Subscriptions: $15 / 6 Issues

Issue #8

# CYBERTEK

The Cyberpunk Technical Journal

Published by OCL/Magnitude, P.O. Box 64, Brewster, NY 10509

## The RS-232 Data Tap

by Thomas Icom

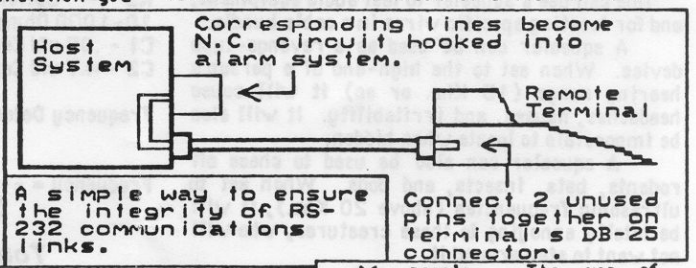
When this device is hooked up to an RS-232 line between the DTE and DCE, it will send all data on that line to a third terminal, serial printer, or RS-232 equipped computer. While it does have the disadvantage of having to be physically connected to the system. It's just the thing for yanking User IDs and passwords off of time-sharing systems at schools. It can be hooked up to an XR-2206 FSK modulator chip, which can be fed into the microphone input of a radio transmitter, where the data can then be received up from a remote location.

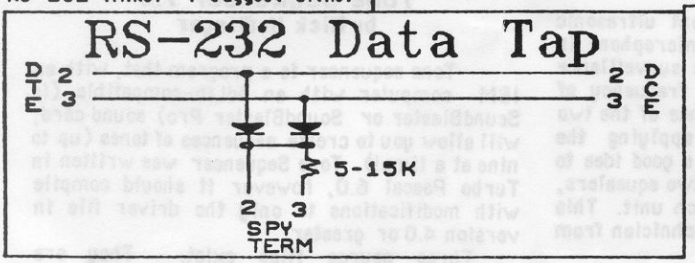
When one uses this device, one must know the baud rate, and communications parameters (word length, stop bits, parity) of the RS-232 link you are eavesdroping on. You should also be aware of any terminal emulation that is being used. For example, DEC VT-52s, VT-100s, and VT-220s send special character sequences to clear the screen, position the cursor, and do other display functions. While you will receive the basic data being sent, without the terminal emulation you will receive "garbage" on the screen when those control sequences are sent. The easiest way to determine the terminal emulation is to simply check the terminal on the RS- 232 link you are eavesdroping on.

For those of you who are now worried about the possibility of someone eavesdroping on your RS-232 links. I suggest either encrypting them, or using some non-standard system to hook your terminals up to your computer system. Wang and IBM use coax cable to hook up their terminals, which is a little harder to tap than RS-232. Most RS-232 terminals can be modified to work with fiber optic cable which is probably as secure as you could get. (Although remember that nothing is ever totally secure.)

You could also modify your terminals so that two unused pins on your terminal's DB-25 connector are connected together. One then uses those two connected lines as a normally closed alarm sensor. When someone pulls out the connector to install the tap, and open circuit condition results and the alarm goes off. Of course, inspecting connections to see if anything weird is on them will do alot to help keep RS-232 (and any other type) taps off your system.

Another simple thing you can do would be to rewire your terminals' and host system's DB-25 connectors so that the transmit and receive data lines go are on pins other than 2 and 3. While with a little bit of time and a breakout box, someone could hack out the connections, that time factor would add too much of a risk for most people. The use of non-standard connectors on your RS-232 link will add even more confusion.

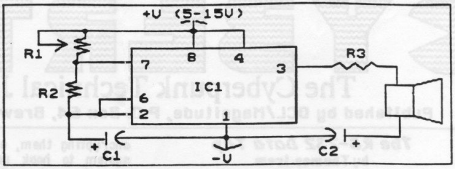


Anyway, thanks go to **John Williams** of **Consumertronics** for his assistance with this project.

## *The Squealer*

by **Thomas Icom**




A squealer is a nickname for a handheld audio tone generator, also known to phone phreaks as a "Pink Box". A squealer is usually able to generate an audio tone anywhere from 20 hz up to 30-40 Khz. Squealers are very versatile devices having many applications for the technological enthusiast or survivalist.

