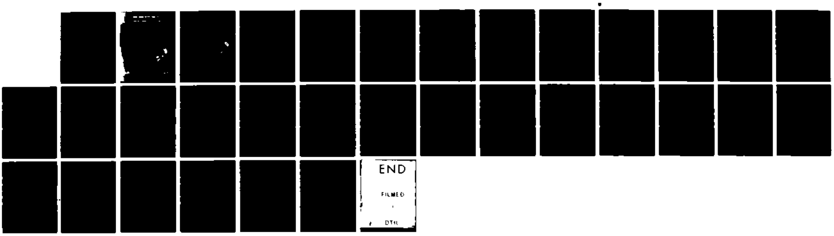
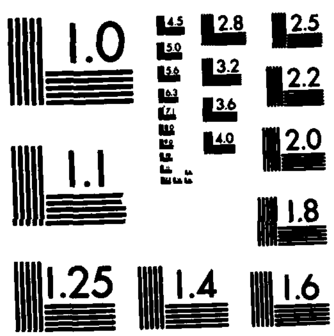
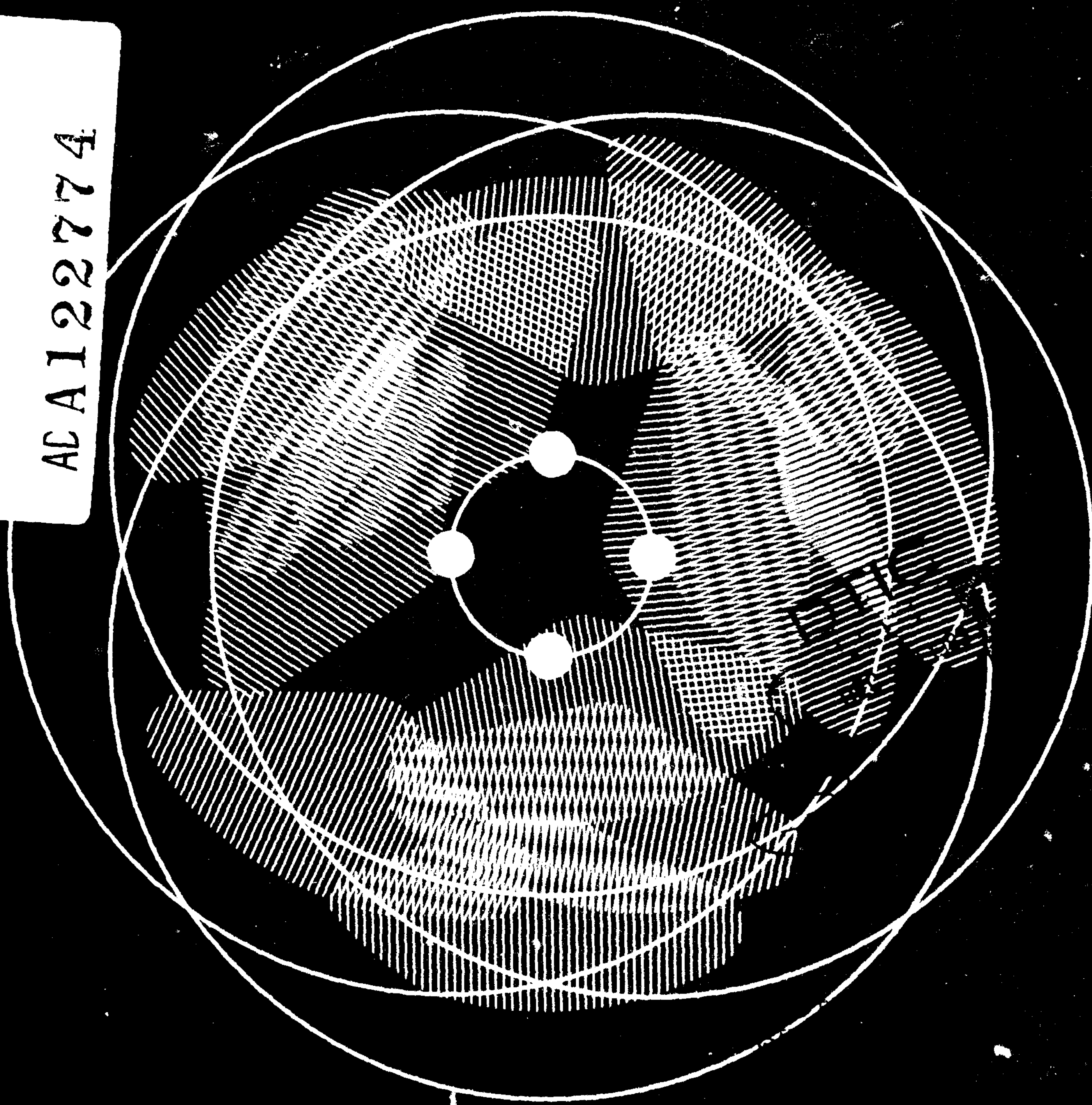
AD-A122 774 DIGITAL TERMINALS FOR PACKET BROADCASTING(U) HAWAII UNIV HONOLULU S C FRALICK ET AL. FEB 75 B75-8 NAS2-8590 1/1

UNCLASSIFIED F/G 17/2.1 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



THE ALOHA SYSTEM

UNIVERSITY OF HAWAII
HONOLULU, HAWAII 96822

(P)

DIGITAL TERMINALS FOR PACKET BROADCASTING*

by

Stanley C. Fralick
David H. Brandin
Information Science and Engineering Division
Stanford Research Institute

Franklin F. Kuo
Christopher Harrison
The ALOHA System
University of Hawaii

*This work was supported by the Advanced Research Projects Agency on Contract No. DAHC 15-73-C-0187 and Contract No. NAS2-8590.

To be presented at NCC 75, Anaheim, California

DIGITAL TERMINALS FOR PACKET BROADCASTING

Stanley C. Fralick and David H. Brandin
Stanford Research Institute

Franklin F. Kuo and Christopher Harrison
University of Hawaii

ABSTRACT

Today's technology makes it possible to build small, personal digital radio terminals with low-power consumption. Studies have shown that such lightweight terminals can be efficiently supported by packet switched radio networks using random access modes and microprocessor controlled relays. Incorporating microprocessors into the personal terminals offers an opportunity to support wider ranges of user requirements and correspondingly reduce communication loads. The capability of new liquid crystal matrix displays, greater integration of CPU memory and other circuits, and thin film RF assemblies, reinforce the possibilities of fabricating these personal units. This paper discusses communication protocol and the state of the art of microprocessor technology in the design and development of compact digital terminals for distributed packet radio networks.

CONTENTS

ACKNOWLEDGMENT

The authors wish to thank the members of the Advanced Research Project Agency research team designated as the Packet Radio Working Group who have contributed ideas, concepts and specific design details to the work discussed in this paper.

