SARM	ROL	SARM		CONN	RCON
SARME		SARME				SARME		SARME		SARME			
SNRME		SNRME				SNRME		SNRME		SNRME			
XID	XID	*	*			XID	XID						

* PROPOSED

FIGURE 8. COMPARISON OF COMMAND/RESPONSE IMPLEMENTATION

BIT ORIENTED COMMUNICATION CONTROL PROTOCOLS A USERS PERSPECTIVE

Bit oriented protocols have a wide variety of potential application. They are suitable for two-way alternate and two-way simultaneous operation using a variety of data link configurations including full and half duplex, point-to-point, multipoint, switched, and non-switched. The three facility configurations expected to be most common in bit oriented applications are illustrated in Figure 9.

A point-to-point facility is one which interconnects two and only two stations. Point-to-point facilities may be either non-switched, sometimes referred to as private line or dedicated, or they may be switched. The difference between switched and non-switched is one of facility acquisition. In the switched case the facility must be acquired prior to the transfer of data and released at the end of the transfer. Non-switched facilities are dedicated and usable on demand.

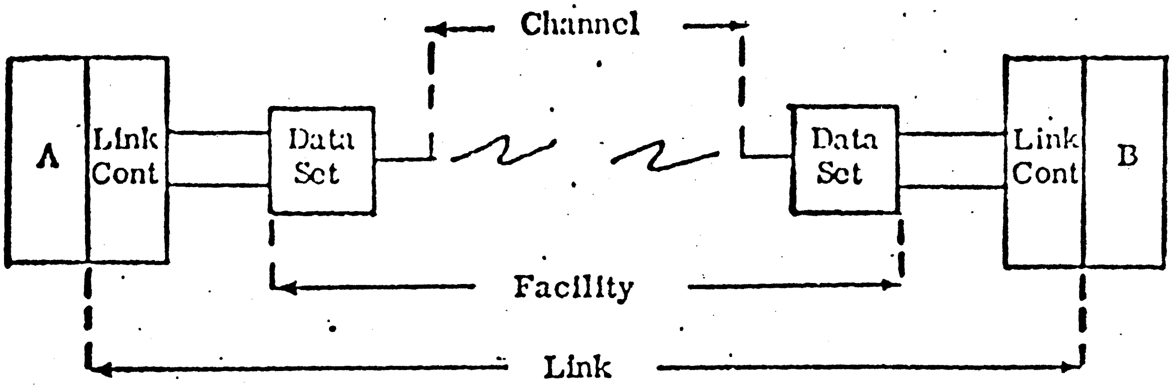
A multipoint arrangement, expected to be very common for these applications, is the broadcast polling arrangement which consists of a single master and two or more remote stations. Transmissions from the master are received by all remotes. Transmissions from the remotes are received only by the master. This multipoint arrangement requires 4-wire channels.

Many special and hybrid combinations of interconnect arrangements are possible. The most likely to be encountered in these applications is the loop arrangement. The loop configuration consists of two or more point-to-point facilities arranged such that the loop starts and ends at the same location. The point-to-point facilities are normally 2-wire channels and operate in simplex mode: A transmits to B, B transmits to C, and so on around the loop. Transmission in the reverse direction is not possible. Each station on the loop operates as a repeater. Loop facilities may be encountered which are completely user-owned, especially when located within the confines of a building. Others may use common carrier facilities when geographically dispersed.

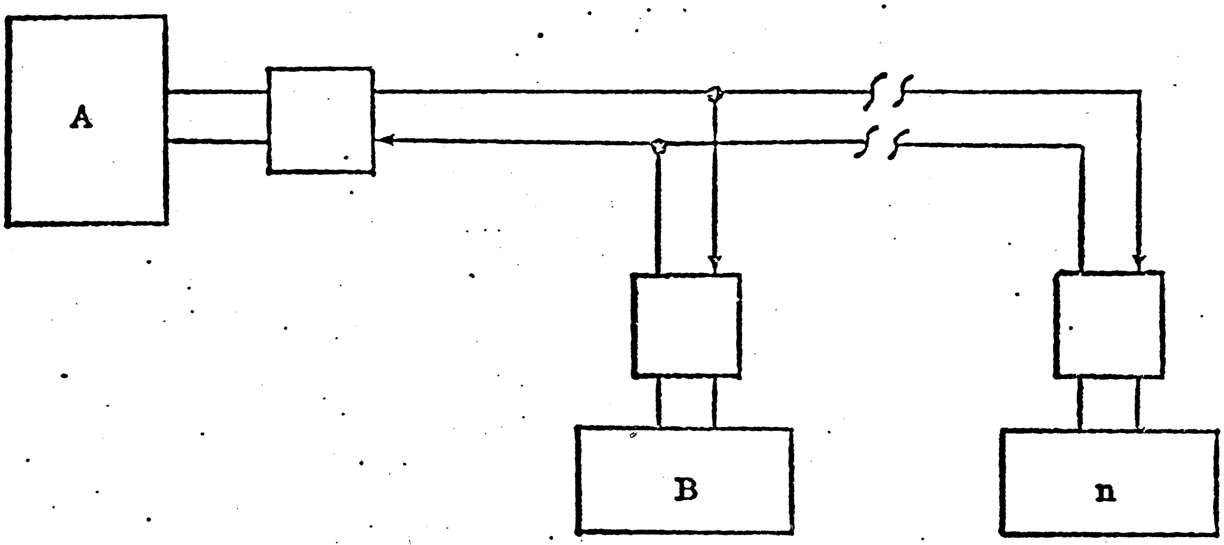
In terms of thruput performance, the user can expect significant improvement over the character oriented protocols. Serious quantitative studies of this aspect of the new approach are just beginning to surface. Preliminary results of studies here and abroad indicate high thruput efficiency and excellent response time performance.

