resent an independent statement, and we have seen how groups of bits can constitute a positional number system, similar to our familiar decimal system, but using only two symbols and doubling or halving column values. Living as we do amidst so many signs, signals, and codes, we can easily see how anything can be represented by a group of bits. One particular bit pattern is assigned to each item to be represented. For example, The American Standard Code for Information Interchange (ASCII) uses a six-bit subgroup to represent all letters, the numerals zero through nine, and a number of punctuation marks and symbols. The assignments are shown in Table III.

To translate a text to ASCII, each letter, figure, or symbol in the text would be replaced by the six bits assigned to that letter, figure, or symbol by ASCII. The title of Hugo Gernsback's prophetic novel, *RALPH 124C 41 +* becomes: 110010 100001 101100 110000 101000 000000 010001 010010 010100 100011 000000 010100 010001 001011 in ASCII.

Now it's time to sharpen up our vocabulary. We have defined the bit as a single binary digit. And we have shown how a group of bits can represent a letter, a number, or a symbol. A group of bits used to represent a letter, a number, or a symbol is called a *character*. This is logical. Character is our term for a single letter, number, or symbol. Thus it is reasonable to call the associated group of bits a character. Most digital systems are built to handle characters of some specific size. Often a number of characters are handled together. Such a group of characters are called a *word*. But do not confuse this with the English term word. Digital systems group characters for convenience of handling. They may not be related in any way. They need not make an English word or anything like it.

The number of different numerals used in a number system is called the *base* or the **radix** (Latin for root) of the system. Thus our familiar decimal system is base (radix) ten, while the binary system is base (radix) two. Computer people often find base eight, called *octal*, very convenient for some purposes. When systems other than base ten are used, the base must be clearly indicated. This is done by stating the base in the text or through subscripts. Thus 27_8 indicates that the number 27 is in base eight (octal). This could be translated to 23_{10}, 23 in base ten. (27_8 means 2 eights plus 7 units = 16 + 7 = 23.)

The number of different characters which can be formed with N bits is 2^N. Thus, for example, 6 bits can form 2^6 (64) different characters. These can be used to count from 0 (000000) to 63 (111111), or to represent any group of 64 symbols or statements.

But sometimes we have good reason not to use all the available characters. For example, noise, dirt, or some momentary malfunction will occasionally cause a solitary bit to be transferred or interpreted incorrectly. It might seem that this would cause the character containing that bit to be changed. However, by making it a rule to use only some of the

TABLE IV

DECIMAL	BINARY CODED DECIMAL (BCD)
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	0001 0000
11	0001 0001
12	0001 0010
13	0001 0011
14	0001 0100
15	0001 0101
16	0001 0110
17	0001 0111
18	0001 1000
19	0001 1001
20	0010 0000
30	0011 0000
40	0100 0000
50	0101 0000
60	0110 0000
70	0111 0000
80	1000 0000
90	1001 0000
100	0001 0000 0000
125	0001 0010 0101

Decimal coded into Binary Coded Decimal form. The four-bit combinations 1010, 1011, 1100, 1101, 1110, and 1111 are never used.

possible characters while making the others forbidden combinations, we can detect and even correct such errors.

Another example is of prime importance in the instrument field. When we buy a digital voltmeter, we expect the data readout to be in decimal form. After all, not many of us sight-read binary. $+1101.001_2$ may be the same as $+13.125_{10}$ volts, but we expect to see the easily read decimal form. If the data inside the voltmeter were generated and stored in binary form, the translation problem would be formidable. To see the problem, inspect Table 1, showing the weights of the binary digits. In general, each bit contributes to many if not all decimal columns. The translator would require a fantastic amount of logic.

But by modifying the binary code, we can substantially reduce the problem. A group of four bits has 16 possible characters. Suppose we take such a group and assign ten of these characters to the numerals zero through nine, as in Table IV, and never use the other six. These four bits now represent one decimal digit. We will use another four bits to represent the next decimal digit and so on. Now to translate from this code, which is called binary-coded-decimal (BCD), only four bits need be inspected for each decimal digit, and the translator problem becomes manageable.

