QEX 31 September 1984 $1

The ARRL Experimenters' Exchange

ARRL Ad Hoc Committee on Amateur Radio Digital Communication

The ARRL digital committee will meet at ARRL Hq. on September 15-16 to explore possible agreement on packet-radio network-layer protocol standards. The current debate seems to center around datagram vs. virtual-connection protocols, with proponents on both sides. It promises to be a spirited debate.

At the same meeting, we hope to wrap up the work done by the committee on packet-radio link-layer protocols and report a finalized document to the ARRL Board of Directors at their October 1984 meeting.

We are pleased to hear that the Radio Society of Great Britain (RSGB) is sending Ian Wade, G3NRW, and John Sager, G8ONH, as observers. In addition to the regular committee members, there will also be observers from most of the packet-radio clubs in the United States.

Amplitude Compandored Single Sideband Contacts

Lab engineer Greg Bonaguide, WA1VUG, made a number of successful ACSB-to-SSB contacts via the OSCAR 10 satellite from W1AW. An ACSB exciter was used to drive an up-converter, an ICOM IC 471A and a Mirage D1010 100-W amplifier. Other stations using normal SSB receivers were able to copy W1AW tests without any problem. We are now in the process of obtaining more ACSB equipment so we can properly demodulate the ACSB signal and make some A-B comparisons. Additional equipment will make tests possible between W1AW and Vern Riportella, WA2LQQ, in Warwick, NY, and Project OSCAR experimenters on the West Coast.

Hamarama

The Mt. Airy VHF Radio Club, also known as the Pack Rats, will hold their annual Mid-Atlantic States VHF Conference on Saturday October 6, 1984, at the Warrington Motor Lodge, Warrington, PA. From PA Turnpike Exit 27, take Rte. 611 north 4.5 miles to one block north of Rte. 132. The all-day VHF program will be from 9 A.M. to 5 P.M. Luncheon will be available at $7.00. A get-together at the Pack-Rat hospitality suite will follow at 6:30, then a buffet dinner at 7:30. Advance registration is $4.00 prior to September 23, payable to the Mt. Airy VHF Radio Club, Inc., mailed to: Hamarama, P. O. Box 311, Southampton, PA 18966. Registration at the door is $5.00. The banquet is $13.00.

Your registration also includes The Flea Market on the following day, Sunday, October 7, from 7 A.M. to 4 P.M. at the Bucks County Drive-In Theater, Warrington, PA, on Rte. 611, 4 miles north of PA Turnpike Exit 27. If you didn't register for the VHF conference, the flea market registration is $3.00.

Jeff Ward, K8KA, and Mark Wilson, AA2Z, from ARRL Hq. will be there -- will you? Jeff will speak on packet radio and will bring an audio tape of amplitude compandored single sideband modulation as heard through OSCAR 10.

Are You Going to be Surprised...

...when you see the 1985 Radio Amateur's Handbook. Even the "nay sayers" and the every-four-or-five-year Handbook purchasers will do a double take when they see it. We'll say no more except to watch for the spread in QST announcing the 1985 edition of the Handbook, then take a look through one at the first opportunity.

Call for Technical Articles

We have been getting a good stream of articles for QEX overall, although there have been a few times when we could have used more material. I for one would like to see some more articles on VHF/microwave experimentation. There is certainly some excellent experimentation going on in Amateur Radio, but much of it is underreported. There is always the problem that if the experimenter takes the time to write, then not much experimenting would get done. On the other hand, there are some good writers looking for something of substance to write about. Isn't there a way of the two getting together? Writers: Find an excuse to visit that bearded experimenter in his damp basement and ask him what wonderful things he's doing. Experimenters: Call that literate friend to bring note pad and camera.

Correspondence

Rather than writing a long-winded article, sometimes you can get your point across in the QEX Correspondence column. If you have a technical question that you think one of your fellow QEX readers can answer, send us a brief note for this section. Experience with such quests show that QEX gets answers. If copies of the answers are sent to us, we'll publish them too. QEX has a short lead time for correspondence — on the order of 3 to 6 weeks. - W4RI

Correspondence

Comments on the Heathkit ET-3400

I hate to write this at the same time you are soliciting input for QEX and not subject requests, but after reading Donald G. Varner's modification article for the Heathkit ET-3400 in the June issue of QEX, (no. 28), I wish to suggest to the readers that there is a lot of power in the ET-3400 for RTTY, AMTOR, and PACKETRADIO. I do not have the time to develop such applications as I am involved with publishing a Spanish bulletin, the Venezuela Column in 73 Magazine, and technical articles for several other magazines. I'm sure a project like this would be successful and certainly of interest for the readers of QEX. It might also provide extra money to the author if the program is available in ROM. -- Luis E. Suarez, OA4KO/YV5, Apartado 66994, Caracas 1061-A, Venezuela.

In Search of Quartz Crystals

I wish to purchase a quartz crystal in its natural state -- uncut, 6 sides, medium quality. I would like it large enough so it can be used in a "show and tell" session in a grammar school. Several companies have been approached, but their reply is that they obtain their crystals from "someone else."

Can anyone suggest an importer or dealer who might be able to supply one? -- James M. Bruning, K2BZ, 14 Noel Drive., North Arlington, NJ 07032.

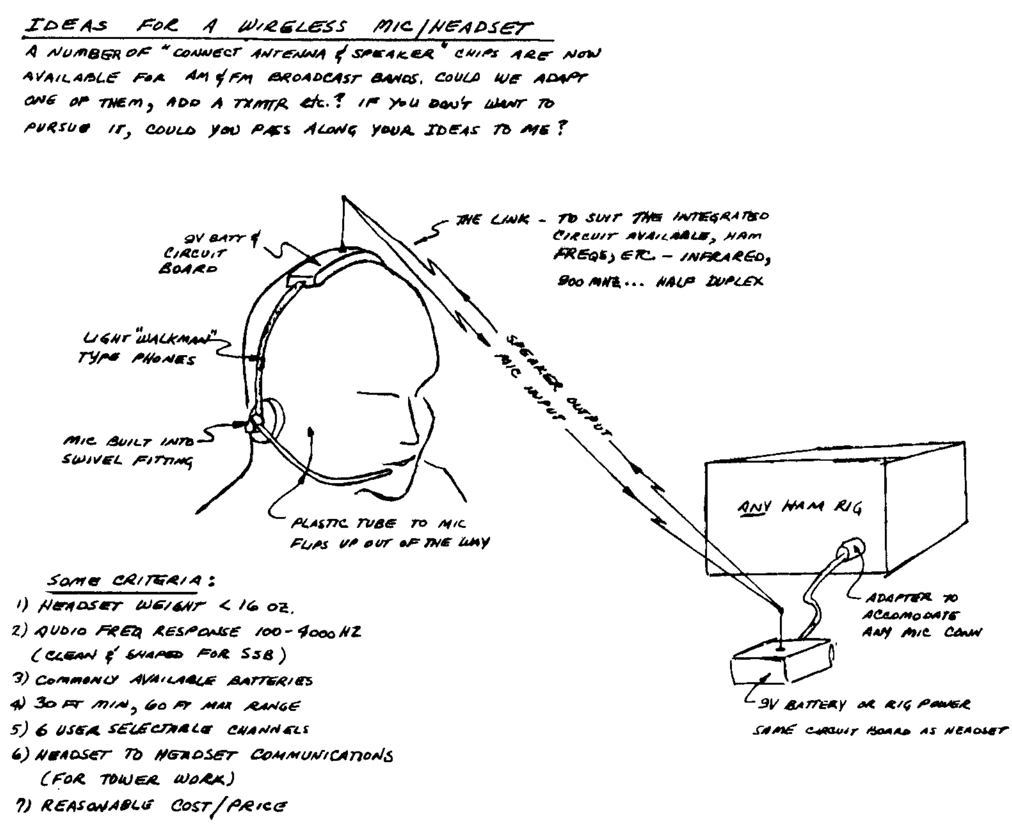
Correction for the Cable Attenuation Program

