SPECIAL CONSTRUCTION ISSUE—6 GREAT PROJECTS

$1.00 ■ JUNE 1977

Radio-Electronics

THE MAGAZINE FOR NEW IDEAS IN ELECTRONICS

A GERNSBACK PUBLICATION

3 octave
MUSIC GENERATOR
uses pink noise

multi feature
TELEPHONE DIALER
has a memory

no digit
DIGITAL CLOCK
for your wall

easy to build
DIGITAL IC TESTER
and identifier

PLUS:

- ★ Quad Scope Adapter
- ★ CB Noise Limiters
- ★ Solving The dB Mystery
- ★ Hi-Fi Lab Test Reports
  Sony Elcaset Deck
  Sherwood HP-2000 Amp
- ★ Jack Darr's Service Clinic
- ★ Build This 2650 Computer

4. If IC again comes up hot, there is an internal short–discard it.

Step 7. Observe the LED's. If only the LED connected to the V_{cc} pin is on, suspect an open circuit in the internal power wiring of the IC. Normally, unconnected input pins will float high and some output pins will be high, thus lighting some of the LED's. Proceed to the next several steps to confirm the open circuit before discarding the IC.

Step 8. Remove power from the tester.

Step 9. Wire 1's (+5) and 0's to the input pins (jumpers between SOURCE and WIRE) as required by the function and pinout of the IC under test.

Step 10. Wire LO5 from the SOURCE to any pins that are open collector outputs. This is a pull-up voltage which makes it possible for the LED to accurately indicate the outputs.

Step 11. Apply power.

Step 12. Observe the output pin LED's to determine whether or not they are behaving as expected.

Step 13. When testing such IC's as flip-flops, counters, registers, multivibrators and the like, it will be necessary to make and break a connection several times while observing the LED's. This is most conveniently done by just touching the wire to the proper pin on the POINT socket rather than inserting it into the WIRE socket.

Step 14. If the outputs do what the pinout (or data book) indicates they are supposed to do as you manipulate the inputs, the IC is good and can be wired into your project without fear that it will cause a problem.

That is about all there is to testing an IC. The simple ones such as gates, flip-flops and counters, can be checked out very quickly. The more complex IC's require more time, but even they are easy and a little experience will make checking them quick, too.

Identifying Unknown IC'S

Identifying an unknown IC can be a tricky business, especially if it is one of the more complex ones. The less complex are rather straightforward. The following procedure has been found to produce the best results:

Step 1. Insert IC into the TEST socket.

Step 2. Briefly touch each pin of the POINT socket with +5 volts (from the SOURCE socket) observing the LED's as you do so.

Step 3. If all or many LED's turn on as +5 volts is applied to *every* pin, discard the IC.

Step 4. If only the LED for the pin with +5 volts applied turns on *every* time, discard the IC.

Step 5. [14-pin DIP's] If pin 7 and pin 14 turn on many LED's and most other pins do not, pin 7 is GND (0V) and pin 14 is V_{cc} (+5V).

Step 6. [16-pin DIP's] If pins 8 and 16 turn on many LED's and most other pins do not, pin 8 is GND (0V) and pin 16 is V_{cc} (+5V).

Step 7. Most non-military IC's (TTL) will display the pin 7 and pin 14 or the pin 8 and pin 16 combination.

Step 8. If one of those combinations does not appear, note which pins turn on many LED's–one is GND and one is V_{cc}.

Step 8.1 Check your catalogs of IC pinouts; the odd combination itself may identify the IC.

Step 8.2 If not, make a guess–apply +5 to one and 0 to the other.

Step 8.21 If the IC gets warm to the touch, reverse the leads and try again (the IC may not be damaged).

Step 9. Apply power to the IC (V_{cc} and GND).

Step 9.1 Since floating inputs go high, the input LED's will be on.

Step 9.2 Some output pin LED's will be on.

Step 10. Briefly, apply HI0 (GND through 330 ohms) to each pin with a high logic-level.

Step 10.1 Each pin pulled low (LED turned off) by the applied HI0 may be labeled an input pin since few high outputs will be pulled low by this procedure.

Step 10.2 Those that are not pulled low *and* those that are low may be labeled output pins.

Step 11. Knowing V_{cc}, GND, INPUT and OUTPUT pins, match this information with the pinout diagrams in your catalogs and/or data books.