We began with a comparison that revealed one of the key advantages of digital techniques: High accuracy at modest cost. Binary coding can represent any collection of data, numbers, letters, statements, et cetera, by using an appropriately large group of bits, each bit being realized by one basic two-state logic circuit. Further, we have seen the flexibility: how code groups (characters) can be left unused for error detection or for simplification in translating to decimal form.

But these advantages are only part of the story. Digital readout gives us fast, accurate data. Completely inexperienced personnel can read and copy a digital display. Compare this with reading the dial of a typical mirror-scale, multi-range VOM. Data can be entered on a numeric keyboard or numeric switches as opposed to multiturn potentiometer dials. Yet even these advantages are only a small part of the story. Unlike an analog system, a digital system maintains full accuracy until its maximum frequency limit is reached. Thus, while most analog systems begin to degrade above 10,000 hertz, digital systems hold full accuracy to 1, 10, or even 100 megahertz, depending on the specific circuits used. And even at 100 megahertz, the power dissipated is low.

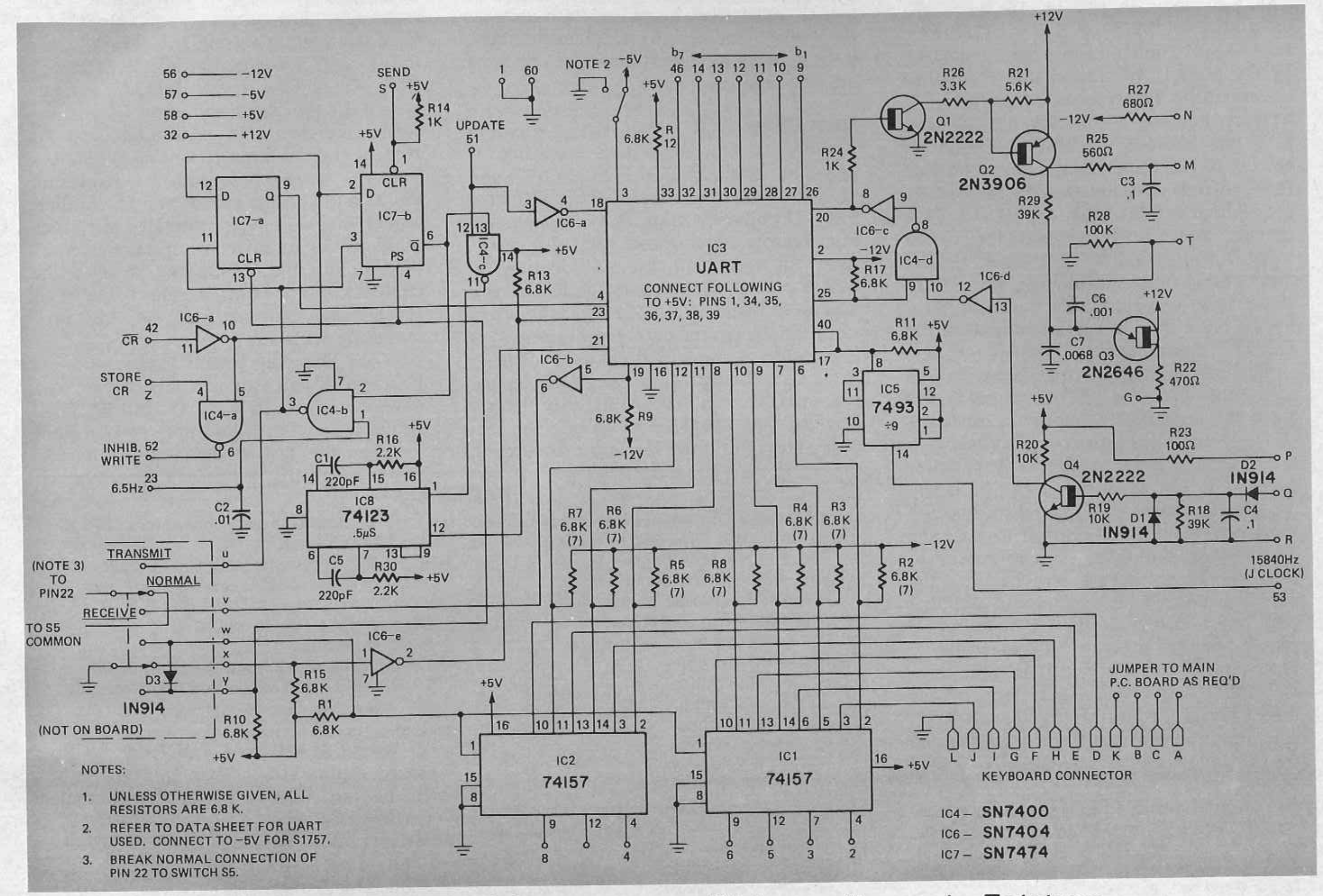
The magic combination of reasonable power, modest cost, high accuracy, and high speed is an open door. This combination allows us to perform many hundreds of operations without loss of accuracy, or to time-share one assembly of equipment by switching it between a number of sources and outputs. Messages can be combined, transmitted, and split apart. Frequencies can be synthesized to unbelievable accuracies. Messages can be coded for security. Digital operations can be repeated to perform digital filtering with results almost impossible to achieve with ordinary filters.