I would like to bring a conceptual error to your attention that had appeared in the cable attenuation program on page 11 of July QEX ("Coax Loss Program for the HP-97 and TRS-80C," issue no. 29). Because of an incorrect approach, the program is much more complex than necessary. Lines 40 through 90 may be replaced with a single line as follows: A = 2.1715 * LOG (PR/PF).

Simply stated, the line loss is one half the loss calculated from the forward and reverse power. Also, this loss will not be the same as the line loss when it is properly terminated because it is operating at a high SWR. -- Jon Lunder, NØFGO, Star Route, Box 214A, Detroit Lake, MN 56501.

Ideas for a Wireless Mic/Headset

I would like to develop the idea shown below on the subject of a wireless mic/headset and perhaps write a construction article about it. If anyone has information in this field, I would appreciate hearing from you. -- Emile Alline, NE5S, 773 Rosa Ave., Metairie, LA 70005.



Complete RTTY for the TIMEX

By Thomas R. Strohl,* KA1VW

This article presents the hardware and software that I have developed to use with my Timex TS-1000 computer. The software runs on a 2k byte unexpanded system, and listings are available for both Baudot and ASCII operation from the ARRL. [1]

Functions of the System

The hardware interface is based on the Z80A SIO serial input/output device. The SIO contains a dual channel USART, with control registers, handshake control lines, and data buffers for each channel. The device has features that make it quite suitable for amateur use, but its full capabilities are far beyond the necessities of amateur RTTY. In asynchronous operation it automatically adds start, stop, and parity bits (if parity is enabled). It is programmable with respect to stop bits per character, parity, bits per character, and data clock rates (1, 16, 32, or 64 times the data baud rate). Thus, the device can be programmed for ASCII as well as Baudot with any combination of stop and parity bits in use by amateurs. It also allows additional protection against spurious signals generated by faster data clock rates.

Since the Z80A SIO is part of the Z80 family, it is easy to use with the Z80A CPU -- the heart of the TS-1000. Many of the pins of the SIO connect directly to the CPU bus. All inputs and outputs are TTL compatible with the necessary pull-up resistors self contained. It requires a single 5-V dc regulated power supply that can be derived from the TS-1000 9-V supply. In the circuit presented here, the address decoding is simple and straight forward.

There are actually four versions of the Z80A SIO designated /Ø, /1, /2, and /9. Internally they are similar, but the main difference among them is that the /9 is a single channel version. Each channel has 3 read registers and 7 write registers. The read registers provide information about the data passing through the device such as buffer overflow, framing errors, and parity errors. In the asynchronous mode, read register Ø is polled to determine when a received character is available when receiving or if the transmit buffer is empty when transmitting. Only write registers 3, 4 and 5 need be programmed for asynchronous operation in the polling mode. Complete information about the Z80A SIO can be found in the "Z80 SIO Technical Manual." [2]

The price of the SIO depends on the supplier. Prices range from $12.50 to $25 for the /Ø version used here. It pays to search for the best price and I have included pin functions for all the versions should you want to make a substitute. My unit was built at a cost of $25 using breadboard construction and the lowest-priced SIO.

Review of the Circuits

Fig. 1 shows the minimum circuit required for RTTY operation. It will interface any TTL compatible terminal unit to the TS-1000, and permits operation at one speed only, in excess of 300 baud. Fig. 2 shows modifications for multiple baud rates and software TR switching.

Address decoding is performed by U1, U2, and U3. This scheme was adapted from that of J. J. Carr. [3] When address bits Ø, 1, and 6 are high and bits 2, 4, 5, and 7 are low, then all eight inputs to U1 will be high. This causes the output of U1 to go low.

If this should happen when an input/output instruction is being performed by the CPU, the input/output request (IORQ) line also goes low. The two low inputs to the NOR gate (U3) cause its output to go high. This output is inverted and serves as both the chip enable (CE) and IORQ for the SIO. Note that bit A3 can be either high or low. The result is that a single channel of the SIO has two port addresses, 67 and 75. Port 67 is used to pass character data between the computer and the terminal unit. Port 75 is used to pass programming information to the SIO and for passing buffer information to the computer.

As may be apparent, the wiring of the address lines and U2 determine the interface addresses. If you wish to use port addresses other than 67 and 75, be sure the ones selected are not being used by the Sinclair Logic chip. The system will crash if you try to write to an address that is already in use. If you try to read from such an address, a value other than 255 will be returned.

All data lines, M1, RD, ground, and the CPU clock are connnected directly to the CPU bus. Power is supplied from a 7805 5-V regulator connected to the 9-V pin of the bus. The edge connector I used was made by cutting a larger connector purchased from Radio Shack (0.1 inch spacing). A prototyping board, marked edge connector, and expansion board (necessary if you want to attach a printer or RAM pack) are available. [4] If you cut your own, it is a good idea to fit it with a key for the slot to be sure the connections line up properly.