Step 11.1 If positive identification is made, test the IC and mark it with its proper number using a carbide scriber.

Step 11.2 If several possible identifications are made, test the IC for each possibility.

Step 11.3 If no identification can be made, further experimentation may reveal additional facts to make identification possible.

Step 11.31 Generally, flip-flops will change output states (toggle) when the T output is grounded and ungrounded.

Step 12. If an IC cannot be identified, put it aside to try again after you have gained some experience.

Step 13. Do not expect to identify them all; some are very complex little monsters–even some of the 14-pin DIP's!

Note that you will sometimes find an IC that is part good and part bad. For example, there may be only two or three good gates in a 7400 or a 7473 may have one good flip-flop and one bad one. Of course, you can throw them out but, if they don't get mixed up with your good ones, they can come in quite handy.

The solution is to mark a partly-bad IC so that you won't wire it into a circuit requiring a fully operational IC. Then, you can keep it until you run into a project that requires fewer functions than are to be found in one IC.

TTL and CMOS are the most popular IC families. Probably most of your work will be with these types. If so, consider wiring up for and building in a 5-volt supply. The TTL's use 5 volts and most CMOS devices will operate with the same voltage.

Now you have an instrument for testing and/or identifying many types of IC's. You will find it good practice to test *every* IC before wiring it into a circuit. Even premium quality IC's are sometimes bad and the testing can be done quickly. The Identifier/Tester will prevent a lot of grief on the workbench. **R-E**

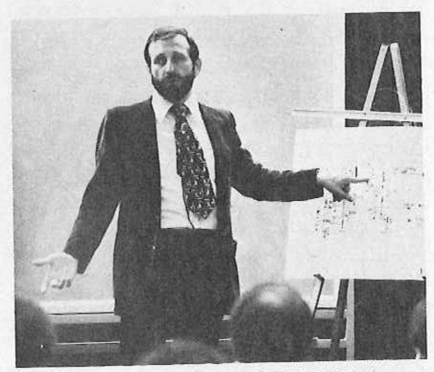
CB training workshops start in Indianapolis

More than 50 technicians attended the first of a series of Forest Belt Training Workshops, held in the Airport Holiday Inn, Indianapolis, during the last week of January.

The first three days of the five-day program were spent in exploring the basics of CB servicing–studying phase-locked loops, single sideband, modulators and demodulators, AGC, ANL and other CB fundamentals. The third day treated specific troubleshooting.

The fourth and fifth days were devoted to studies in preparation for the FCC Second Class Radiotelephone license.

A number of awards were given out at the banquet that concluded the program. NESDA (*National Electronic Service Dealers Association*) awarded three gift memberships. High point of the banquet was the Hickok Prize, which consisted of the right to choose between a Hickok



AUTHOR-INSTRUCTOR FOREST BELT, former editor of Radio-Electronics, explains a point in his trademarked *Easi-Way Servicing*.

model 388 In-Line frequency counter and power/VSWR/modulation meter, or to put the ticket into a grand-prize drawing at the last of the series of fifteen 1977 workshops. The winner of that 15-person drawing will receive an entire Hickok COMM-Line six-instrument service center.

Further workshops are planned at Atlanta, Baltimore, St. Paul, Boston, Chicago, again at Indianapolis and probably one at Toronto. Enrollment fee for a five-day workshop is $280. For further information, write for Brochure 24 to: Forest Belt's Training Workshops, Box 68120, Indianapolis, IN 46268. **R-E**

Build 2650-Based Microcomputer System

Part III. Built on a single printed-circuit board, this 2650 microcomputer contains a video and cassette tape interface and resident supervisor program. Add a keyboard, video monitor, cassette tape recorder and power supply for a complete working system

JEFF ROLOFF

THE FIRST TWO PARTS OF THIS ARTICLE appeared in the April and May issues and provided the construction details and an in-depth look at how the circuit works.

This month, the article concludes with a look at the software associated with the 2650 microcomputer and a look at how it's programmed.