A squealer can be used as a tone signaling unit for quick and dirty remote control operations. It can used by itself for limited range R/C functions, similar to "the clapper" remote control device. A good sqealer design will be capable of outputting a audio signal up to 90db in loudness. That's about equal to a person yelling. How far can you yell? That's a bit further than the distance one can reliably activate an R/C device with a squealer. Ambient noise in the area will reduce range however, and if the ambient noise's frequency is close to your device's, you might get an accidental activation. For added range and reliability, one can use it over a radio or phone link.

Also in the R/C application realm, a squealer can be used to replace lost ultrasonic remote control units used in many older TVs. It also makes the employment of sound activated accessories such as "The Clapper" easier.

One can use a squealer to test audio equipment, and for locating specific wires in a cable bundle.

A squealer can be used as a revenge-type device. When set to the high-end of a person's hearing range (15 Khz. or so) it will cause headaches, nausea, and irritability. It will also be impossible to locate when hidden.

A squealer can also be used to chase off rodents, bats, insects, and dogs. When set to ultrasonic frequencies (above 20 Khz.), it will be totally annoying to these creatures, who will not want to stay around it.

Two sqealers set at different ultrasonic frequencies will jam any hidden microphone in the area they are operating. The surveillance technician will hear a tone; the frequency of which will be equal to the difference of the two squealers' frequencies. When applying the squealer in this role, it would be a good idea to sweep the output frequency of the two squealers, and do so at different rates for each unit. This will help prevent a surveillance technician from filtering out the noise.

Any good audio oscillator circuit can be used as a squealer. The schematic in this article uses a 555 timer IC as the heart of the circuit. The 555 is inexpensive and available at Radio Shack. Other ICs that can be used are the ICL8038 and XR2207. Optionally, you might also want to add an audio amplifier circuit to your squealer for a greater power output.

Since the audio output characteristics of a "stock" squealer are somewhat low, a high efficiency speaker is almost a must. Typical of a good speaker is the Radio Shack #273-091. It is capable of high output with a minimal amount of drive power. This speaker will output a 90db sound level with the circuit in this article.

### Parts List

==========

IC1 - 555 Timer IC
R1 - 10 Meg Ohm Potentiometer
R2 - 1K Ohm Resistor
R3 - Optional Volume Control Resistor, 10-1000 Ohms
C1 - .05 ufd Capacitor
C2 - 4.7 ufd Capacitor

Frequency Determination Formula:

$$\text{Frequency} = \frac{1.44}{(R1+2R2)C1}$$

## *Tone Sequencer 1.0*

by **Nick Haflinger**

Tone sequencer is a program that, with an IBM computer with an AdLib-compatible (ie SoundBlaster or SoundBlaster Pro) sound card, will allow you to create sequences of tones (up to nine at a time!). Tone Sequencer was written in Turbo Pascal 6.0, however it should compile with modifications to only the driver file in version 4.0 or greater.

Three source files exist. They are

DRIVER.PAS, SEQUENCE.PAS, and DIALER.PAS. DRIVER.PAS must be compiled first. Compile it with the TPC command-line compiler by typing TPC DRIVER, with TPC in your path and DRIVER in your current directory. Once you have done this with the driver, you may compile SEQUENCE.PAS and DIALER.PAS in the same manner.

DRIVER is simply an FM-Sound driver for the AdLib that allows you to play pure tones at frequencies of your choice. The difference from the speaker in your IBM-clone is that the speaker in your machine can do only one voice at a time. DRIVER, with a sound board, can do 9 simultaneous tones. This is important because the Phone Company uses DTMF tones, which are two simultaneous tones at different frequencies.

SEQUENCE is the tone sequencer. It allows you to create simple macros of tones that you can play back. Up to 100 sequences, each having up to 100 commands each, can be stored simultaneously. One command is something like End note on channel 0, or Play 2600hz tone on channel 2. You can also call other macros, so that if you want to start the same frequencies at different times, that could be a macro.

DIALER is a program, primarily used to demonstrate the capabilities of the DRIVER. It produces phone tones, namely the normal digits and symbols, the four military tones (A, B, C, and D), and red-box tones.