The data clock is built using a 555 timer IC. The clock pulses do not need to be perfectly

*Rural Route 1, Box 123, Hope Valley, RI 02832

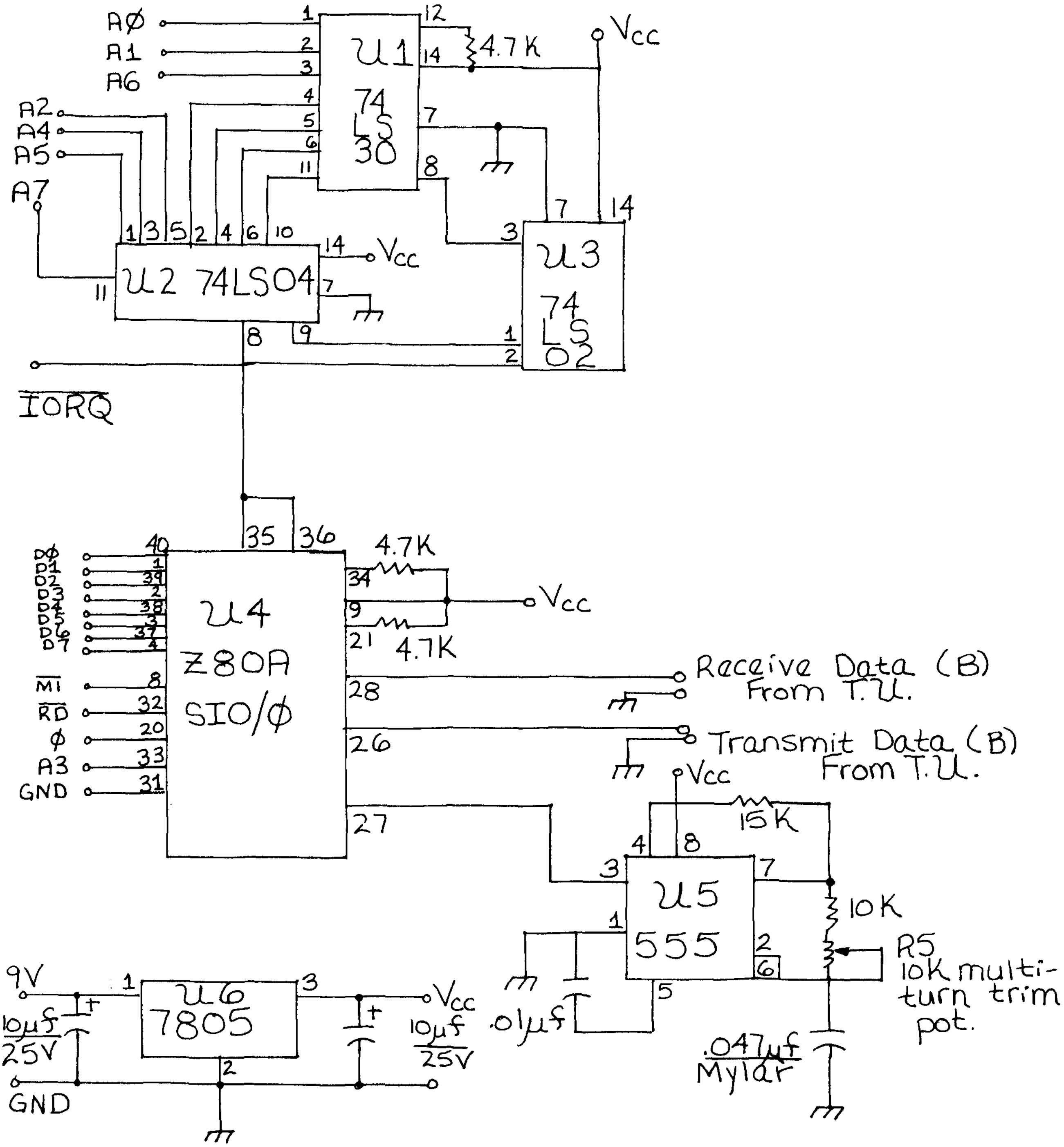
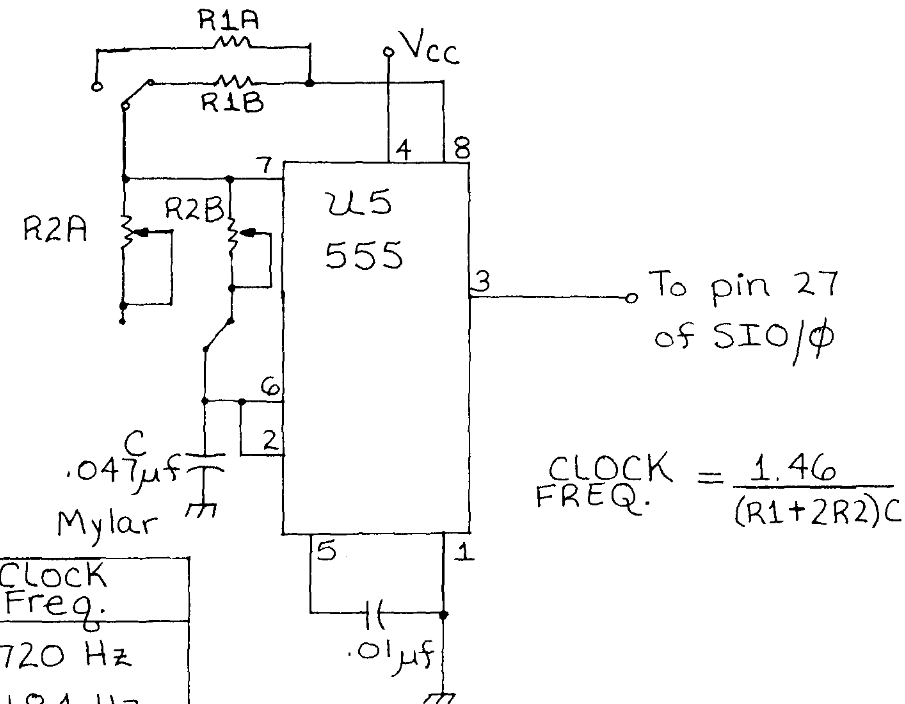


Fig. 1 — Timer circuit values are for 45-baud (60 WPM) operation.



R1	R2 (multi-turn)	Baud Rate	Clock Freq.
15K	20K	45	720 Hz
15K	10K	74	1184 Hz
6.2K	10K	110	1760 Hz
2.2K	5K	300	4800 Hz

Fig. 2A -- The clock circuit for selecting two different baud rates. This circuit may be expanded to provide for any number of clock frequencies.