Using the supervisor

In all, there are nine basic functions of the supervisor program. First, you can alter or display any position of memory. (You cannot, obviously, alter data that is in ROM.) This will allow you to enter and inspect your own programs in the RAM. After entering and checking your program, you can use the supervisor to jump to your program and execute it. When your program returns control to the supervisor (by a branch instruction), it saves the contents of the CPU registers so that you can inspect them. You can also set the CPU registers before you jump to your program. When the program is finished and you want to turn off the microcomputer, the program can be transferred to cassette tape in blocks of two-256-bytes and then transferred back to the microcomputer at a later time. There is also a command to turn on the tape recorder so that you can manually rewind it, etc. To troubleshoot the program, a breakpoint (a point in the program where processing will be interrupted) can be set. When this address is reached, a message is written on the screen and the CPU registers or any memory location can be inspected to see what they were immediately before the breakpoint address. You can also clear this address if you wish to change it. Another command permits verification of what is on tape against any block of memory.

The specific instructions for the operation of the supervisor are provided. In the examples, all underlined characters are ones entered by the operator. Everything else is printed on the screen by the supervisor program. A period (.) indicates that the supervisor program is ready for a command. An A indicates that it is waiting for you to type in an address. At any time the supervisor is looking for a keyboard input, you can press ES (escape) which will terminate the present command and wait for a new one.

To alter or display memory, depress the A on the keyboard. It will then ask for an address, which should be entered in hexadecimal form. The address and the data then appear on the next line of the video monitor. You can now do one of three things: depress the ES key to quit the alter/display routine, enter C to change the data at that location, or depress the space-bar to display the next memory location. If you decide to alter the memory, the supervisor will wait for you to type in two hex characters to fill the memory location. The following is an example of this routine:

.A A100A	
100A 05	data is 05, space indicates go on
100B 10C3B	data is 10, change to 3B
100C 38ES	data is 38, press escape to terminate routine

To execute a program, type an E for the command. The supervisor will then ask for the address that it should start executing at. It will then jump to the address and start executing instructions:

.E A163B execute at 163B, press space to start

If the program returns to the supervisor (by a branch instruction), all of the CPU registers are saved, and then it asks for a new command.

If you did return from your program by a branch instruction, or because of a breakpoint, you can inspect the memory using the alter routine, or you can inspect the CPU registers entering I. It will then ask you to type in a register number corresponding to the register that you want, as follows:

Enter	For
0	Register 0
1	Register 1, Bank 0

2 Register 2, Bank 0
3 Register 3, Bank 0
4 Register 1, Bank 1
5 Register 2, Bank 1
6 Register 3, Bank 1
7 Program Status Word, Lower
8 Program Status Word, Upper

The microcomputer will then display the data that was in this register right before the program returned to the supervisor. Similar to the alter/display routine, you now have three options: to stop by depressing the ES key, to change the register value by entering C, or to inspect another register by depressing the space-bar:

.I
R3 2CC 02 register 3, bank 0 has 2C, change to a 02
R4 C3_ space to go on
R8 B7ES escape to quit

To transfer your program to tape, enter a D. The supervisor will then ask for the beginning address and the length (in bytes–up to 256) of the data to be transferred. Remember that everything must be entered in hexadecimal for the supervisor to interpret it correctly. The supervisor actually dumps one-byte more than the length that is entered, so that a length of FF (255 in decimal) will cause a dump of 256 bytes. Also, a length of zero indicates that this is the last block that the load routine should read in, and will cause any load of this data to be completed. This allows the load routine to load multiple blocks without having to re-enter the L (load command) and allows it to stop itself automatically when all the data has been loaded. Therefore, a block with length of zero should be inserted after all of your data blocks have been transferred to the cassette tape:

.D A10DB LFF dump 256 bytes start at 10DB
.D A11DB L10 dump the next 17 bytes
.D A0000 L00 dump an end of file block

After all of the data has been transferred, the supervisor will automatically ask for a new command.

If you wish to check the data that has been transferred to the cassette tape, use the verify (V) command. After entering V, the supervisor will then ask for an address. After this has been entered, the supervisor will start the tape recorder and will look for a block starting with this address. When the block is found, the data in the block is compared with the actual data in memory at the time of the verify. If the data is not the same as what is on the tape, an error occurs. Also, if the first block on the tape has an address different than the one that you typed in you will get an error message.

It should be noted that the dump routine transfers the data along with the address and the length of the block:

.V A1000 be sure that the first block on tape is for address 1000, and that the data is correct

The verify routine returns to ask for a new command if the verify was all right.

When using the cassette tape routines, the supervisor takes care of turning the recorder off and on. To implement this feature, you must hook the auxiliary control wires of the tape recorder to a relay, and drive this relay with the TD ON line from the board. You must be sure to have the recorder in the correct mode (i.e., record or play).