I INTRODUCTION

Roberts illustrated the potential use of packet switching technology by postulating a personal computer terminal using radio broadcasting to connect the user to a computer.[1]* The proposed terminal had a unique five-finger keyboard and plasma-discharge display. The keyboard would generate and send characters, one at a time, to the computer using 64 bit packets per character. The computer could convert these to a 35-bit (5 X 7) pattern and retransmit a 144-bit packet to the terminal to control a 5 X 7 dot matrix character. Thus, the terminal needed no character generation logic and only a minimum of digital control logic to interface keyboard and display to a radio modem. This was a reasonable concept insofar as the terminal was intended to operate within a short distance of the computer to accommodate low-power radios, and so long as only a few terminals were in use.

Roberts assumed a random access packet broadcasting transmission mode formerly developed by Abramson[2] and now known as the pure ALOHA technique. Under a pure ALOHA mode of operation, packets are sent by the terminals to the central station computer(s) in an unsynchronized manner. In this scheme the lack of positive acknowledgments (POSACK) controls retransmissions, as necessary. Using the pure ALOHA technique with a 10-character per second terminal and assuming 64-bits per character (a peak data rate of 640 bits per second), it can be shown that a 100 kilobits per second channel will simultaneously support only 26 terminals.[2] To accommodate more terminals, higher bit-rate channels are needed, along with a more efficient packet structure. For example, with Robert's proposed packet structure, modified by sending 10 characters instead of one character per packet, the same channel will support twice as many terminals.

Higher bit-rates require more transmitter power for the same range. Greater efficiency requires more memory and logic in the terminal. It has been found that the size, weight, and power consumption of the radio transmitter will dominate the terminal at high bit-rates unless the terminal range is small. To obtain larger coverage areas, a network of radio repeaters is needed. Because of the random nature of propagation, repeaters must have overlapping coverage for reliability. Any repeater

---

*References are listed at the end of this paper.

network generates a good deal of overhead traffic in the form of acknowledgment and duplicate messages, and some form of network protocol for routing and flow control is needed. Fortunately, the implication of this protocol is readily accomplished by distributing the network control functions in the repeaters.

One design for such a network, currently in the experimental stage, indicates that a microprocessor with 3 μs cycle time and 3K 16-bit words of memory will supply the needed control functions at each repeater to support a 100 kbs throughput.[3] However, this is only a preliminary estimate and subsequent experiments may suggest more or less computing capacity.

Terminals interfaced to this network must also have the capability to perform the packet formatting and network protocol functions. It has been found that terminals with microprocessors are generally more cost-effective in terms of size, weight, and power consumption than terminals without central processor unit (CPU) power. Such hard-wired units tend to use radio channel resources inefficiently.

Several years have elapsed since the Robert's paper, and microcomputer technology has emerged from its infancy. Because of this single major innovation, the outlook for a packet radio terminal has radically changed. In this paper we reexamine, in light of today's technology and the requirements imposed by packet radio development efforts,[3,4,5] the specifications and design issues for a digital terminal for packet broadcasting. We also discuss how these issues have influenced the design and implementation of two terminal prototypes fabricated at the University of Hawaii and SRI.

II FUNCTIONAL ORGANIZATION OF RADIO TERMINALS

Figure 1 is a functional diagram of a radio terminal. It is shown in three parts: radio communications (RC), network access and control logic [network control logic (NL)], and input/output (I/O). We are accustomed to thinking of terminals primarily as an I/O interface device because in a wired network the I/O is usually packaged separately and separated from the NL by very simple communications devices and long wires. However, in a radio network, the NL and RC devices must be physically adjacent to the I/O, or the mobility advantage of the radio will be lost. Hence, a radio terminal must be approached from a new point of view; it must contain a share of the NL.

The communications package in a terminal containing transceiver, modem, and codec is best designed for a specific network, since frequency, modulation, and coding may be different in each network. Thus, a single design that is able to operate in several networks would be very inefficient. In short, the RC package is network-specific and should be hard-wired.

On the other hand, network access and control logic are likely to be similar enough from one packet network to another, in terms of logical functions and required throughput, so that such functions may be efficiently implemented using a microprocessor. In fact, since a few of the net-control functions overlap in time, time-sharing a microprocessor CPU and memory may prove the most efficient approach.[6]

The I/O devices are terminal-specific and may be physically integrated with NL or separately packaged. If separately packaged, then standard I/O devices such as CRT or TTY wired-net terminals might be used; however, since no small portable I/O terminals are available, this approach must sacrifice much in mobility.

Thus, four terminal packaging configurations are possible:

(1) Separately packaged RC, NL, I/O

(2) Integrated RC, NL with separate I/O

(3) Integrated NL, I/O with separate RC

(4) Integrated RC, NL, I/O.

Each configuration has advantages and disadvantages, and each has been built for experimental purposes to verify these. We discuss the impact of these configurations on hardware and software design in the next section.

III TERMINALS

A. ALOHA System Terminal Control Unit

The ALOHA system[2] terminal control unit (TCU) consists of UHF antenna, transceiver, modem, and buffer. The first versions of the ALOHA TCU were packaged in configuration (1) of the previous section (i.e., RC, NL, and I/O were packaged separately) and the total cost was $8,000 to $10,000. The next version was packaged in Configuration (2)--integrated RC, NL with separate I/O. These first versions used hard-wired logic for net control functions, and the protocols, once set, could not be easily altered.

The most recent version of the TCU, called the Integrated Control Unit (ICU) uses INTEL 8008 and 8080 microcomputers. The ICU is completely programmable and its flexibility enables the use of a variety of different transmission protocols including variable length packets and character-by-character transmissions.

A block diagram of an ICU with an INTEL 8080 microcomputer is shown in Figure 2. The hard-wired interfaces establish synchronization and transmit bytes after converting them to bit serial form. The receiver interface performs a serial to parallel conversion and performs byte synchronization. The functions of the 8080 CPU which are performed in software are:

- Packet receive, which checks the header and text parity of an incoming packet
- Parity generation, which generates parity for both the header and text of an outgoing packet
- Packet transmit, which formats header, adds parity, sends the packet to the radio for transmission, and waits for ACK (acknowledgment) to be posted by the RCV (receive) routine.

If POSACK is not received after a certain preset interval, it sends the same packet to the radio for retransmission. After "n" tries, the routine signals a "failure to transmit." The software also contains a CRT or TTY I/O routine. The state transition diagram of an ICU program is given in Figure 3.

The evolutionary process of designing the various versions of the ALOHA TCUs has indicated that unless speed considerations dictate hard-wired logic, it is always preferable to use programmable logic. The added advantages of flexibility, ease of design, speed in implementation, and lower development costs that microprocessors provide clearly outweigh the speed advantages of hard-wired logic. One particular exception to this rule is the case of the parity encoder/decoder which needs to be implemented in hardware because the present-day microprocessors are not fast enough to meet the requirements.

B. Suitcase Packet Radio Terminal

A different portable packet radio terminal (PRT) has been developed at SRI in conjunction with some experimental traffic studies for a packet radio project. This terminal is packaged in a small suitcase with RC, NL, and I/O integrated. Parts cost approximately $5,000.