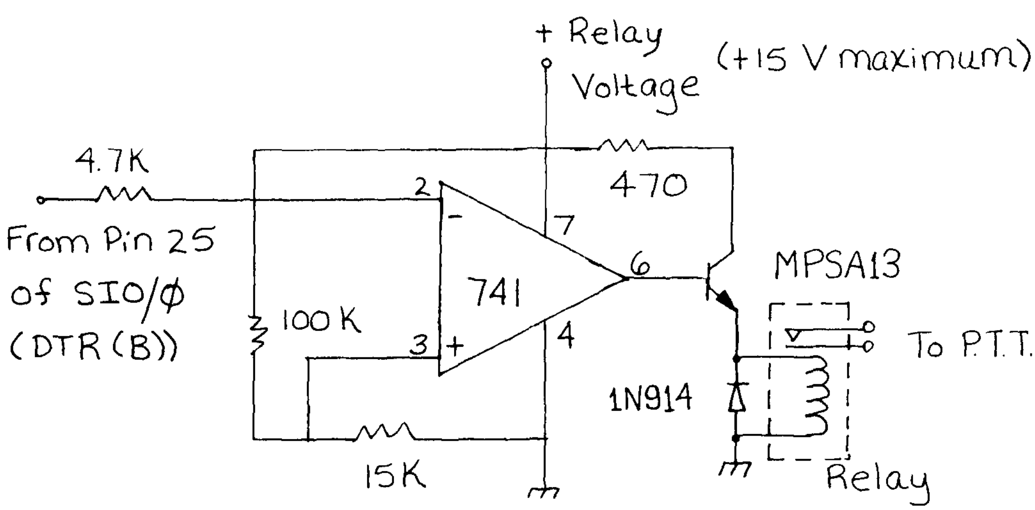


Fig. 2B — Circuit to provide software TR switching. Software required is listed on page 9. The 470-Ω resistor may need to be changed to ensure relay activation.

symmetric and after two years of use I have not found any frequency drift in the one I am using. The clock rate should be adjusted with R5 to run at 16 times the data rate (720 Hz for 60 WPM Baudot).

The Baudot Program

Space requirements will not allow an explanation of how the machine language programs work. I present only the details of how to use them, some customizing that can be done, and the program listings themselves. If you are interested in how it works, I have added comments to the listing to help you understand it. Let's first look at how the Baudot program is operated once it's on tape.

The program is loaded using the name "Baudot." After loading, it starts to run automatically in the receive mode. The upper ten lines of the display show the received text, printed on line ten and scrolled automatically. The software searches for and scrolls on a received space after 21 characters have been printed on the line. This almost completely eliminates half words at the end of the line. If no space is received, scrolling takes place after printing 32 characters on the line. Line 11 is blank and separates the received text from the transmit buffer text.

The next ten lines display the contents of the transmit buffer. As you type on the keyboard, the typed text appears across the bottom line and scrolls up as the line is filled. When the buffer is full, text is not accepted from the keyboard or from the string load routine. Once nine full lines have been typed, the buffer cannot be written to until the whole top line has been sent. As each character is transmitted, its appearance in the buffer is changed from black to white to white on black. This allows you to keep track of what is being sent and how much room is available in the buffer.

To change from receive to transmit, press shift-S (as if to type the keyword LPRINT). You may continue to type into the buffer, space allowing, while characters are being sent. Pressing shift-D (SLOW) will cause transmission of the buffer to cease and return to the receive mode. When the transmit mode is again selected, transmission from the buffer will resume where it left off. There is no way to delete characters from the buffer to prevent transmission of erroneous characters except to re-start the program.

Once you have a copy of the Baudot listing, take a look at page 3. Addresses 16640 through 16703 contain the TS-1000/Baudot conversion table. The code column contains the TS-1000 print code for a character. The op column shows the corresponding Baudot character sent or received (encoded in the last five bits of the address). The comment column indicates if the character in Baudot is not contained in TS-1000 and what to type as an equivalent. Some of the equivalents are inverse characters. If you include the TR switching option, you will not be able to send these characters because the software necessary for switching will replace the software for getting into and out of "inverse." Inverse is entered and exited by pressing shift-ENTER (FUNCTION). For example, to send the bell signal, type shift-ENTER, shift-B, and shift-ENTER. When the bell signal is received, a normal asterisk is printed.

The automatic sending of the figs and ltrs signals has not been included so you must be sure to type < before any series of numbers or punctuation and > afterwards. To send the text

SO 73 NICK. KL7KA DE KA1VW SK

type

SO <73> NICK <.> KL<7>KA DE KA<1>VW SK.

Additional Features

There are three additional features to point out. The first situation occurs when you are in the receive mode; the figs signal is sometimes received, but for various reasons the following ltrs signal is not received. In this case the system can get hung up printing all sorts of unnecessary numbers and punctuation. Pressing ENTER at any time while in the receive mode sets the system to ltrs. In fact, you can hold the key down and prevent the system from printing any figs. If you do this, any numbers received are printed as their letter equivalent (5 will be T, 7 will be U, ? will be B, and so on).

The carriage return and line feed signals are automatically sent after each set of 63 characters. If you include these signals in the transmitted text, they will **not** force a reset of the counter and will be included in the count. Changing the count requirement or defeating this function is explained in the next section.

As mentioned before, there is a special string, A$, that can be defined explicitly (without a line number). Once defined, it can be loaded into the transmit buffer by pressing shift-A (STOP). If you did not define A$ before saving the program, you may stop the program by typing shift-A. This will cause the error message 2/10, indicating that the string in line 10 of the BASIC program is undefined. If you have already defined A$ and want to change it, type shift-A and then press the space (BREAK) key. **IMPORTANT:** Whenever you stop the program you must type POKE 16595,205 before you do anything else.

Once you've stopped the program and have completed the necessary pokeing, define A$ explicitly. One example is LET A$=" CQ CQ CQ CQ CQ DE KA<1>VW ". A$ may be up to 120 characters long in a 2k RAM system. Type GOTO 2 to re-enter the program. One additional comment, by pressing shift-A you will load A$ into the transmit buffer and change the mode to transmit regardless of what mode you were in. (A special note to users who select the software TR option: Pressing shift-A does everything I have described above, but it will not activate the TR relay. The only way to activate the relay is to press shift-S or shift-D.)

System Operation

Before you type in the machine language code, decide how you want the system to work with respect to the following adjustable conditions. Do you want to use your Timex printer that is limited for use at 45 baud only? If not, you will want to delete the calls to the printer routine that cause irritating screen flicker. This is done by typing in Ø's (NOP - no operation) for the code at the following addresses

16542	16543	16544	
16609	16610	16611	
16863	16864	16865	16866
16899	16900	16901	
17037	17038	17039	

As a reminder that you are not going to use the printer, (NOP) will appear at these locations in the comment column.

Do you want to change the automatic carriage return line feed transmission operation? If so, address 17086 contains one more than the number of characters sent between carriage return and line feed signals. If you want to defeat the automatic operation, load code 201 at location 17087. The carriage return or line feed signal is now sent only when included in the typed text. This is very handy for message formatting.