To load data from a tape, simply enter L for the command and be sure the recorder is in play mode. All of the data is recorded on the tape along with the address to load it at and the length of the load. The supervisor will ask for a new command when it is done loading the tape:

.L load from tape

Recorded on tape are sumcheck characters also. Their purpose is to check against errors while recording or playing back data. The first sumcheck is sent after the address and length, while the second is sent after the block of data. Therefore, you can receive an error indication while loading or verifying in either of two places.

To set a breakpoint address in your program, enter a B as the command. It will then ask you for the address of the breakpoint:

.B A1703 set breakpoint address to be 1703

When this address is reached in the program, the supervisor will save all of the registers and wait for a new command. It signifies that the breakpoint address has been reached by writing the message:

BP 1703 indicates breakpoint address was reached

The registers and memory can now be examined as you see fit. After the breakpoint has been executed, it is cleared and the program will be allowed to run past the point next time through.

If you decide that a breakpoint that you set was at the wrong address, you must clear the breakpoint address by entering C. If you do not do this, the program will still have a supervisor inserted instruction and will not operate correctly:

.C 1703

The supervisor responds by typing the address that the breakpoint was set at. Note that you must set breakpoints in an address position where an instruction would begin. In other words, you cannot set a breakpoint to be executed at an address which is the second or third byte of an instruction.

To run the tape recorder (to rewind the tape, etc.) enter R. Pressing escape will return you to the supervisor.

Subroutines

The supervisor program includes many useful subroutines that can be used by branching to them. The more useful ones are shown in Table I.

All registers used are in the bank currently selected.

More information about the 2650 microprocessor and its language can be found in the *2650 Microprocessor Manual,* which is available from Signetics.

TV typewriter

Now that your system is finished and you know how to use the supervisor program, what can you do with it? One obvious use is for a TV typewriter display, which is also quite simple to do. Table 2 has the listing for the TV-typewriter program that accepts any printable character along with the backspace code and carriage return. The first thing that the program does is branch to the keyboard input routine (KBIN) with a branch-to-subroutine instruction (BSTA). This subroutine receives one character from the keyboard and prints it on the display, if it is not a control character. After the character has been printed (if it is printable), the subroutine returns to the program to check if it was a backspace or a carriage return. If it was either of these two, the result of the respective compare instruction will be to clear the condition code. Then the branch instructions immediately following the compare instructions check the condition code to see if the compare was equal. If it was equal, the program branches to the correct subroutine. The carriage-return subroutine is simply a branch to the line feed-carriage return (LFCR) subroutine in the monitor, while the backspace routine is contained in the TV typewriter program.

The backspace routine simply takes the cursor pointer and decrements it. It also writes a space at the present cursor position and writes a new cursor at the new position.

The RAM positions 17FE and 17FF are used to store the present address of the cursor. To store a character in the cursor position, indirect addressing is used. This causes the processor to read what is in 17FE and 17FF and use this data as the actual address where it should do the operation.

Tape format

The cassette tape routines take care of all data encoding and decoding needed to interface with your tape unit, but if data is to be transferred between two different types of machines, you must have the format of the tape. That format is as follows:

Character	Description
1	colon indicating the start of a block
2	high order address byte for load
3	low order address byte for load
4	length of data block
5	sumcheck character for bytes 1-4
6 to n–1	data
n	sumcheck for data

Character 4 is the length of the data block. If it is zero, it represents the fact that this is the last block and that the load routine can stop. If it is from 1 to 255 (H01 to HFF), it is one less than the length of the data field. This allows transferring data blocks of exactly 256 bytes.

All characters are 8 bits wide, with one start and two stop bits. The least significant bit is recorded first, with the other bits following in order.

The sumcheck is generated by feeding each data byte into an EXCLUSIVE-OR gate with the sumcheck character and then rotating the resulting byte to the left one bit. The sumcheck is cleared before data is started. When read back, each byte (including the sumcheck) goes through this routine. If no errors have occurred, the ending sumcheck character should be zero. Each block has two sumchecks and they are totally independent of one another.

Loading a program

After you have written a program, how do you load it? To many people who have been around microcomputers, the answer is obvious: use the alter routine that the supervisor provides. To people who are having their first computer experience, this solution may not be so clear.

Recall that the alter routine allows you to change the data contained in any RAM memory location. Thus, by using this routine to change all of the memory locations that your program needs, you can enter your program into the system.