The general organization of the terminal is illustrated in Figure 4. Central to the terminal is the system data bus of the National IMP-16L Microprocessor. The CPU, peripheral controller, and modem controller all communicate with each other and with the main memory via the bus. The peripheral controller contains a buffer memory of 256 characters and controls operation of a 72-key ASCII encoded keyboard, an 80-character LED display organized in four 20-character rows, and a 20-character/line printer. The modem controller operates the modem and radio to receive and transmit packets.

In use, the operator generates a message on the keyboard in a local mode. When the carriage return key is stroked, the CPU automatically formats the message into a packet, places the packet in a transmit buffer, and passes control to the modem controller. The modem controller fetches the packet, generates parity bits, and transmits the packet. If the message is successfully received at the central computing facility, an ACK packet is transmitted to the terminal. The modem controller places the ACK, as well as all received traffic, in a receive buffer. The CPU analyzes the packet and takes the necessary action. It may retransmit when no ACK is received, or it may abort, depending on operator-specified parameters, and so forth. The user may specify whether received traffic is to be displayed, printed, or both.

Several lessons have been learned in the development of this terminal. In particular, we found that off-the-shelf microprocessor systems are not densely packaged, do not use power conservatively, and are difficult to interface. Each observation suggests that off-the-shelf microprocessor systems are not optimally suited for future terminals so that the next generation should emphasize the use of arithmetic logic unit (ALU) chips combined with microcoded read only memory (ROM) chips to tailor a microprocessor to the packet terminal.

As noted later, microprocessor technology is changing so rapidly that new devices better suited to the needs of a packet radio terminal may ultimately be available. At that time the flexibility of an off-the-shelf device may cause a change in design philosophy.

We have found that the microprocessor component should have both a bit-serial interface to exchange packets with the RC component (and characters with standard TTY type I/O devices) and a byte-parallel interface to exchange characters with integrated I/O devices. These two interfaces should be standardized for all broadcast packet networks so that terminals can be interoperated by changing the microcoded software and the RC component.

IV PROTOCOL IMPLICATIONS

A. Demands

Communications protocols, which are essential for an orderly flow of information to and from the terminal, place a heavy burden on digital terminals. Introducing a digital radio broadcast system places even greater demands on the logical capability of the terminal. This is primarily because terminals must accept traffic as it is offered; that is, there is no significant memory capability in a radio channel.

Because traffic must be accepted in an absolute on-line real-time sense, the terminal must be carefully designed around the network protocol. Thus, data rates and packet formats become crucial design elements. From a user point of view, it is essential that the radio system be transparent, that is, the user must view his terminal as a conventional time-sharing terminal. Thus, power/on-power/off functions must automatically introduce the terminal into the radio network and correspondingly indicate the terminal's departure. Acknowledgments, error control, retransmissions, and a host of other protocol issues must be imbedded in the terminal and invoked automatically.

In a sense, the protocol issues pervade the entire design of the terminal. Because buffering is related to acknowledgment procedures and ultimately to a display (or output) philosophy, it is apparent that protocol affects the organization and control of the terminal's peripherals. There are also impacts in the area of interrupt structure, keyboard interface, and so forth.

B. Protocol

The key protocol issues which must be addressed in a terminal include:

- Validation of ID
- ACK/text discrimination
- Duplicate packet rejection
- Error control
- Text handling (buffering)

- Transmission and retransmission logic
- Encrypting of text.

The issues of packet routing through a network of radio relays have considerable impact on terminal logic;[4] however, these issues are considered outside the scope of this paper.

To illustrate the protocol aspects of terminal organization, it is useful to examine an exemplary digital packet radio format. Figure 5 represents a typical simplified ALOHA format.

The ALOHA system staff have found that three general packet formats meet their needs. These include an ACK packet (header data only) and two text packets (either 40 or 80 characters).

Inasmuch as a radio transceiver has no way of determining a priori what type of packet is being received, it is clear that it must make such a determination on the fly. Therefore, certain fields in the packet header must be searched.

The first logical check performed is to analyze the ID field to ascertain whether the terminal is the proper destination. Presuming the packet is directed to this terminal, the processing continues. Otherwise, the receiver is reset. Parity checks are ordinarily conducted in hardware in parallel with the software-controlled tests of header fields. (This discussion assumes the packet was received with no errors.)

Given a valid ID, the terminal must then check the type of packet. In our example, this means examining Bit No. 10. If the bit is set, indicating an ACK packet, the terminal must check its transmission buffer to verify whether a recently sent packet is awaiting acknowledgment. If so, the transmission buffer and the receive buffer occupied by the ACK are released.

If the packet is not an ACK, it is assumed to be text (in our example). In this case, the ALT Bit (Bit No. 11) is checked to reduce the probability of receiving the same packet twice since this bit is complemented every time a new packet is sent to the terminal. If the terminal fails to acknowledge receipt (or the ACK is not received at the sender), the packet is resent with the ALT bit unchanged. In the case of a duplicate packet, it is ignored; the receive buffer is released, and the ACK transmission logic is exercised again. Note that this particular discussion is idealized--the ALOHA terminals do not currently acknowledge received traffic and this discussion applies only to traffic received at the station.

Assuming a valid text packet is arriving, the terminal then checks Bit No. 9 to determine whether it is a 40- or 80-character packet. This usually requires setting a hardware counter in the modem interface so that text parity is checked properly.

Presuming all of this logic is satisfied, the terminal is then free to output the packet at its leisure.

In our particular example, it is possible for several errors to occur. A packet may be received with parity errors in either the header, text, or both. These errors can be monitored or ignored since the terminal has the option of accepting the packet or immediately resetting. From an experimental point of view, it makes sense to monitor errors since the channel is effectively blocked for the packet duration.

If errors are monitored, the station is frequently used to send control packets to each terminal for error counts to be broadcast back to the station.

Transmission logic for a broadcast channel is limited but not particularly complex. Protocol demands that a transmitted packet be saved until an ACK is received from the destination. If an ACK is not received in a predetermined time, the typical protocol dictates retransmission in a pseudo-random time interval. Pseudo-random times are selected to reduce the probability of several terminals jamming each other repeatedly while competing for the channel. In the ALOHA system, a packet may be retransmitted up to five to eight times before the terminal gives up and notifies the user. Inasmuch as most time-sharing system users require their traffic to arrive in sequence, it is common to have only one or two transmission buffers and to lock the keyboard when the buffers are filled.

C. Software

1. Terminal Control

Terminal control is readily exercised through a microprocessor supervisor or executive programmer. Because of the real-time demand of the radio channels and the less constrained output demands, it is convenient to think of the software as being organized in a foreground and background mode.

In such a "multiprogrammed environment," the foreground partition is interrupt driven to accommodate the modem interface and its real-time demands. Such a partition must have first priority and if the software is organized properly, the response time to interrupts can satisfy most data rate requirements.

In our terminal, the data transfer between modem interface and memory is performed under DMA control. Since data are transferred on a word by word basis, the processor has more time to react, and even slow microprocessors can accommodate large data rates (in excess of 100 kbs).