When the line feed signal is received, do you want the display system to ingore it, print a space, or scroll the line? This is controlled by the codes shown at 16843.

The feature that checks for a space after printing 21 characters on the received line may be changed. You may prefer to print a full 32 character line and not care about parts of words at the end of a line. The value that controls when the search begins is located at 16874. If you want a full 32 character line, put 32 into this location.

There is one other area of customizing you may want to experiment with once you've got the system running. There are two timing loops for controlling the rate at which the keyboard is scanned for characters. If you are a fast typist, you may want to lower these values to prevent missing characters. If you are slow at removing your finger from the keys you may need to increase the values to prevent unwanted multiple letters. This means that characters are auto-repeating if you hold the key down. The locations of the time constants are 16578 (receive) and 16712 (transmit). The values included in the listing are for typing at about 45 wpm (comfortable for the author).

The machine language routine is 620 bytes long. It is written to be located starting at address 16514, the first byte in a remark statement that is the first line of a BASIC program. Therefore, you will need to make a REM statement, line one, that has 620 characters in it. I have seen several schemes for making long REM statements but the only reliable method I know of is to type it in directly. This is a long process (in FAST mode), so once you have it, put it on tape. The ASCII program needs 630 bytes so you may want to use it again, and there is always the chance that the system will crash while typing in the code. Once you've got the REM statement, type in the following BASIC program.

```
FAST
1 REM LLL1111111---620 characters
2 FOR N=16514 TO 17131
3 SCROLL
4 PRINT N,
5 INPUT I
6 POKE N,I
7 PRINT PEEK N
8 NEXT N
```

Run this program. As each address appears at the bottom of the display, type in the corresponding code from the Baudot listing. If you make an error, you can stop by pressing shift-A (STOP), explicitly assigning the address of the error to N, then type GOTO 3. The correct code can be re-entered from where the error occured and not from the beginning.

After all of the code has been entered, change the following lines.

```
4 PRINT N, PEEK N
5 NEXT N
6 STOP
SLOW
```

Run it again. As each address and its code appear, check it against the listing. If an error is discovered note the address. After checking, poke any addresses with the correct code as necessary.

Once you are sure the code is correct save what you have temporarily. Then enter the following changes.

```
FAST
2 FOR N=Ø TO 21
3 PRINT AT N,31;" "
4 NEXT N
5 SLOW
6 LET A=USR 16514
7 LET A=USR 16517
8 FAST
9 POKE 16595,201
10 FOR N=1 TO LEN A$
11 POKE 17118, CODE A$(N)
12 LET A=USR 17111
13 NEXT N
14 POKE 16595,205
15 SLOW
16 LET A=USR 16592
17 GOTO 8
18 SAVE "BAUDOT"
19 GOTO 2
```

Turn on your tape recorder. When you're ready to save the program, type GOTO 18. The program saves itself and comes out of the save routine running in receive. If you don't have the serial interface attached, it may crash or get hung in a loop. This is not critical because you've got the program on tape and this should not happen with the interface attached.

If you want to save a particular A$ with the program, type it in without a line number, but before you type GOTO 18. The A$ will be saved on tape with the rest of the program. Remember that the maximum length of A$ is 120 characters in a 2k byte RAM system.

If you now have more than 2k RAM and you want to have a longer string that's fine. Be sure, however, there is enough room in the buffer to load the whole string thought. I suggest that you keep it under 256 characters. Another way is to define more than one string. You can define 25 strings, B$ through Z$, and call them up by number using the BASIC routine that follows. Use the same lines for lines 2 through 8 as shown above.

```
9 GOTO 12
10 LET A$=B$
11 GOTO 21
12 INPUT Z
13 GOTO Z*10
18 SAVE "BAUDOT"
19 GOTO 2
20 LET A$=C$
21 POKE 16595,201
22 FOR N=1 TO LEN A$
23 POKE 17118, CODE A$(N)
24 LET A=USR 17111
25 NEXT N
26 POKE 16595,205
27 SLOW
28 LET A=USR 16592
29 GOTO 8
30 LET A$=D$
31 GOTO 21
40 LET A$=E$
41 GOTO 21
50 LET A$=F$
51 GOTO 21  ...and so on.
```

This continues for as many strings as you have room for up to Z$. You can define various strings explicitly for RY's, CQ's, brag, ID, or whatever. When you press shift-A an inverse L prompt appears at the bottom of the screen. Enter the number corresponding to the string number that you want to send (1 sends B$, 2 sends C$, 3 sends D$, you get the idea). Be careful that there is enough room in the transmit buffer to hold all of the string.

The ASCII Program

Since the operation of the ASCII software is similar to that of the Baudot software, I will only describe the significant differences. The major difference is that no provision is made to include the use of the printer. The printer control routine written in the BASIC ROM will not operate fast enough to print over 60 WPM. The summary of key functions for the Baudot program apply to the ASCII program as well except that pressing the ENTER key has no effect. This is so because there is no equivalent to figs and ltrs in ASCII. It is tested for in the machine language program and can be used at some time for future features.

The character code conversion table is on pages three and four of the ASCII listing. The Timex character code 136 is used for characters not printed on reception nor capable of being transmitted. These are mostly control characters. If you want or need to include any of these ASCII characters, you will have to swap codes with another character, thereby losing the ability to send that character. Note that graphics characters and keywords codes cannot be used and that lower case ASCII characters are printed as inverse characters.

The characters are sent as seven bits plus parity set to mark and two stop bits. This is compatible with almost all Amateur Radio ASCII communications. No parity checking is done on the received code and only the least significant seven bits are used for character conversion.

The customizing locations are as follows. Automatic carriage return and line feed transmission options are at 17098 and 17099, the line feed interpret options are at 16857, the character count for the scroll on space routine is at 16896, and the time constants for the keyboard scan are at 16566 and 16625.

The machine language routine will require a 630 byte remark statement for line one of the BASIC program. This is ten bytes longer than the Baudot program required. The single string (A$) BASIC program is

```
1 REM 111111--- 630 bytes --- 1111
2 FOR N=Ø TO 21
3 PRINT AT N,31;" "
4 NEXT N
5 SLOW
6 LET A=USR 16514
7 LET A=USR 16517
8 FAST
9 POKE 16583,201
10 POKE 16625,Ø
11 FOR N=1 TO LEN a$
12 POKE 17130, CODE A$(N)
13 LET A=USR 17123
14 NEXT N
15 POKE 16625,3
16 POKE 16583,205
17 SLOW
18 LET A=USR 16580
19 GOTO 8
20 SAVE "ASCII"
21 GOTO 2
```

Very Important: Whenever you stop the program be sure to POKE 16583,205 and POKE 16625,3 before restarting the program.