A question comes up immediately: At

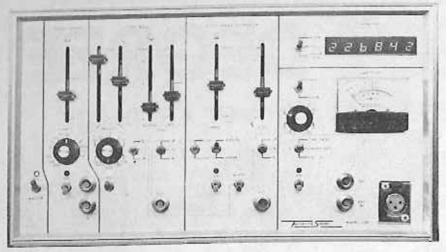
continued on page 84

TABLE I

Address	Mnemonic	Description
0396	WCHR	writes the character that is in R3 on the screen and updates the cursor position.
0024	LFCR	moves the cursor to the leftmost position of the next line.
030F	KBIN	inputs one ASCII character from the keyboard and puts in R3.
006A	HXOT	takes the binary data in R2 and displays it as two hex characters.
01B6	INHX	inputs two hex characters and converts them to binary in R3.
0083	RETU	branch to this address to return to the supervisor and save the register values.

TABLE II—TV TYPEWRITER PROGRAM

Line	Address	Instruction	Label	Operation	Operand	Comments
1	0000		LFCR	EQU	0024	ADDRESS OF LINE-FEED ROUTINE
2	0000		KBIN	EQU	0309	ADDRESS OF KEY-BOARD INPUT ROUTINE
3	0000			ORG	1600	START AT ADDRESS 1600 IN HEX
4	1600	75 08	TVT	CPSL	08	SET OPERATIONS WITHOUT CARRY/BORROW
5	1602	3F 03 09		BSTA,3	KBIN	GET KEYBOARD INPUT NOTE THAT KBIN ALSO WRITES THE CHAR
6	1605	E7 08		COMI,R3	08	COMPARE THE CHARACTER TO A BACKSPACE
7	1607	18 07		BCTR,0	BACK	IF A BACKSPACE, DO BS ROUTINE
8	1609	E7 0D		COMI,R3	0D	COMPARE THE CHARACTER TO A RETURN
9	160B	3C 00 24		BSTA,0	LFCR	IF A RETURN, DO CARRIAGE RETURN ROUTINE
10	160E	1B 70		BCTR,3	TVT	JUMP BACK TO BEGINNING—GET NEW CHAR
11	1610	07 20	BACK	LODI,R3	20	ASCII FOR A SPACE
12	1612	CF 97 FE		STRA,R3	I17FE	STORE THE SPACE AT THE CURSOR LOCATION
13	1615	0F 17 FF		LODA,R3	17FF	LOAD THE LOW ORDER CURSOR ADDR INTO R3
14	1618	A7 10		SUBI,R3	10	SUBTRACT ONE CHAR POSITION FROM IT
15	161A	CF 17 FF		STRA,R3	17FF	STORE THE NEW CHARACTER
16	161D	77 08		PPSL	08	OPERATIONS NOW WITH CARRY/BORROW
17	161F	0F 17 FE		LODA,R3	17FE	HIGH ORDER CURSOR ADDRESS
18	1622	A7 00		SUBI,R3	00	SUBTRACT BORROW FROM PREVIOUS SUBRACT
19	1624	CF 17 FE		STRA,R3	17FE	STORE THE NEW HIGH ORDER ADDR
20	1627	07 5C		LODI,R3	5C	CODE FOR THE CURSOR
21	1629	CF 97 FE		STRA,R3	I17FE	STORE THIS IN THE NEW CURSOR POSITION
22	162C	1B 52		BCTR,3	TVT	JUMP BACK—DO NEXT CHARACTER
23	162E			END		



BUILD A COMPUTER

continued from page 49

which memory location should I start the program? The answer is that your program can be put in any RAM area, so you can start your program anywhere from address H1510 to H17E9. This is the usable RAM space that the board provides you. If you expand the RAM you could, of course, put your programs in that space also.

For an example, suppose you have the following program and want to load it into the system:

```
07 00      LODI,R3    00
06 10      LODI,R2    10
1F 00 00   BCTA,UN    0000
```

If you choose to load this program at the start of RAM, then you would alter location 1510 first. The supervisor displays the contents of 1510, and permits you to change it. Once it is changed, the next byte is displayed (1511) and you are again allowed to change it. This process continues until the entire program is loaded.

TABLE III

```
A A1510
1510 00C07
1511 00C00
1512 00C06
1513 00C10
1514 00C1E      ***note that
                this line has a
                mistake on it
1515 00e
A A1514
1514 1EC1F
1515 00C00
1516 00C00
1517 00e        ***done!
```

(e) represents the pressing of the escape key.