Suddenly techniques that were difficult or impossible in analog systems are open to us. Our next article will explore the rules of logic that govern these digital techniques. **R-E**

ANSWERS

A. 8 + 4 + 1 = 13
B. 16 + 8 + 2 + 1 = 27
C. 16 + 1 = 17
D. 128 + 64 + 32 + 4 + 2 + 1 = 231
E. 4 + 2 + ½ + ⅛ = 6⅝
F. 101, 111, 1000, 1111, 1010001

add this UART to your TV typewriter



You can connect it between your TV typewriter and a Teletype for a "hard print" copy. Or you can connect it between a cassette recorder and the Minicomputer for extended memory.

by ROGER L. SMITH

IF YOU OWN AN ASCII ENCODED KEYBOARD, Teletype®, 'TV Typewriter', **(Radio-Electronics,** Sept. 1973), or a Mark 8 Minicomputer, **(Radio-Electronics,** July 1974) you need a UART (Universal Asynchronous Receiver/Transmitter) to communicate with other such equipment or to record data on a tape. This article shows you how to add a UART and support circuitry to your 'TV Typewriter' so you can send your data to a Teletype for "hard copy" print out. The same circuit will also output data into or accept data from a regular cassette tape recorder. It is also possible to transmit data from Teletype to recorder and back (could be used to automatically type out mailing lists). When a cassette recorder is used with the Mark 8 Minicomputer, the result is a large bulk storage device.

Although it is not included on the PC board, we'll show you the circuits needed for a FSK (Frequency Shift Keyed) MODEM using the XR-210. The "Mark 8 Minicomputer", "TV Typewriter", MODEM, and DAA (Data Access Arrangement) together give you a powerful computer terminal.

How is all of this interfacing done? The communication problem boils down to this: characters in the form of 6 to 8-bit ASCII logic levels must be sent one bit at a time to a remote location and thereupon converted back to 6 or 8 parallel data bits. All of this requires an elaborate combination of AND gates, shift registers, start and stop bit generators and detectors, and timing circuits. Thanks to MOS-LSI circuits, most of this has already been done for you at a terrific saving of time, space, and money in the UART.

The size of a UART, a 40 pin package, may startle you, but the price is $15 or so, which can't be equalled by hardwired circuits. Several companies presently make UARTS including Western Digital, American Micro-Systems, General Instrument, Texas Instr., Signetics, and Intel (see UART Table). Detailed operation of a UART can be obtained from any of these manufacturers and won't be covered here. Basically, however, as each ASCII character is presented to the transmitter input, start, stop, and parity bits are added and the bits are sent out in serial fashion at a rate determined by an external oscillator. The receiver portion reverses the process for incoming serial data.

The transmitter output of the UART is a TTL logic signal that should be converted to a 20-mA current-loop signal for Teletype operation or to the standard EIA RS-232C interface signals. You could also use your own driver design, provided you are communicating to your own gear.

The main idea is to provide an error-free signal to the receiver with moderate rise and fall times (30 volts/μs or slower).