Most of you probably know what a red box is, but for those who don't, here's a brief explanation: When you put a coin into a pay phone, it makes a noise that the phone company interprets to be you putting money in. The first noise must be accompanied by a real coin, because you need an electrical connection. Afterwards, you can just play the tones and have it believe that you just put more money in. In some areas, this will work only on long-distance calls, however. DIALER has a simple menu to access all the functions. All numbers, the pound sign, and the asterisk, are the keys they represent. Q makes a red-box quarter tone, I makes a red-box dime tone, and N, a nickel tone. A, B, C, and D are the military tones.

Now that you can generate the tones, the question becomes what to do with them. I suggest that you obtain a decent minicassette recorder that has a microphone jack which can be plugged directly into a sound board. If you want to make a specific call, you can record, using either SEQUENCE or DIALER, the complete sequence which you will need played into the phone, including the phone number. When you get to a pay-phone you want to use (Remember - NEVER, EVER, EVER phreak from your own home!), just put in a nickel (if you're using red-box tones), and play the sequence into the handset.

Other uses for the tones are possible, of course. You can write, using DRIVER, a phonebook program that will dial for you through the sound board, for example. There are many devices which use non-phone tones, and you can generate these as well. If you want to access your voice-mail number, you could program in macros for dialing your number, and other peoples' numbers as well.

DRIVER was written out of a logical desire to want to program my $100 sound board without spending another $150 on manuals. Through on-line docs posted by others, I was able to piece together this driver. Jeff Lee's document on programming the FM Chips was invaluable, and many others helped along the way. SEQUENCE was a logical consequence (for anyone in my position, anyway) of DRIVER, but I would not have written it if not for Tom Icom's encouragement. I thank everyone involved in the project. If you have any questions, I can be reached on Uncensored BBS (914)-761- 6877.

## SEQUENCE.PAS

```
{Tone-sequencing program for AdLib/SoundBlaster compatible sound boards
 by Nick Haflinger for Cybertek magazine}
PROGRAM tone_sequencer (input, output, datafile);

USES crt, driver;  {Standard PC functions and AdLib driver}

TYPE
  functions = (play, wait, stop, func, term);   {The allowable functions}
  item = RECORD                                 {variant record for each}
         CASE item : functions OF               {event in the sequence}
           play : (voice, tone : INTEGER);
           wait : (time : INTEGER);
```

```
        stop : (channel : INTEGER);
        func : (macnum : INTEGER);
    END; {item record}
  macro = ARRAY [1..100] OF
  item;
  storage = RECORD
        macr : ARRAY [1..100] OF
        macro;
        name : ARRAY [1..100] OF
        STRING [20];
      END; {storage record}
  data = FILE OF
  storage;

VAR macros  : storage;
  datafile : data;

{Forward Declaration for the DoMacro function so that doitem can call it}
PROCEDURE domacro (macnum : INTEGER); FORWARD;

{Doitem executes an item event (see record, above, for structure) and returns
 a true if the macro is complete}
FUNCTION doitem (what : item) : BOOLEAN;
BEGIN {doitem}
  IF what.item = play THEN
    sendnote (what.voice, what.tone);
  IF what.item = wait THEN
    DELAY (what.time);
  IF what.item = stop THEN
    endnote (what.channel);
  IF what.item = func THEN
    domacro (what.macnum);
  IF what.item = term THEN
    doitem := TRUE
  ELSE
    doitem := FALSE;
END; {doitem}

{Domacro allows you to execute a macro within (or outside) another macro}
PROCEDURE domacro (macnum : INTEGER);
VAR ip : INTEGER;
BEGIN {domacro}
  ip := 0;
  REPEAT INC(ip) UNTIL ( (doitem (macros.macr [macnum] [ip]) ) OR (ip = 100) ); END; {domacro}

{init_all initializes all macros as blank}
PROCEDURE init_all;
VAR count1, count2 : INTEGER;
BEGIN {init_all}
  FOR count1 := 1 TO 100 DO
    BEGIN {for}
    macros.name [count1] := '';
    FOR count2 := 1 TO 100 DO
      macros.macr [count1] [count2] .item := term;
    END; {for}
END; {init_all}
```

# The Omega Man presents:

## Home Defense Made Easy!

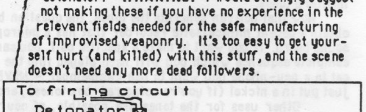
- OR -

## Don't try this at home, kids.

### *Making Your Own Claymore Mines*

(For educational purposes only, of course...)