The foreground partition is generally devoted to the receive logic and receive buffer control. Certain modem-dependent transmit codes must also reside in the foreground; however, the general transmit logic (e.g., retransmission timing, and so on) is less time dependent and therefore can reside in the background partition.

The background partition ordinarily contains the terminal's peripheral control routines. Such activities as display, edit, format, print (if appropriate), keyboard interface, and so forth, are monitored in the background. Data transfer between peripherals and the CPU is most conveniently handled under CPU control so that the hardware interface is simpler and standard programming techniques can be used.

2. Interrupt Polling

The interrupt structures and options provided by microprocessor vendors are varied. In the suitcase terminal we have used a National Semiconductor IMP 16/L CPU. This machine has four interrupt levels. One level is vectored and is an obvious choice for responding to the modem interface.

3. Software Development

Software development for digital radio terminals is limited by the constraints inherent in microprocessors. There are not only limitations in the instruction sets but also memory problems, speed conflicts, and poor (in general) programming aids.

D. Programming Aids

In practice, microprocessor vendors provide both resident-assemblers and cross-assemblers. In the IMP-16L case, National Semiconductor provides a cross-assembler for the IBM 360. INTEL provides an algebraic processor in addition to assemblers.

The National Semiconductor resident-assembler is provided in object form on paper tape. Approximately 25 minutes are required to load the assembler. As a three-pass assembler, it requires entering the source code three times. An inherent disadvantage in the National Semiconductor software is the inability to load the loader and assembler in memory simultaneously. Thus, the constant loading and reloading compounds the debugging problems considerably.

However, the cross-assembler can be a significant program development tool. The National Semiconductor cross-assembler is supplied in source FORTRAN IV for an IBM 360. SRI modified this code extensively and implemented it on a DEC PDP-11/20 with 28K words and a Vector General Display System. This system provides high speed paper tape facilities and an extremely powerful editing system. In addition, it was very successful in providing hands-on debugging at the source level. An emulator would be even more beneficial.

Our software experiences with microprocessors were not surprising. They are, indeed, much less sophisticated than minicomputers and the software support from the vendor is limited--at best. Our experience with the IMP-16L software was very unsatisfactory as supplied; however, our investment in the cross-assembler was very worthwhile. Furthermore, we would expect similar quality in software supplied for any new digital processor--micro, mini, or large main frame.

V TECHNOLOGICAL CONSIDERATIONS

A. Physical Characteristics

The natural evolution of packet broadcast terminals has been toward greater portability and lower cost. Initially, terminals packaged in Configuration (1) were not portable, and cost $8,000 to $10,000. Currently, the ICU occupies .6 cubic feet; weighs 15 lb without keyboard, display or battery; and costs $2,000 in parts. The battery pack, including charger, is the size and weight of an automotive battery. Although the latter two terminals were developed with the intention of portability, little attempt has been made to minimize their size or weight.

To realize the full potential of packet broadcasting, future design efforts must concentrate on achieving a single physical package containing RC, NL, I/O, and power supply. The I/O must be engineered from a human factors viewpoint to be convenient, easily learned and operated, and must be designed to avoid operator fatigue. The self-contained power supply should provide a minimum of four hours continuous operation and should be readily recharged or inexpensively replaced. Finally, the entire package should be as small and light as possible consistent with other objectives. In this section we discuss the possibility of applying existing technology to achieve the goals and discuss where advances are needed.

B. Input/Output

The I/O elements interface the man to the network. If they are poorly conceived or implemented, the best technological design of all other network elements cannot compensate for these deficiencies. In this paper we assume that the input element is a keyboard, because that seems the most likely initial component. Subsequent study may show that some other form of input (such as hand-written characters or spoken words) is preferable. Similarly, we have assumed an alphanumeric display as the most likely initial candidate, although subsequent study may not support this assumption.

1. Displays

The important physical characteristics of the display include character clarity, size, color, and contrast; display format; power consumption; and overall size. The trend of portable display technology seems to be toward 5×7 or 7×9 dot matrix characters between 0.1 and 0.2 in. high, although nine and fourteen-segment displays are available. Most displays (LED or Plasma) are red on black with some yellow and green displays now on the market. Liquid crystal display color depends on ambient light, and virtually any color is possible. Since character size will determine the maximum total number of characters displayed, a determination of minimum acceptable character size is very important. A human factors study of character size, color, and contrast for portable terminals would help greatly in design of a suitable display; however, pending the results of such a study early terminals will depend more on selections of available display devices.

The display format most desirable is probably a 12- to 16-line page with 72 or 80 characters per line. Such a display may eventually be possible with 0.1 in. characters on a 3×8 in. area; however, except for CRT displays, this density is not available today. If the maximum display dimension is limited to 8 in., then off-the-shelf technology of either LED or plasma displays limits the number of characters per line to 40. Using some advanced LED technology not yet in production, it would be possible to construct a display of twelve 40-character lines in an 8×3 in. area; however, the cost and power consumption would be excessive.

The normal dot on a standard LED matrix requires approximately 30 mW of input power. Assuming 20 active dots per character, a 40-character line would require 24 watts. To be consistent with desired size and weight properties, the display should consume no more than one or two watts. Thus, a single-line 40-character dot matrix LED display would require a 24-fold improvement in efficiency, and it is not likely that a multiple-line dot matrix LED display will ever be satisfactory. Other LED configurations are possible, however, both 9-segment and 14-segment character fonts are available. If an average of 5 segments are needed to display a character, the power requirement would be only 1/4 that of a dot-matrix.

It is possible to use a CRT display as an interim solution. A 4 in. CRT will display 12 lines of 40 characters in very readable fashion. Such a display can be packaged in a 250 cubic in., 8 lb, 11 watts package. However, it is not likely that any significant size, weight, or power reductions are possible, so the CRT is not a promising solution.

Liquid crystal matrix displays recently announced by Hughes[7] and Hitachi[8] will approach CRT dot density at low power-consumption, and may provide the desired full-page display; however, these displays are not yet in production, and no detailed specifications have been published.

A great deal of research effort is being directed toward alphanumeric displays, with goals of obtaining higher character density and lower power. This brief description was not intended to be a survey, but to indicate that the display requirement for a packet broadcasting terminal, though severe, will soon be met.

2. Keyboards

In a sense, the keyboard is a more difficult problem than the display, since it must provide the ability to enter any one of a large number of characters. Furthermore, it must be arranged so that a large finger or fat thumb will depress only a single key.