With the UART and additional circuits shown in the schematic, the output (Q2, pins **M** & **N** and input (Q4, pins **P** & **Q** are directly compatable with a Teletype. The oscillator input (pins 17 & 40) was adjusted to be Teletype compatable. The frequency should be 16 times the baud rate (110HZ × 16 = 1760HZ). STOP, START, and IDLE bits are generated by the UART. We found that communicating with a Teletype was made easier if we increased the TV Typewriter memory to 7 bits because the Teletype must receive CR (carriage return) and LF (line feed) signals. Adding the 7th bit is done by adding a 2524 memory IC to your memory board as described in the section titled "Changes to TV Typewriter." If you don't intend to use a Teletype, you should study only those circuits that suit your needs. Note that the serial output of the UART is normally high.

The rest of the circuit shown in the schematic provides the SEND signal to start the transmitter and data selectors (74157) to select either the keyboard or the UART to input to the TV Typewriter memories. The SEND signal can be a key on your keyboard or a special pushbutton that provides a ground signal to start the transmitter. We found it convenient to Stop the transmissions by detecting the carriage return (CR) signal and stopping any time it occurred (the CR is found on pin 42—see "changes to TV Typewriter"). Note that the entire circuit is built on a PC board that will plug on top of the board "stack" of your "TV Typewriter". The Transmit-Normal-Receive switch can be any DP3T switch conveniently located on the unit.

You will need 4 lines to connect the output of the board to the Teletype. Connect as follows:

I/O Board to Teletype

M	7	Refer to the section titled
N	6	"Teletype Information" for
P	3	proper connections on the
Q	4	Teletype.

If you plan to use your UART with only a keyboard to transmit data, you will need an oscillator on pins 17 & 40. You can change IC5 to an MC4024 and alter the wiring to provide 1760 HZ. Since you will be transmitting only 1 bit at a time, all you need is a one-shot for your KP signal (use IC8 connected to pin 51). Received data will appear in parallel form at pins 5 thru 12 of the UART which you connect to your particular output device. Note that when an output is available, pin 19 goes high and pin 18 must then be pulsed low to reset the output register. Also, when no input data is being received, input pin 20 must be held high.

Recording data

Most cassette tape data recording is done by converting the bits ("O" and "1") to two tone signals. This is called FSK (Frequency Shift Keying) recording. Unfortunately, the need for error free FSK recording and reproduction requires precision equipment which to you and me translates into $$$. Since the data rates we are using here are quite low, it is possible to use a $20 cassette recorder *if* we record single frequency tone *bursts*. Note that we can't record the bits directly because they occur at a 110 bits per second rate—too low for most recorders. With this tone burst method, the tape speed can vary considerably before any errors are introduced. This method can be called PPK for Pulse Position Keying and consists of recording a pulse of tone signal for a "1" and not pulse for an "0" during each 9.1-ms bit time.

On our Input/Output board, the tone bursts are generated by unijunction oscillator Q3. The tone signal (about 3.8 KHZ) is present during the "Mark" time ("1") and absent during the "Space" ("0") time. The output (pin **T**) is connected to the auxiliary input of the recorder (or thru 470K to the MIC. input). If you make a tape recording without a Teletype connected, you must ground pin **M** and connect pins **P** and **Q** together. Notice that we haven't provided any automatic START or STOP for the recorder. A dual monostable (74123) in the SEND line could be used (with a relay) but it is much easier to start and stop the tape manually with a switch.

When you play back the tape into the I/O board, you should connect the "earphone" output to pins **Q** and **R.** You could incorporate a switch arrangement to select the desired input/output device.

PARTS LIST

All resistors ¼ watt unless noted
R1, thru R13, R15, R17 — 6800 ohms
R14, R24 — 1000 ohms
R16, R30 — 2200 ohms
R18, R29 — 39,000 ohms
R19, R20 — 10,000 ohms
R21 — 5600 ohms
R22 — 470 ohms
R23 — 100 ohms
R25 — 560 ohms
R26 — 3300 ohms
R27 — 680 ohms
R28 — 100,000 ohms
C1, C5 — 220-pF disc
C2 — .01 μF
C3, C4 — .1 μF
C6 — .001 μF
C7 — .0068 μF
Q1, Q4 — 2N2222
Q2 — 2N3906
Q3 — 2N2646
D1 - D3 — 1N914
IC1, IC2 — SN74157
IC3 — UART (see table)
IC4 — SN7400
IC5 — SN7493
IC6 — SN7404
IC7 — SN7474
IC8 — SN74123
Misc.: P.C. board, molex connectors, 22Ga. wire, DP3T switch.
Add to TV Typewriter; Signetics 2524 IC, 2200-ohm and 6800-ohm resistors, 10-μF electrolytic (C9).
Printed-circuit board is available from Techniques Inc.
235 Jackson St., Englewood, N.J. 07631
$6.50 postpaid (New Jersey residents add 5% sales tax).