If you should make a mistake, simply press escape, and alter the location with the mistake and continue on. The way the screen would look for program would appear on the video monitor is shown in Table 3. **R-E**

NO DIGIT DIGITAL CLOCK

continued from page 37

clock, both switches are in the RUN position with the switch bats down. If you should overshoot the correct time when setting, let the hand sweep around again.

Construction

Although the actual circuit is simple, the wiring can get complex. Multiplexing to the 72 LED's necessitates the use of a double-sided printed circuit board. The foil patterns for the PC board are shown in Figs. 2 and 3. If the board is square, the clock can be mounted by the corners in a square enclosure or if cut round, it can be mounted by a single screw in the center of the round case.

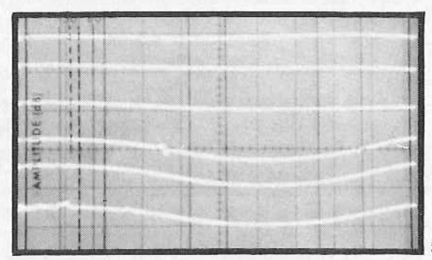
The LED's and driver transistors are mounted on the face side as shown in Fig. 4 and the balance of circuitry mounts on the rear as shown in Fig. 5. Care should be taken when mounting the LED's to insure that they are of equal height and are aligned to give an even display.

The clock can be mounted in a number of different cases. The one shown here is a clear plastic tube with a clear front. The hour positions are indicated by white plastic squares glued to the front. The old fashioned octagonal wall clock cases can also be used. This makes for an interesting combination of old craftsmanship and modern technology. **R-E**

HI-FI LAB TESTS

continued from page 64

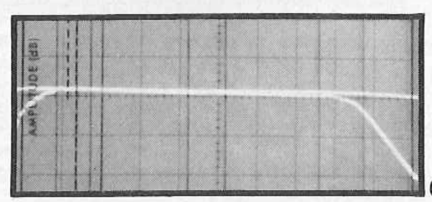
moderate compensation at progressively lower DB LEVEL control settings and again, examining the –40-dB line, we now see a bass boost of only around 6 dB at 50 Hz for this setting. It should be noted that this variable



5

loudness compensation applies only to the bass end, while the moderate amount of treble boost incorporated in the loudness circuitry remains constant regardless of the CONTOUR control position.

Figure 6 illustrates the steep and effective action of the low-cut and high-cut filters, both of which have 12 dB-per-octave slopes with the –3-dB cutoff points falling exactly as specified by Sherwood.



6

Our overall product analysis, together with our summary comments concerning features and listenability of the *model HP-2000* will be found in Table II. Even on the basis of superficial price/performance ratios, the Sherwood *model HP-2000* is a winner in every sense. But, aside from good clean power, the *model HP-2000* offers a degree of flexibility and control that rivals that of many preamplifier/basic-amplifier two-component systems costing considerably more. **R-E**

market center

CLASSIFIED COMMERCIAL RATE (for firms or individuals offering commercial products or services). **$1.40 per word (no charge for zip code) . . .** minimum 15 words.

NONCOMMERCIAL RATE (for individuals who want to buy or sell personal items) **85¢ per word . . .** no minimum.

ONLY FIRST WORD AND NAME set in bold caps. Additional bold face (not available as all caps) at 10¢ per word. Payment must accompany all ads except those placed by accredited advertising agencies. 5% discount for 6 issues, 10% for 12 issues within one year, if paid in advance. All copy subject to publisher's approval. Advertisements using P.O. Box address will not be accepted until advertiser supplies publisher with permanent address and phone number. Copy to be in our hands on the 26th of the third month preceding the date of the issue (i.e., August issue closes May 26). When normal closing date falls on Saturday, Sunday or a holiday, issue closes on preceding working day.