Many approaches are possible. Roberts proposed the use of a binary encoded 5-key device developed and used at SRI.[9] Although this device may be operated almost as rapidly and mastered more quickly than touch-typing, it has the disadvantage of many specialized computer devices that must be learned. Uncoded keyboards (every key is a separate character) have the disadvantage that the number of keys must equal the size of the character set; however, a novice can use an uncoded keyboard with no instruction. A reasonable compromise is a partially encoded, or multifunction keyboard such as that used on most hand calculators. These keyboards can be made self-explanatory so that a novice can use one by examining the labels of the keys. If at all possible, a standard TTY layout of the alphabetic keys should be used with the interkey spacing and switch "feel" as similar to TTY keyboards as possible. A possible keyboard organization would divide the character set into three subsets: uppercase (standard data entry); number symbols; and control characters (TTY control set). Two shift keys, a space bar, and an "enter" key are also needed. Special functions such as character or line delete can be encoded as part of the number/symbol case. Such a keyboard, with standard spacing, will occupy only 8 X 4 in. and will be almost as easily used as the TTY or typewriter keyboard. An example of such a keyboard with a nine-segment, 32-character LED display is a terminal manufactured by MICON Inc., for communications use by the deaf.

To obtain suitable contact pressure and overtravel, a standard set of keyswitches could be used; however, many other technologies offer thinner keyboards with slight loss of "feel." Although these cannot all be reviewed here, the conductive elastomer keyboard can be considered as representative. Such a keyboard can be designed with 0.8 in. travel and fitted with a silicon cover to provide an impression of overtravel. It is only 0.25 in. thick (including the silicon), and has no holes or cracks for entry of dirt or water; this is a very desirable feature in a portable terminal.

C. Network Control Logic Components

An examination of the internal hardware design problems leads to the conclusion that to eliminate extra circuitry, all inputs and outputs to the network control logic should be handled in their natural form, and control functions should be centralized in a microprocessor.

A wide variety and number of microprocessors are available today, and technology is changing so rapidly that each month a number of new devices appear on the market. Although microprocessors were originally introduced by semiconductor houses to help sell memory, they are also sold as complete systems, nominally for prototyping new products.

The microprocessors available vary greatly in speed, number of bits per CPU, number of chips per CPU, type of instruction set, and power requirements. Compared to minicomputers, today's MOS microprocessors are limited in all performance factors; however, recently announced or planned microprocessors which use SOS, IIL, bipolar, and low-power Schottky technology are rapidly approaching minicomputer performance in all parameters. A sampling--by no means inclusive--of available and announced microprocessor chip-sets and chips is given in Table 1. This table is not complete, but is provided only to show the wide variety of performance soon to be available. Reference 10 can be consulted for more complete information.

Since the radio I/O logic is naturally serial it should be handled by the central microprocessor in that form. The microprocessor definitely must have the power to do serial to parallel or parallel to serial conversion to satisfy other requirements so these functions should be centralized.

Centralizing the conversion processes also allows the microprocessor to take on the burden of packet synchronization. All packet synchronization and parity checks should be accomplished in microcode. Bit synchronization must be accomplished in the modem. Encoding and decoding processes are independent of other functions and should be incorporated in hybrid circuitry external to the microprocessor and modem modules.

Table 1

MICROPROCESSORS

Manufacturer and Identification	Semiconductor Technology	Instruction Fetch Cycle (μs)	Available 1974
Fairchild PPS-25	NMOS*	62.5	--
RCA COSMAC	CMOS†	3.0	No
Intel 8080	NMOS*	0.5	Yes
Intersil IM 6100	CMOS†	0.5	--
T.I.	IIL ‡	0.5	--
Inselek	SOS§	0.3	No
Raytheon RP-16	Bipolar	0.2	--
Monolithic Memories 6701	Lowpower Schottky**	0.15	--
Intel 3000 Series	Bipolar	0.12	Yes

*N-channel metal oxide silicon.

†Complementary--metal oxide silicon.

‡Integrated injection logic.

§Silicon on sapphire.

**Low-Power Schottky.

Other I/O functions should be accomplished in parallel to take advantage of their natural form. Data to be displayed should be output in parallel form. The display itself could be addressed as part of the microcode memory space or by a separate register and bus using the microprocessor hardware. The data should be decoded from standard ASCII on the data bus through a row-column generator. The display itself should be refreshed by microcode during the idle state of the microprocessor. When the processor is busy checking parity or transmitting, the display could probably be blanked for short time intervals without affecting the user. Blanking the screen could be used to notify the user that his packet had been transmitted or that a new packet had arrived and was in the terminal. Keyboard data should be encoded into an 8-bit code and input in parallel, since the keyboard lends itself to a parallel format and the data bus will be 8 bits wide. Assuming a 32-key multifunction keyboard, such as described in Section V, is used, 5 bits can be used for the 32 keys, and the other three bits for the "enter" and "shift" keys. A sample coding format is shown in Figure 6.

This formatting allows quick table lookup in microcode to translate to ASCII from keycode input using a 7-bit address.

An initial conservative estimate of memory size for the microprocessor is 256 bytes of read/write RAM for buffering and 2K words of 16-bit ROM for microcode. These estimates are based on the memory requirements of the ICU modified to compensate for the microcode type of operation and a 128-character display size. For example, a pair of Intel 8316 MOS ROM may make an attractive circuit package for holding the microcode. The circuit, organized 2048 $\times$ 8 bits, has a low power dissipation of 10.7 μW/bit and runs from a single 5 volt supply. This pair of ROM packages provides the required 2K words of microcode using two 24-pin spaces. Although the masking charges are expensive, a standaard ROM package to hold microcode has advantages over a special CROM package which contains control logic for the microprocessor. A writable microcode memory external to the package can be substituted for the ROM for testing and microprogram development.

Judging by the current rate of change in the industry, suitable microprocessors will probably be available in a year or two; however, to meet the power and size requirements in the interim the microprocessor element probably must be custom designed. To restrict power consumption it will probably be necessary to construct the unit using hybrid techniques and low-power Schottky MSI devices. Speed is extremely important because of the serial interface to the radio; however, fancy microinstructions are not. The microprocessor needs only the primitive operations, such as AND, OR, XOR, RIGHT-SHIFT, ADD, COMPLEMENT, and INCREMENT, plus a few positive and negative branch type instructions. It must also have some internal routing microinstructions. Experience with the ICU indicates that microinstructions should execute on the order of 200 ns/instruction. Typical power consumptions for ALU and microcomputer integrated circuits are shown in Table 2.

The complexity and speed vary quite drastically from the 74LS181 which can perform one of 32 operations on two 4-bit wide binary numbers in 25 ns to the 8008 Intel CPU which can perform an 8-bit ALU operation in 20 μs. To achieve adequate speed performance, a low-power Schottky implementation is probably necessary. Such a unit with memory should consume no more than one watt.

Table 2

POWER CONSUMPTION

Technology	Integrated Circuit		Power/Circuit
MOS	4004	Intel CPU 4 bit	420 mW
MOS	8008	Intel CPU 8 bit	420 mW
MOS	8080	Intel CPU 8 bit	1 W
BIP	3001	Microprogram control	900 mW
BIP	3002	Central processing element	750 mW
LS	74LS181	Arithmetic logic unit	125 mW

D. Radio Communications Components

Transceiver technology is available to allow very small, low-power packages; however, it must be applied to the specific modulation and coding design for packet broadcasting. The Motorola Dynatac[11] terminal contains a transceiver and digital modem which would satisfy the ALOHA requirements as to bit rate and transmitter power. The Dynatac package occupies 60 cubic in. and includes touch-tone pad, headset, audio circuitry, control logic, and batteries. It seems likely that the transceiver and modem portion occupy no more than 10 cubic in. Other efforts are underway to apply thick film hybrid technology to miniaturize packet broadcasting RF and modem circuitry. These promise to achieve required performance in a 10 cubic in. package with receiver power consumption below one watt.