UART MANUFACTURERS

Western Digital Co., *(TR1602), (TR1402)* 19242 Red Hill Ave., Newport Beach, CA 92663

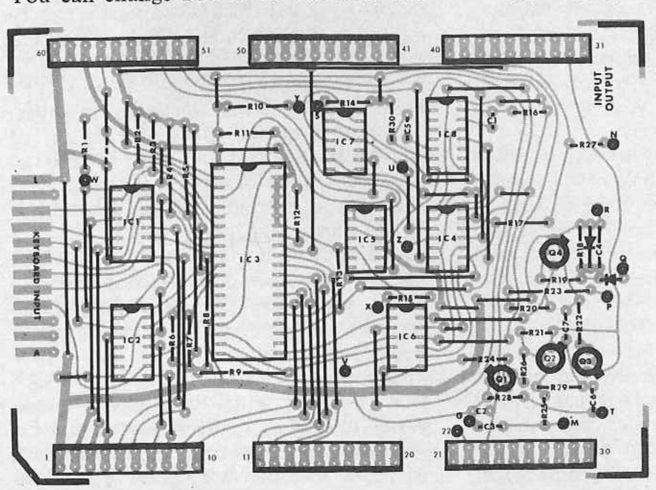
American Micro-Systems *(S1757; S1883* —the most available at this writing) 3800 Homestead Rd., Santa Clara, CA 95051

General Instrument *(AY-5-1013A)* Box 600, Hicksville, N.Y. 11802

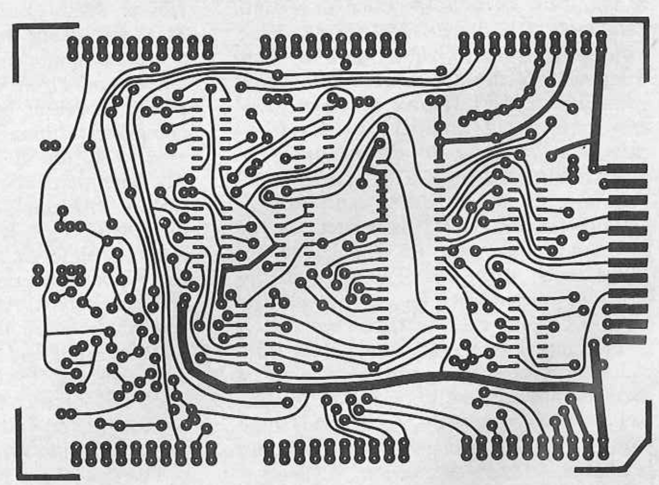
Texas Instruments *(TMS 6010NC; TMS 6012)* Box 5012, Dallas, Texas 75222

Signetics *(2536)* 811 E. Arques Ave., Sunnyvale, CA

Intel *(8201)* 3065 Bowers Ave., Santa Clara, CA



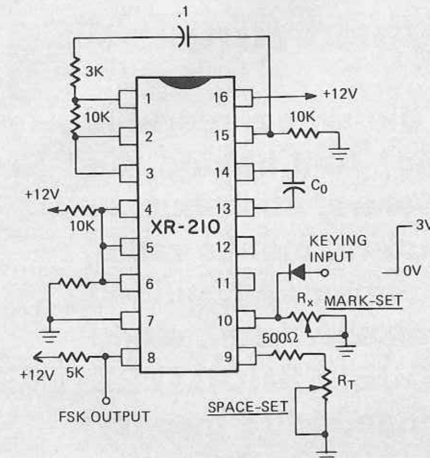
FOIL PATTERN FOR THE INPUT/OUTPUT BOARD shown half-size. The UART is a 40-pin dip package.