BUSINESS OPPORTUNITIES

TV picture tube rebuilding operation, includes 4 position automatic oven, only one going in area $12,000. Building available. Details: **2333 Compton,** St. Louis, MO 63104

HIGHLY PROFITABLE **ONE-MAN ELECTRONIC FACTORY**
Investment unnecessary, knowledge not required, sales handled by professionals. Ideal home business. Write today for facts! **Postcard will do. Barta-BR, Box 248, Walnut Creek, CA 94597.**

CASH with your camera! Earn $5000-$15,000 yearly selling to 4,000 customers. Sell photos fifteen or more times. Guaranteed. Details—13¢ stamp. **PICTUREPROFITS,** 26 Josephine Blvd., Suite RE677, Shoreham, NY 11786

WANTED

QUICK cash . . . for electronic equipment, components, unused tubes. Send list now! **BARRY,** 512 Broadway, New York, NY 10012, 212 WAlker 5-7000

EDUCATION & INSTRUCTION

TELEPHONE bugged? Don't be Watergated! Countermeasures brochure $1.00. **NEGEYE LABORATORIES,** Box 547-RE, Pennsboro, WV 26415

GRANTHAM's FCC License Study Guide—377 pages, 1465 questions with answers/discussions—covering third, second, first radiotelephone examinations. $13.45 postpaid. **GSE PUBLICATIONS,** 2000 Stoner, Los Angeles, CA 90025

F.C.C. EXAM MANUAL
PASS FCC EXAMS! Memorize, study—Tests-Answers for FCC 1st and 2nd class Radiotelephone licenses. Newly revised multiple choice questions and diagrams cover all areas tested in FCC exams, plus Self-Study Ability Test. $9.95 postpaid. Money-back guarantee.
Tests-Answers for FCC First and Second Class Commercial License
COMMAND PRODUCTIONS RADIO ENGINEERING DIV. P.O. BOX 26348 E SAN FRANCISCO, CAL. 94126

CB'ers—Be a "Ham"—We'll teach you! Life membership: $4.00—**AMERICAN RADIO COUNCIL;** Box #1171-F; Garland, TX 75041

HIGHLY effective degree program in electronics engineering. Advance rapidly! Our 31st year. Free literature. **COOK'S INSTITUTE,** Box 20345, Jackson, MS 39209

CB antenna construction manual. Assemble beams, quads, groundplanes, roll-up-tennas from common hardware. Save 80%. Non-technical instructions with pictorials, easy assembly, excellent performance, complete $3.00—TENNA-FARM, 1117 Dewitt Terrace, Linden, NJ 07036

BURGLAR fire alarm experts needed for cars, homes, industry. Learn high-profit systems installation at home spare time. Simple, quick, complete. Free information by mail. No salesmen. **SECURITY SYSTEMS MANAGEMENT SCHOOL** (homestudy), Dept. 7339-067, Little Falls, NJ 07424

PLANS & KITS

BI-LINEAR amplifier, 3-30MHz, 100 watt mobile. Construction plans, $3.00. **IGOR,** Box 5516-FF Walnut Creek, CA 94596

U-BUILD IT COMPUTER SYSTEM $599

A MPU, CRT TERMINAL AND AUDIO CASSETTE AT A ROCK BOTTOM PRICE.
- SC/MP MICROCOMPUTER: 256x8 RAM, 512x8 ROM
- GREEN PHOSPHOR VIDEO MONITOR: 12" RASTER SCAN 24 LINES x 80 CHAR.
- SOUTHWEST TECH 1024-32 CH x 16 LNS,ASCII KEYBD.
- SOUTHWEST TECH AC-30 AUDIO CASSETTE INTERFACE

SEND $1 FOR OUR CATALOG
DESCRIBES COMPLETE LINE OF KITS & UNITS BOOK REVIEWS, LIST OF NEW & SURPLUS PARTS

COMPUTER WAREHOUSE STORE DEPT R, PO BOX 68, KENMORE STATION BOSTON, MA 02215 671/261-2701

KITS, semiconductors, components. Free flyer. **CHIPS ELECTRONICS,** Box 1030, Oakville, Ont., Canada L6J5E9. U.S. inquiries.

BREATHE better air with negative ion generator. Kit $165.00/Details, $1.00. **GOLDEN ENTERPRISES,** Box 1282RE, Glendale, AZ 85311

Wurlitzer reproductions

ORGAN KITS KEYBOARDS
THE ULTIMATE IN DESIGN AND SOUND
DEMO RECORD AND BROCHURE $1.00
DEVTRONIX ORGAN PRODUCTS, Dept.4B
5872 Amapola Dr. • San Jose, CA 95129

SAVE 50%. Build your own speaker systems. Write: **MCGEE RADIO ELECTRONICS,** 1901 McGee Street, Kansas City, MO 64108