Although the transmitter peak power is nominally 10 watts, the duty cycle will be very slow so that the transmitter will require only a few milliwatts average power. The power source must be able to supply this low average power in short 10-watt bursts.

E. Power Source Considerations

The power source will most probably dominate the other components in determining the size and weight of a packet broadcast terminal. The ICU, containing no keyboard or display, requires 15 watts. The suitcase terminal, including an 80-character display and full ASCII keyboard, requires an additional 33 watts. Although components were carefully selected to conserve power, no attempt was made to go beyond components available off-the-shelf.

As discussed above, it is probable that a terminal can be developed which will require no more than 5 to 10 watts, with the display being the unknown factor.

Batteries are available in power densities varying from 0.5 Whr/cubic in. and 10 Whr/lb to 5 Whr/cubic in. and 100 Whr/lb. Assuming that inexpensive rechargeable batteries will be used, a nominal density of 20 Whr/lb and 1 Whr/cubic in. are possible, so that a 20 Whr battery pack will weigh one pound and occupy 20 cubic in.

With this battery, the terminal will provide from two to seven hours continuous operation depending on the display power required.

Regulation of battery-supplied power can be power consuming if close voltage tolerances are required. Selection of logic families and circuit design that are tolerant of voltage variation are major design considerations.

VI CONCLUSION

With today's technology, a small, lightweight personal terminal is within the state of the art. The display is the only unsolved problem; however, as the liquid crystal dot matrix displays recently announced are brought to production, that problem will disappear. Suitable efforts concentrated on developing an RF hybrid package, a microcoded microprocessor, and packaging the entire unit should result in a terminal which occupies no more than 100 cubic in., weighs less than 5 lb, and costs on the order of $3,000. Quantity production can reduce this cost drastically.

Although we have only discussed possible alphanumeric terminals, future technology will make other types possible.

Using packet broadcasting technology, it will be possible to make very simple one-way terminals to either send or receive messages. Transmit-only terminals may find application in monitoring remote sensors such as weather metering instruments or the state of traffic at a busy intersection. Receive-only terminals may be used to change traffic signals or possibly to control remote advertising signs.

Current efforts to digitize speech may result in very compact, low-power speech digitizers that could be combined with packet broadcasting technology to provide hand-held terminals with both direct voice communication and data I/O using remote word recognition at the central computing station.

To understand the operational context under which the ALOHA and suitcase terminals were developed, please refer to the other papers on packet radio in these p.oceedings.

REFERENCES

1. L. G. Roberts, "Extension of Packet Communication Technology to a Hand-Held Personal Terminal," Proceedings of AFIPS, 1972 Spring Joint Computer Conference, Vol. 40, pp. 295-298.

2. N. Abramson, The ALOHA System, "Another Alternative for Computer Communications, "Proceedings of AFIPS 1970 Fall Joint Computer Conference, Vol. 37, pp. 281-285.

3. S. C. Fralick and J. C. Garrett, "Technological Considerations for Packet Radio Networks," National Computer Conference 1975, Anaheim, California, Proceedings of this conference.

4. H. Frank, I. Gitman, and R. Van Slyke, "Packet Radio System: Network Considerations," National Computer Conference 1975, Anaheim, California, Proceedings of this conference.

5. J. Burchfiel, R. Tomlinson, and M. Beeler, "Functions and Structure of a Packet Radio Station," National Computer Conference 1975, Anaheim, California, Proceedings of this conference.

6. S. Fralick and D. Brandin, "The Role of Microprocessors in High Speed Portable Data Communications Terminals," Proceedings of Journees d' Electronique, Lausanne, SW, 1974.

7. "Developments," Computer Design, March 1974, p. 42.

8. "International Newsletter," Electronics, October 3, 1974.

9. D. E. Englebart and W. K. English, "A Research Center for Augmenting Human Intellect," AFIPS Conference Proceedings, Vol. 33, p. 397, 1968.

10. Hermann Schmid, "Monolithic Processors," Computer Design; Vol. 13, No. 10, October 1974.

11. "The Dynatac Concept and the 900 MHZ Mobile Radio Band," Motorola Technical Report submitted to the FCC related to docket No. 18262, April 1973.