X-RAY VIEW OF THE FOIL PATTERN for the input/output board shows the component layout.

You may also want to use an 8 to 500-ohm transformer on the output of your recorder to help match it to the I/O board.

Although the Input/Output board was designed primarily with a Teletype and cassette recorder in mind, it represents a large portion of a MODEM. All that needs to be added are a couple of XR-210 IC's to provide modulation and demodulation of the serial data. For this application, you may want to increase the data rate to 150, 300, or 1200 baud. You would need to increase the oscillator input (pins 17 & 40) accordingly, to 16 times the baud rate.



Select C_0, R_T & R_X for desired frequencies.

$$C_0 = \frac{220}{f_0} \quad R_T = \frac{.1f_0}{f_1 - f_0} \quad R_X = \frac{.3fl}{f2 - f1}$$

where f_0 = 5% lower than fl, R_T & R_X in KΩ.
for fl = 1070 Hz, f2 = 1270 Hz:
Co = .22 μf, R_T = 2K, R_X = 1.6K

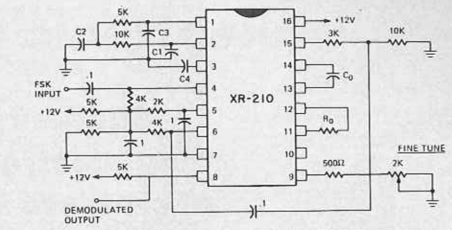
FIG. 1—SCHEMATIC DIAGRAM FOR FSK MODULATOR shows how the XR-210 IC is connected. This circuit plus FSK demodulator is needed for MODEM.

The XR-210 circuits necessary for a MODEM are shown in Figs. 1 and 2. You can write to EXAR Integrated Systems, 750 Palomar Ave., Sunnyvale, Ca. 94086 for their 8 page brochure on the XR-210 for complete details.

Changes to TV Typewriter

1. The "blinking" of the cursor is used as a KP signal to send out each character, so its speed should be increased to 6.5 Hz by changing C9 on the timing board to 10 μF. Also, change wiring of pin 22 as shown in the input/output schematic. Note that the 1760-Hz UART oscillator input limits your maximum character rate to 10 characters/second.

2. Storage of bit 7 must be added to



Operating Conditions	Typical Component Values
300 Baud	
low baud: f1 = 1070 Hz, f2 = 1270 Hz	R_0 = 5.1K, C_0 = .22 μf; Cl = C2 = .047 μf, C3 = .033 μf, C4 = 3 μf
high baud: f1 = 2025 Hz, f2 = 2225 Hz	R_0 = 8.2K, C_0 = .1 μf; Cl = C2 = C3 = .033 μf
1200 baud: f1 = 1200 Hz, f2 = 2200 Hz	R_0 = 2K, C_0 = .14 μf, Cl = .033 μf; C2 = .01 μf, C3 = .02 μf, C4 = .1 μf

FIG. 2 — FSK DEMODULATOR schematic shows how the XR-210 is connected. This circuit plus circuit in Fig. 1 is needed for MODEM.

give Teletype a carriage return & line feed. Do this by adding a new 2524 IC to the *memory board.* "Piggyback" the new 2524 on top of IC4 by first bending pins 2 and 6 straight out from the new IC. Pinch remaining pins together slightly (1 toward 8, 4 toward 5, etc.) so the IC will make physical contact with IC4's pins and only pins 4 & 8 will need soldering (just a *very* light touch.) Tack on a 2.2K resistor from pin 6 to common of RI thru R5 (+5V) and a 6.8K from pin 2 to bus on pin 4 (−5V). Add jumper wires from pin 6 of IC to pin 2 on pc board and another from pin 2 of IC to pin 46 of board.

3. The TV Typewriter must distinguish between a line feed (LF) and carriage return (CR.) This is done by detecting bits 3, 6 and 7 (CR = $\bar{A}_6 \cdot \bar{A}_7 \cdot \bar{A}_3$). On the cursor board, cut a break in foil from IC6-13 to R12 and add jumpers to connect IC-8, 9 and 10. Cut break in foil from IC6-10 to pin 52 and add jumper from IC6-10 to pin 42 (this allows you the choice of storing or not storing the CR).