CONTROL DATA'S CDCCP

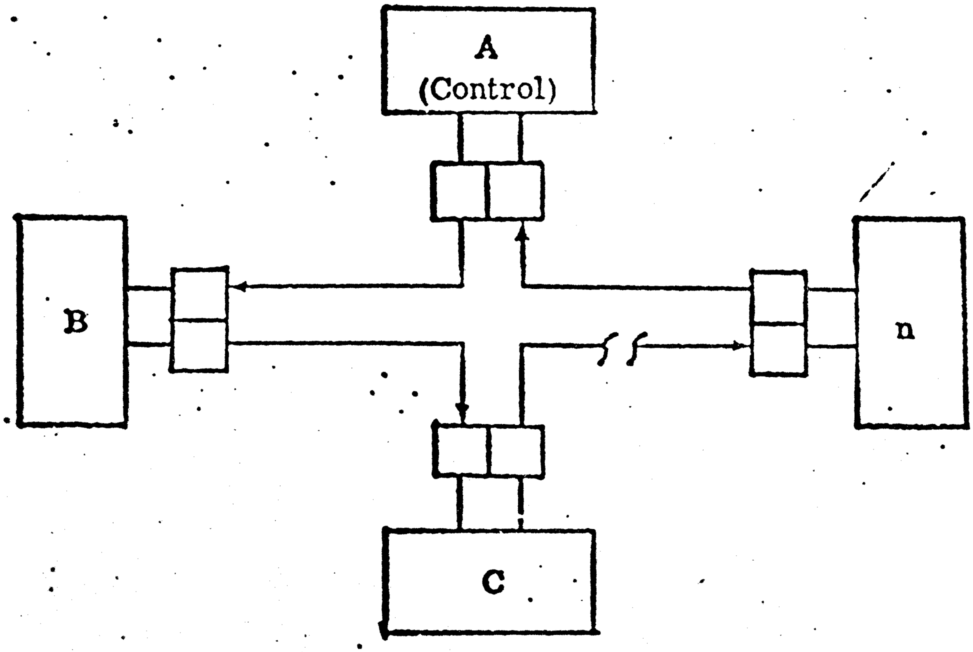
Control Data Corporation formally initiated an effort to define a Corporate Standard bit-oriented link control protocol in mid-1974. The objective of this effort was to generate a standard protocol which would facilitate the exchange of information in a variety of applications and be capable of accommodating simple to complex, low to high speed synchronous sources and sinks. The minimum requirements were that the new protocol provide for two way alternate to two way simultaneous operation, permit multi-drop configuration, be suitable for satellite transmission, provide for non-symetric and symetric operation, and include effective levels of error detection. It was also required that the protocol be modular in definition and implementation to permit wide application and to permit revision with minimum impact on implementation.



a) POINT-TO-POINT



b) BROADCAST POLL MULTIPOINT



c) LOOP

To meet these objectives, a task force was established with representatives from various divisions of the Corporation with an interest in communications. The protocol to be standardized was called Control Data Communications Control Procedure (CDCCP).

The efforts of the task force resulted in a draft of a proposed standard for CDCCP. This draft is now being reviewed by the various concerned divisions and should become a standard in early 1976.

CDCCP spans the entire set of bit oriented protocols now in the process of standardization and implementation. These include IBM's SDLC, ANSI's ADCCP, and ISO's HDLC. CDCCP is, therefore, geared to satisfy any of these requirements by use of a subset of the CDCCP protocol. The CDCCP draft is already serving to provide design guidelines to various developing divisions.

Control Data is also following closely, thru its representation on ANSI and thru liaison with other groups, developments leading to standardization of device control and message formats. These functions which would be contained within the I field of CDCCP are, of course, of major interest to CDC and its users. Other areas being pursued include the emerging application possibilities of packet switching and public and private data networks.

Control Data Corporation, as part of its total services concept, is dedicated to provide cost effective, and efficient solutions for the user's data communications problems. The development of bit oriented protocols is but one example of this committment.