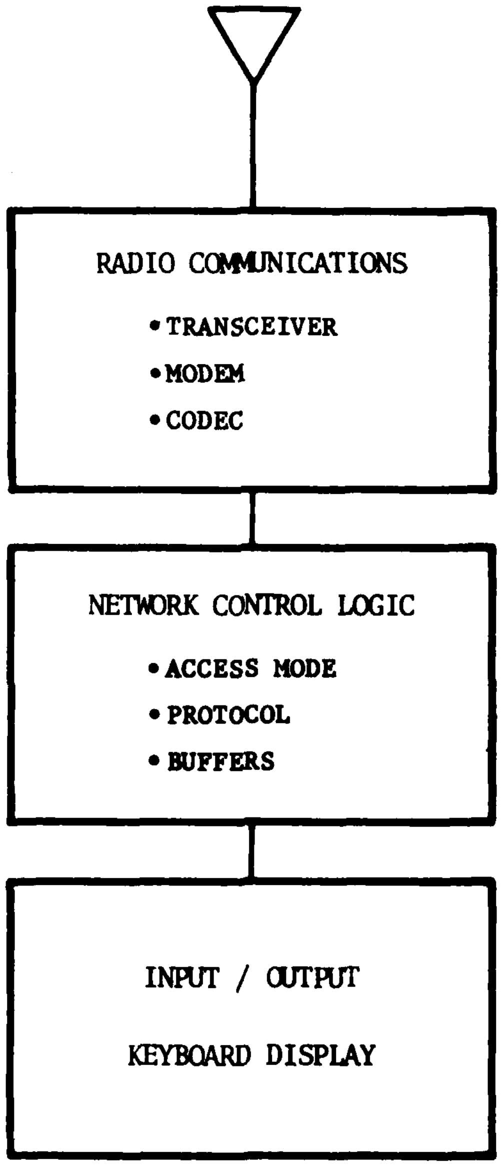
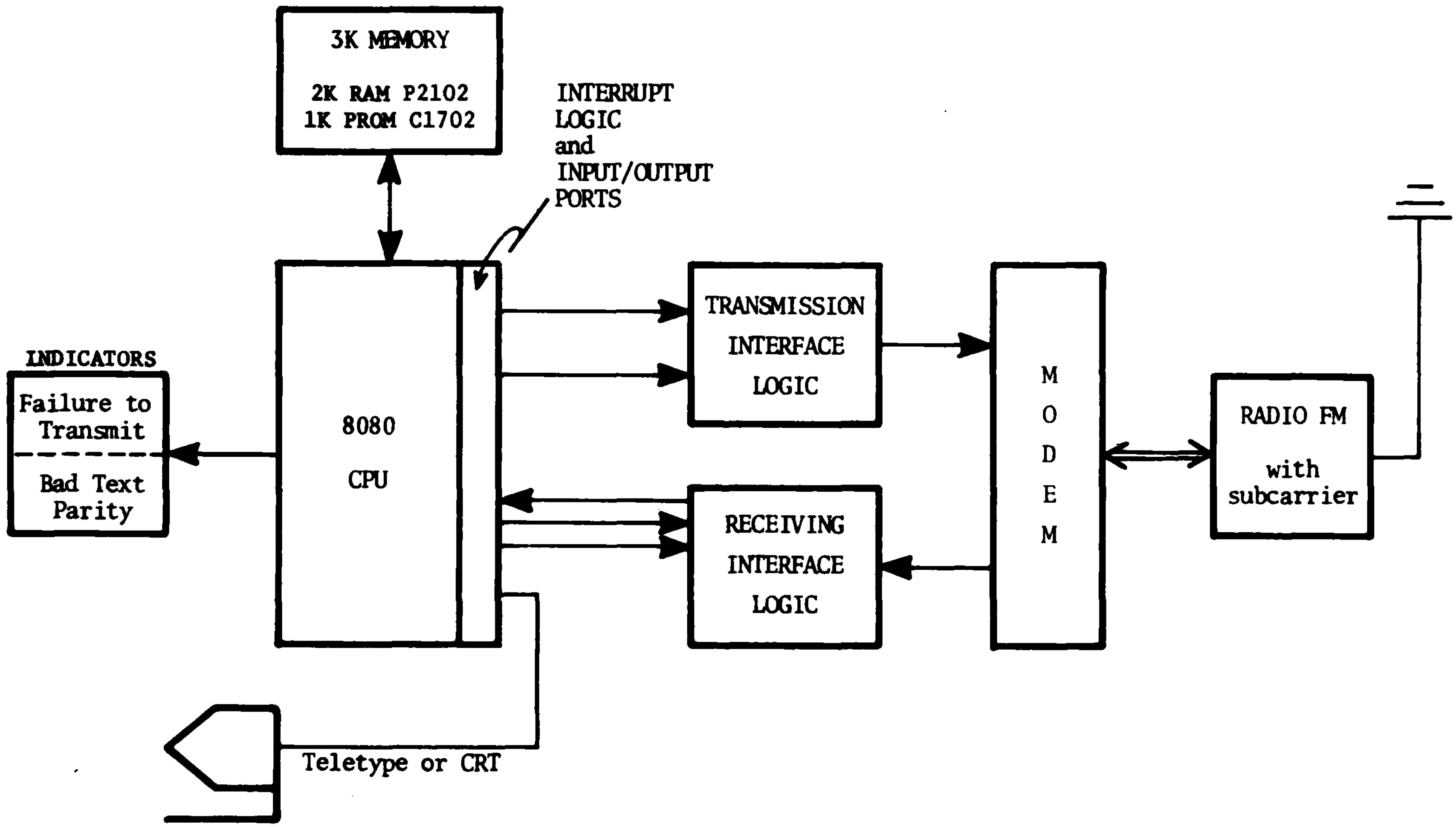
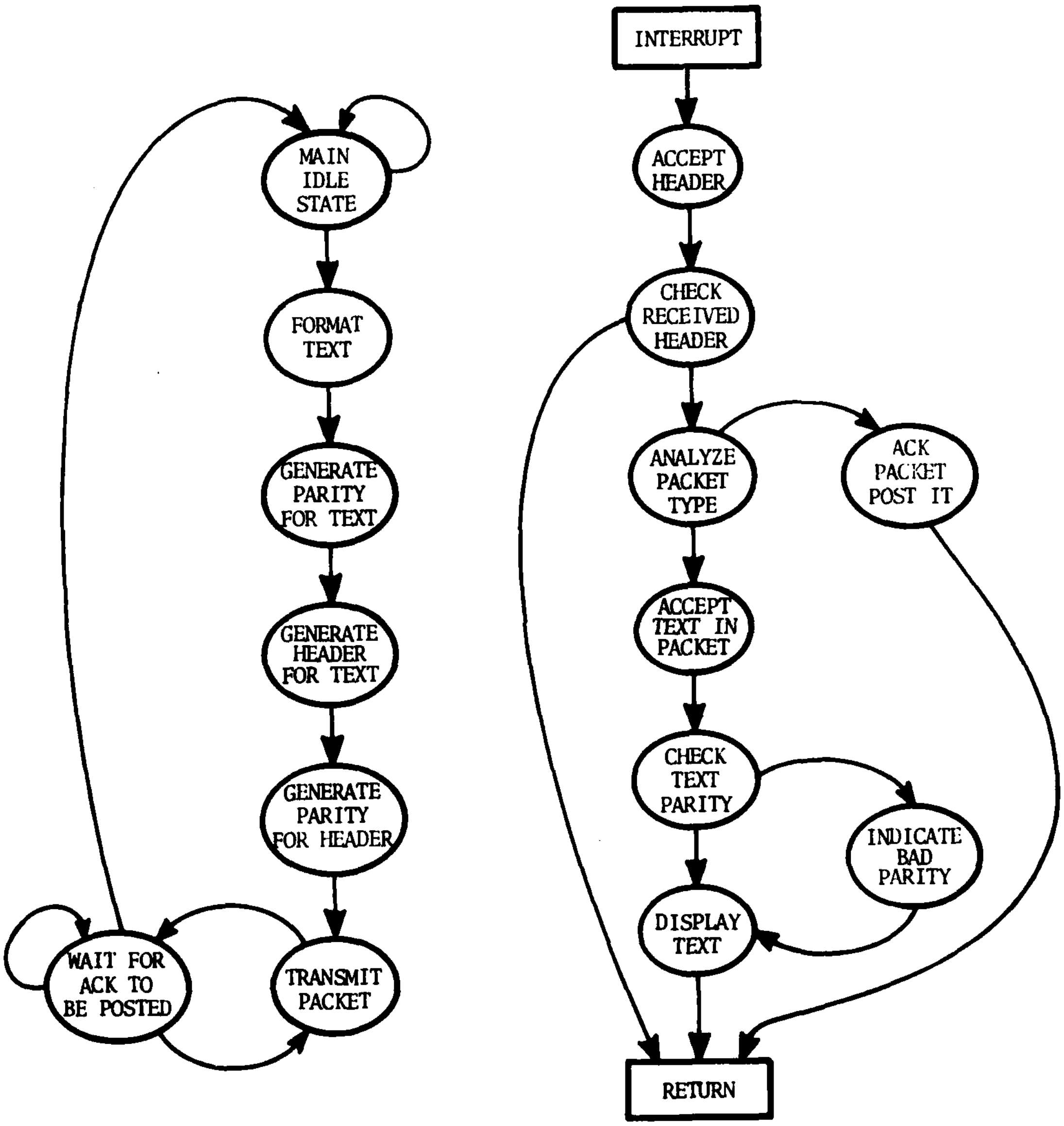