4. On the main "TV Typewriter" board, remove the wiring between switch S2 and pin 32 and add wiring to bring +12 volts to pin 32.

Notes: 1. When you elect to store CR and you type a CR on the TV Typewriter, an M will appear on the screen. Typing a LF (line feed) will produce a J. This is because the character generator is designed for only 6 bits. The Teletype printout will be OK.

2. If you wish to use the "TV Typewriter" *without* the input/output board, jump pins 42 and 52.

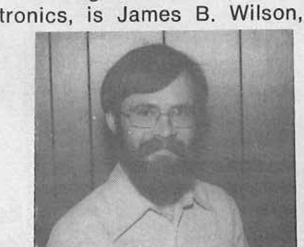
Teletype information

The input terminal strip of the Teletype must be connected to provide full duplex operation using the 20-mA loop signals. This is done by moving the blue wire on the current source resistor (the large wire-wound resistor) to the 1450-ohm position. Move the brown/yellow wire from terminal 3 to 5. Move the white/blue wire from terminal 4 to 5. Move the violet wire from terminal 8 to 9. Signal input will be to terminals 6 and 7 and the output at terminals 3 and 4.

It requires 9.1 ms for a Teletype to send or receive a bit. Each charcter transmission consists of a start bit, 8 data bits, a stop bit and an idle bit. The start bit is a "space" (no current), and the stop and idle bits are each a "mark" (current). **R-E**

Gernsback Award goes to Jim Wilson and Richard Mitchell

Another winner of the Hugo Gernsback Scholarship Award, a $125 check given annually to a student in each of eight leading home-study schools of electronics, is James B. Wilson, Jr., a



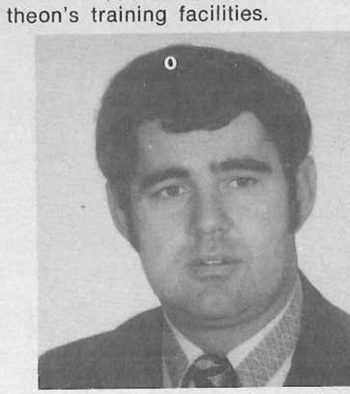
student in the Capitol Radio Engineering Institute (CREI). The Award is in memory of Hugo Gernsback, who throughout his life devoted much time and energy to encouraging young men in the study of radio and electronics.

Mr. Wilson, 24, a Nebraskan, has just moved to Lincoln with his wife Lorraine and three-month-old daughter Kyla, where he has been transferred to the Video Tape Duplication Center of the Nebraska Educational Network, as a recording engineer. Previously he was a transmitter engineer for the Network at KRNE-TV for six months, then was transferred to KLNE-TV, with remote of KHNE-TV, where he has been for the past year and a half.

During that time he completed CREI's Electronic Engineering Technology program, with a major in Television Engineering Technology, and gone on to additional majors. These were Communications Engineering, Automatic Control Engineering, and Computer Engineering technologies. Credits for these courses have been applied toward the New York Institute of Technology college credit program through CREI. He has now 47 semester hours at NYIT and is continuing his studies toward the Associate in Applied Science degree in electronic engineering technology.

Jim is an Associate Member of the Society of Motion Picture and Television Engineers (SMPTE) and of the IEEE. While at CREI he was also listed in *Who's Who Among Students in American Vocational and Technical Schools.*

The RCA WV-529-A "Service Special" VOM, given each month by RCA to the runner-up in the Scholarship Award contest, is won this month by Richard D. Mitchell, 35, of Deerfield, NJ. An ex-Marine, Mr. Mitchell began his study of electronics in the service, and later enrolled in the CREI Electronics Engineering Technology program. He has also taken courses in several other home-study schools, including Motorola, Cleveland, Jerrold, Grantham and Raytheon's training facilities.



Now confined to the house for several months following a serious auto accident, he is continuing his studies in his major of Missile and Spacecraft Guidance. Before the accident he was—and still is—employed with Control Data Corporation as a technical manager. He has held a First Class FCC license since 1959. **R-E**