Summary of Key Functions

ENTER causes received characters to print in ltrs mode.

shift-ENTER causes entry to or exit from printing inverse characters in the transmit buffer.

shift-A causes the predefined string to load into the transmit buffer.

shift-S changes to transmit buffer.

shift-D changes to receive.

To stop program run press shift-A and then press the space (BREAK) key while the string is loading.

References

[1] Listings for both the ASCII and Baudot operation of this program are available for $1 and a business-sized s.a.s.e. Write to QEX, Technical Dept., 225 Main St., Newington, CT 06111.

[2] "Z80 SIO Technical Manual," SGS-ATES Semiconductor Corporation, 240 Bear Hill Rd., Waltham, MA 02154

[3] Carr, Joseph J.,"Z80 Users Manual," Reston Publishing Company, Reston, VA

[4] Zebra Systems, Inc., 78-06 Jamaica Ave., Woodhaven, NY 11421

Code Changes for TR Switching. Note that this eliminates the "inverse" mode.

```
     ELE   CODE   OP (Comment)

                  BAUDOT LISTING

  16 538    32    JR NZ 21
     539    21
     540    62    LD A, 118
     541   118
     542   205    CALL (2129)   (NOP)
     543    81                  (NOP)
     544     8                  (NOP)
     545    62    LD A, 5
     546     5
     547   211    OUT (75), A
     548    75
     549    62    LD A, 136
     550   136
     551   211    OUT (75), A
     552    75
     553    24    JR 37
     554    37
  16 604    32    JR NZ 22
     605    22
     606    62    LD A, 118
     607   118
     608   205    CALL (2129)   (NOP)
     609    81                  (NOP)
     610     8                  (NOP)
     611    62    LD A, 5
     612     5
     613   211    OUT (75), A
     614    75
     615    62    LD A, 8
     616     8
     617   211    OUT (75), A
     618    75
     619    24    JR 212
     620   212

                  ASCII LISTING

  16 533    32    JR NZ 14
     534    14
     535    62    LD A, 5
     536     5
     537   211    OUT (75), A
     538    75
     539    62    LD A, 232
     540   232
     541   211    OUT (75), A
     542    75
     543    24    JR 35
     544    35
  16 592    32    JR NZ 14
     593    14
     594    62    LD A, 5
     595     5
     596   211    OUT (75), A
     597    75
     598    62    LD A, 104
     599   104
     600   211    OUT (75), A
     601    75
     602    24    JR 217
     603   217
```

TABLE 1: Z80A SIO/0 PIN FUNCTIONS

PIN	FUNCTION	PIN	FUNCTION
1.	DATA 1	21.	RESET
2.	DATA 3	22.	DATA CARRIER DETECT (B)
3.	DATA 5	23.	CLEAR TO SEND (B)
4.	DATA 7	24.	REQUEST TO SEND (B)
5.	INTERRUPT CPU	25.	DATA TERMINAL READY (B)
6.	INT. ENABLE IN (DAISY CHAIN)	26.	TRANSMIT DATA (B)
7.	INT. ENABLE OUT (DAISY CHAIN)	27.	TRANSMIT & RECEIVE DATA CLOCK (B)
8.	$\overline{M1}$ FROM CPU	28.	RECEIVE DATA (B)
9.	V_{cc} (+5 VDC)	29.	EXT. SYNCH (B)
10.	WAIT/READY (A)	30.	WAIT/READY (B)
11.	EXT. SYNCH (A)	31.	GROUND
12.	RECEIVE DATA (A)	32.	$\overline{RD}$ FROM CPU
13.	RECEIVE DATA CLOCK (A)	33.	COMMAND/DATA SELECT
14.	TRANSMIT DATA CLOCK (A)	35.	CHIP ENABLE
15.	TRANSMIT DATA (A)	36.	$\overline{IORQ}$ FROM CPU
16.	DATA TERMINAL READY (A)	37.	DATA 6
17.	REQUEST TO SEND (A)	38.	DATA 4
18.	CLEAR TO SEND (A)	39.	DATA 2
19.	DATA CARRIER DETECT (A)	40.	DATA 0
20.	CPU CLOCK		

/1 version: 25 is TRANSMIT DATA (B); 26 is TRANSMIT DATA CLOCK (B); 27 is RECEIVE DATA CLOCK (B); DATA TERMINAL READY (B) is not available.

/2 version: 27 is TRANSMIT DATA CLOCK (B); 28 is RECEIVE DATA CLOCK (B); 29 is RECEIVE DATA (B); EXT. SYNCH (B) is not available.

/9 version: single channel only (A); pins 22 through 30 must be left unconnected.

Land Mobile Communications Engineering Review

By Ladimer S. Nagurney,* WA3EEC

The IEEE Vehicular Technology Society has produced a book for the IEEE Press that consists of reprinted papers in the area of VHF and UHF Mobile Communications. A total of 36 papers make up three general areas: Mobile Propagation, Mobile Data Communication, and Mobile Communications Above 800 MHz. They were selected to provide an overview of the field and to serve as a tutorial to introduce the relevant issues and considerations in Mobile communications design engineering.

The field of mobile communications, as represented by the Vehicular Technology Society, is closely related to Amateur Radio. Looking at the Society's history, a unique perspective on the beginnings of mobile communications exists, especially FM and the debate between SSB and FM in the VHF range. The appearance of papers on Vehicular Technology in IEEE Transactions that might be of interest to some amateurs are still published today.

The book is divided into three major parts, the first part written about Mobile Propagation. This section begins with Kenneth Bullington's classic, "Radio Propagation for Vehicular Communications." Although this paper has existed for 30 years, it represents a compendium on propagation experiences from 30 MHz to 3 GHz. Additional papers in this section offer experimental studies of propagation at 200, 450, 900, 1310, and 1430 MHz, as well as studies of 900 MHz.

The second part, Mobile Data Communications, contains six fairly recent reprints of papers about an experimental simulator for 1780-MHz Packet Radio and 450 MHz., 5 W, 1200-baud data channels. It provides a theoretical picture of mobile communications. Other papers model error rates in digital mobile systems.

The final section, Mobile Communications Above 800 MHz, discusses cellular radio. To the newcomer, a cellular system such as a new mobile telephone, breaks a region into small subregions called cells. Various frequencies are used in each cell and a mobile unit traveling from one cell to another will, unknowingly to the user, change the frequency without interruption to the conversation. Four papers from the Bell System Technical Journal that thoroughly describe the cellular system are included in the 13 papers presented here. The concluding paper is on the study of the bioeffects of an HT at 800 MHz on the human head. This research was conducted by Quirino Balzano, Oscar Gary, and Francis Steel of Motorola.