Figure 1. FUNCTIONAL DIAGRAM OF RADIO TERMINAL

Figure 2. ICU BLOCK DIAGRAM





PACKET TRANSMISSION PROCESS

PACKET RECEPTION PROCESS

INTEL 8080 ICU TCU PROGRAM

Figure 3. STATE TRANSITION DIAGRAM

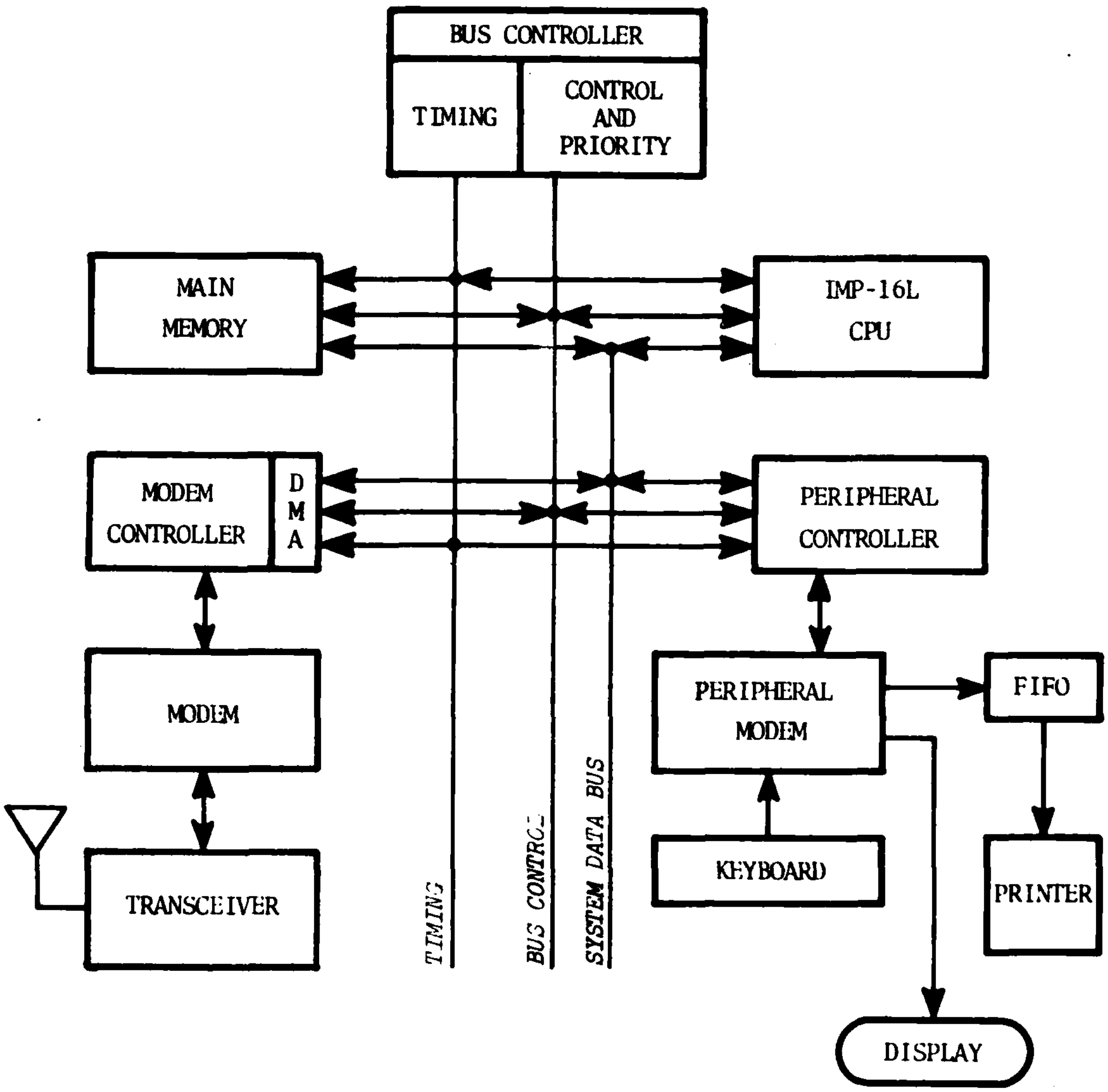


Figure 4. TERMINAL ORGANIZATION

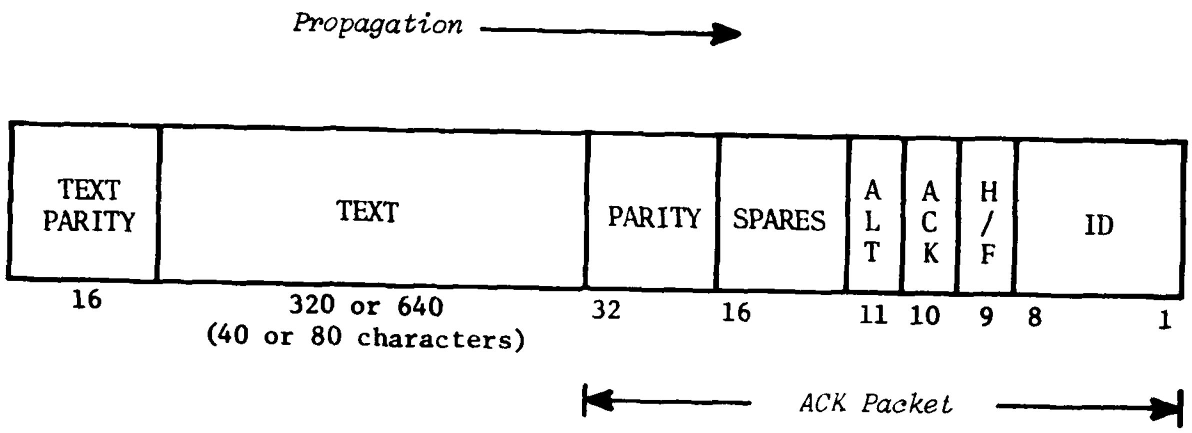


Figure 5. TYPICAL PACKET FORMAT

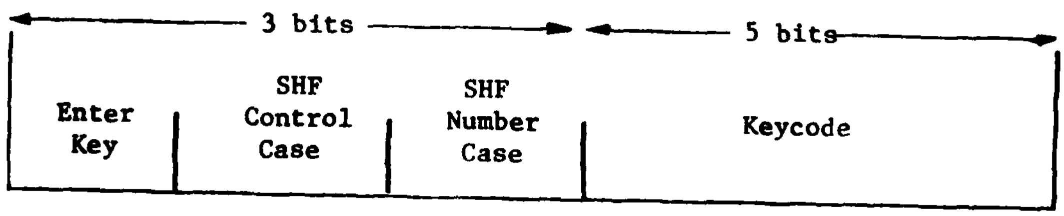


Figure 6. A SAMPLE CODING FORMAT

END

FILMED

2-83

DTIC