After referencing this book several times, I was pleased to own it, although, as with most reprint volumes, I already had many of the papers in my files. To the amateur with some training in Communications Theory, this book represents a good compendium. With access to 902 MHz and the spread of Packet Radio, we will witness many of the topics discussed. In this respect, I highly recommend the book. Amateur VHF and UHF communications is not, except for FM, primarily mobile. Many use directional antennas, high power, and narrow bandwidths. None of this is described here.

I would like to commend the Vehicular Technology Society and the editors, Dennis Bodson W4PWF, George McClure, and Samuel McConoughey for providing an interesting collection that will be useful for some time to come. I hope many amateurs will find it as helpful as I did.

This book (ordering number PC01685) is available through the Publication Sales Department at the IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854, tel. (201) 981-0060. The price is $49.95 for nonmembers, and $29.95 for members.

*73 Blackberry Lane, Amherst, MA 01002

Gateway — The ARRL Packet-Radio Newsletter

The premier issue of Gateway has just hit the newsstands, but many of you might ask what it is. This bi-weekly newsletter is exclusively written for the packet-radio enthusiast and interested parties. As Editor Jeff Ward, K8KA, states, "There are several technical packet-radio newsletters in publication and Gateway is meant to be a "news" newsletter, nothing technical."

If you are interested in gateways, packet meteor scatter, and the field of digital communication, you might be interested in subscribing. The following rates are available to ARRL members and nonmembers:

For 25 issues of Gateway:
- In the U.S.
  - ARRL Member $6.00
  - Nonmember $9.00
- In U.S., Canada, and Mexico by First Class Mail
  - ARRL Member $11.00
  - Nonmember $14.00
- Elsewhere by Airmail
  - ARRL Member $14.00
  - Nonmember $17.00

Prices are subject to change without notice. Remittance must be in U.S. funds, and checks must be drawn on a bank in the U.S.

TIPS FOR USING RIBBON CABLE

*Brian Weller ZS2AB

When working with the very popular "ribbon cable" used so extensively in logic and computer equipment, there are a number of things which can easily happen to make OM Murphy jump for joy. I have suffered at this man's hands all too frequently recently and have devised a couple of little tricks to wipe the grin off his face. Try these next time you use the cable.

The cable has a very tough but relatively thin outer insulation, and the wire size is not very large either, these two combining to produce broken strands when the wire is stripped – if one is not very careful. I have found an excellent way to overcome this. When you are preparing to strip the cable for soldering, draw a line with a pen across the cable at whatever distance back from the end that you wish to strip it, lay a ruler or other straight edge across the cable, and with a very sharp, fairly rigid blade, make a cut across the insulation. Do this gently, so as not to nick the strands inside. Turn the cable over and do the same on the other side. You will find that only a gentle pull with your sidecutters will cleanly remove the insulation from the wires. When doing this cut, use a blade with a fairly rigid body. The so-called "carpet knife" blades are ideal. A razor blade is a bit thin and is not always easy to control.

When separating the wires of the ribbon cable prior to stripping, the blade trick can be very useful as well. Again, mark the cable by means of a line drawn across its width at whatever distance from the end you wish to start separating the cores. Place the cable flat on a hard surface and holding the blade at about 45 degrees to the horizontal, press the sharp point of the blade into the web between the first two strands at the line you have drawn. Hold the blade in this position and draw the cable slowly along the blade. Keep the cable parallel with the blade, and you will find that the two strands are separated perfectly without removing the plastic from either core. The process is merely repeated for each core.

When you need to connect a ribbon cable to a plug which is wider than the cable, or does not have all its contacts in a convenient layout, merely stripping and connecting the cable, leaving its ends all the same length, you will invariably find that the end cores are stretched into the outer pins, whilst the inner cores all bunch up causing an unsightly mess with other possible problems. This can be overcome by trimming the cable and before stripping. Here a little experimentation may be required depending on your particular socket or plug, but the principle remains the same. It consists of taking a pair of scissors (don't let the XYL find out!) and, starting at one edge of the cable, cut through the strands so that the end of the cable is curved in towards the centre and back out to its full length at the other side. This will result in the outer cores getting progressively shorter as the centre of the cable is approached and when the cable is made off, all or certainly most of the cores will end up looking as though each were cut to the right length. Actually cutting each one to the correct length as you are making off the cable can be a very tedious process which may result in several broken cores before you have finished.

One last tip which can save a lot of trouble. When soldering such cable to a plug, slip a small piece of systoflex over each core and push this down over the completed connection after soldering. This supports the joint very well and will overcome the problem of broken wires after a bit of movement of the cable.

*P.O. Box 10317 Linton Grange Port Elizabeth 6015.

RADIO ZS APRIL 1984

Mr. Weller's article originally appeared in Radio ZS, the Journal of the South African Radio League. Information on this slim, but complete journal can be obtained by writing: P. O. Box 3911, 8000 Cape Town, Republic of South Africa.

Magazine Bibliography RBBS-PC

Download Bibliographies via Remote BBS: Central Washington RBBS-PC (509) 697-7298 at 1200/300 baud. The weekday schedule is 9 P. M. - 10 A. M. eastern time, and 24-hour service on weekends. There is no charge for online service.

Computer articles are referenced from nine magazines: PC Magazine, PC World, PC Tech Journal, IBM PC Softalk, Infoworld, PC Week, Personal Computing, Online Today, Cap PC Magazine.

Amateur Radio articles are referenced from eight magazines: QST, CQ, Ham Radio Magazine, 73 Magazine, QEX, RTTY Journal, Popular Communications, Monitoring Times.

For those without modems, bibliography lists can be ordered directly. Please select the magazine type (computer, Amateur Radio, or SWL) and enclose $5 for each. The address is Central Washington RBBS-PC, P. O. Box 538, Selah, WA 98942.

New HF Antenna Design Program for the TRS-80C Color Computer and MC-10

Cynwyn's latest offering in the software series for Amateur Radio hobbyists is the HF Antenna Design. The program makes the necessary calculations for bulding three popular types of antennas -- the dipole, Yagi, and quad, for frequencies of 1.8-30 MHz, and displays them in an easy-to-read tabular format. Dimensions for the Yagi and quad are optimized for maximum gain.

HF Antenna Design requires a TRS-80C color computer with 16k RAM and extended color BASIC, or and MC-10 with 4k RAM. The program is available from Cynwyn, 4791 Broadway, Suite 2F, New York, NY 10034, telephone (212) 567-8493. It is in cassette form and the cost is $10 plus $2 for shipping and handling purposes.

Bits

OMNICOM Features Two Seminars on Networking Standards

X.25, the worldwide packet network access protocol, was introduced in 1976 by the International Telegraph and Telephone Consultative Committee (CCITT). Since that time, this pervasive implementation for user interfaces with packet-switched data networks has "come of age" with a more comprehensive, expanded specification.

Provisions have been included that greatly facilitate **internetworking** between Public Data Networks and private networks, including the popular Local Area Networks (LANs). X.25 has also been fully adjusted to comply with the Network Service Definition requirements of the new **Open Systems Interconnection** (OSI) architecture.

The new draft Recommendation X.32 defines the procedures for access between users and packet networks via the Public Switched Telephone Network and Circuit-Switched Public Data Networks. The specification has now reached an advanced stage of development and will be finalized in early 1985.

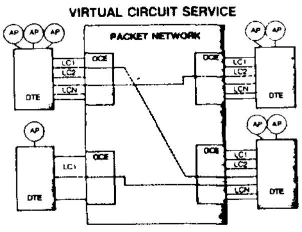
The packet assembly/disassembly (PAD) facility enables simple start-stop terminals to utilize X.25 packet networks. The PAD specifications are also extensively used in statistical multiplexing applications. X.3, X.28, and X.29 have now been enhanced and refined in the new 1984 versions.

The above information will be part of the seminar outline as well as topics on Concepts and Terminology, Physical Layer, Data Link Layer, Network Layer, Additional Terminal Considerations, Transport Layer, Transport Service Scenario, X.32 Dial-up Access, Internetworking, Packet Assembly/-Disassembly, and an Open Discussion. Held in Orlando, Florida on September 26-28, 1984, and in Houston, Texas on November 28-30, 1984, call (703) 281-1135 for Karen, or write Omnicom, Inc., 501 Church St., N.E., Suite 206, Vienna, VA 22180 for further information. There is a registration fee of $750 for each 3-day seminar.

Local Area Networks are Standardized

Local Area Networks (LANs) are key elements for data transmission in distributed information systems (interconnecting terminals, host and personal computers, or gateways to other networks).

VIRTUAL CIRCUIT SERVICE



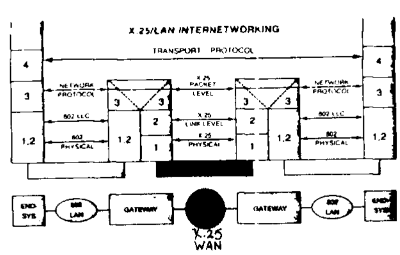
They serve applications in geographically limited areas and provide highspeed, low-delay transfer of data.

A vast quantity of vendor designs have entered the marketplace during the last few years. The diversity of designs and lack of standards have made it difficult for users to realize consistency and economics in implementing their system requirements. As a result of this confusion, the IEEE organized the 802 standards committee in 1980 to develop Local Area Network specifications.

The IEEE 802 committee divided into six subgroups to deal with the various elements of operation. 802.1 deals with architectural and internetworking aspects. 802.2 covers the Logical Link Control (LLC) protocol. This is common to various types of media implementations. 802.3, 802.4, and 802.5 are each responsible for the media access and physical elements of Carrier Sense Multiple Access with Collision Detection (CSMA/CD) baseband, Token Bus broadband, and Token Ring baseband, respectively. Finally, 802.6, the last to be established and slowest to gain momentum, deals with the Metropolitan Area Networks (MANs) for a larger, but still limited, area of geographical coverage.

In close liaison with the IEEE work, the European Computer Manufacturers Association (ECMA) has adopted a compatible family of LAN standards. In September of 1983, the International Organization for Standardization (ISO) agreed to adopt the IEEE 802 LAN standards as International Standards. The family of 802 specifications will be published as subsections in a single standard designated ISO 8802 when finally approved by the ISO member bodies.

On October 1-3, 1984 in Orlando, Florida, and on December 3-5, 1984 in Houston, Texas, you can learn about the role of LANs through discussions on Concepts and Terminology, LAN Technology, Logical Link Control (LLC), Medium Access Control, Physical Layer Standards, Internetworking and Transport Layer Functions, Vendor Status, Future IEEE 802 Work, and Other LAN Standards. An Open Discussion will conclude the seminar at the end of each day for review of the presentations, or to discuss other related issues. Registration information for this seminar is the same as noted above.



QEX: The ARRL Experimenter's Exchange is published by the

American Radio Relay League
225 Main Street
Newington, CT 06111 USA
telephone 203-666-1541

Larry E. Price, W4RA
President

David Sumner, K1ZZ
General Manager

Paul L. Rinaldo, W4RI
Editor

Maureen Thompson, KA1DYZ
Assistant Editor

David W. Borden, K8MMO (Data Communications)
Geoffrey H. Krauss, WA2GFP (VHF+ Technology)
Associate Editors

Debbie Chapor
QEX Circulation Manager

The purposes of QEX are to:

1) provide a medium for the exchange of ideas and information between Amateur Radio experimenters,

2) document advanced technical work in the Amateur Radio field, and

3) support efforts to advance the state of the Amateur Radio art.

Subscriptions are available to ARRL members and non-members at the rates shown on the QEX Subscription Order Card inside this issue.

All correspondence concerning QEX should be addressed to the American Radio Relay League, 225 Main Street, Newington, CT USA 06111. Envelopes containing manuscripts and correspondence for publication in QEX should be marked: Editor, QEX. QEX subscription orders, changes of address, and reports of missing or damaged copies may be marked: QEX Circulation. Members are asked to include their membership control number or a label from their QST wrapper when applying.

Both theoretical and practical technical articles are welcomed. Manuscripts should be typed and double spaced. Please use the standard ARRL abbreviations found in the January 1984 edition of QST. Authors should supply their own artwork using black ink on white paper. When essential to the article, photographs may be included. Photos should be glossy, black-and-white positive prints of good definition and contrast, and should be the same size or larger than the size it will be when printed in QEX.

Any opinions expressed in QEX are those of the authors, not necessarily those of the editor or the League. While we attempt to ensure that all articles are technically valid, authors are expected to defend their own material. Products mentioned in QEX are included for your information, not advertising, nor is any endorsement implied. The information is believed to be correct, but readers are cautioned to verify availability of the product before sending money to the vendor. Material may be excerpted from QEX without prior permission provided that the original contributor is credited, and QEX is identified as the source.

QEX: The ARRL
Experimenters' Exchange
American Radio Relay League
225 Main Street
Newington, CT USA 06111

Nonprofit Organization
U.S. Postage
PAID
Hartford, Conn.
Permit No. 2929