Claymore mines are easy to make, and devastatingly effective pieces of weaponry. All that is required is an open-ended container of some sort, some high explosive, a detonator, and something to use as shrapnel. You put the high explosive in the container, put the shrapnel on top, and when you're ready to deploy your new toy, put the open end towards the area you want to mess up, and insert the detonator. Anything that's within 50+ yards of the open end will be seriously hurting when it goes off. Those of you who don't have access to high explosives and detonators can substitute a low explosive and a solar igniter for the high explosive and detonator. **WARNING:** I would strongly suggest not making these if you have no experience in the relevant fields needed for the safe manufacturing of improvised weaponry. It's too easy to get yourself hurt (and killed) with this stuff, and the scene doesn't need any more dead followers.



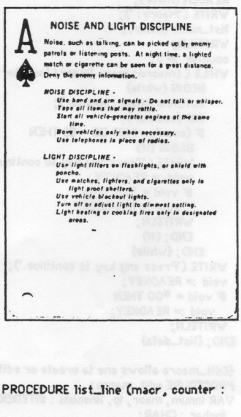

This diagram assumes the use of an electric detonator/blasting cap or setting off the claymore. However, other methods of detonation could be used. For those of you who are interested in improvised weaponry, I strongly suggest you pick up the four volumes of **The Poor Mans James Bond**, and a subscription to **U.S. Militia**.

The **PMJB** Volumes 1-4 are $18 each, and a sub to **U.S. Militia** is $35 a year. Write to:

**Atlan Formularies**
**P.O. Box 95**
**Alpena, AR 72611**

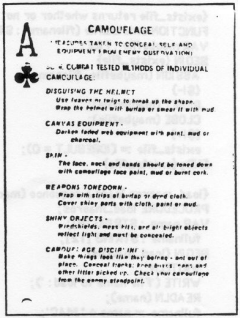
*FIGHT DISINFORMATION WITH THIS INFORMATION*

```
{exists_file returns whether or not a user-specified file exists}
FUNCTION exists_file (filename : STRING) : BOOLEAN;
VAR maybefile : FILE;
BEGIN {exists_file}
  ASSIGN (maybefile, filename);
  {$I-}
  RESET (maybefile);
  CLOSE (maybefile);
  {$I+}
  exists_file := (IORESULT = 0);
END; {exists_file}

{load_macros loads a sequence (macro) file}
PROCEDURE load_macros;
VAR name : STRING [8];
  fullname : STRING [12];
BEGIN {load_macros}
  WRITELN;
  WRITE ('File name to load: ');
  READLN (name);
  fullname := name + '.MAC';
  IF exists_file (fullname) THEN
    BEGIN
    ASSIGN (datafile, fullname);
    RESET (datafile);
    READ (datafile, macros);
    CLOSE (datafile);
    END
  ELSE
    WRITELN ('File not found.');
END; {load_macros}

{save_macros saves all sequences currently in memory}
PROCEDURE save_macros;
VAR name : STRING [8];
  fullname : STRING [12];
BEGIN {save_macros}
  WRITELN;
  WRITE ('File name to save: ');
  READLN (name);
  fullname := name + '.MAC';
  ASSIGN (datafile, fullname);
  REWRITE (datafile);
  WRITE (datafile, macros);
  CLOSE (datafile);
  WRITELN ('File saved.');
END; {save_macros}
```




```
{list_line displays the item on the specified line of the specified macro} PROCEDURE list_line (macr, counter :
INTEGER);
BEGIN {list_line}
  CASE macros.macr [macr] [counter] .item OF
      func : WRITELN (counter : 3, '  Call macro number ',
      macros.macr [macr] [counter] .macnum);
      term : WRITELN (counter : 3, '  TERMINATE MACRO');
      wait : WRITELN (counter : 3, '  Delay          ',
      macros.macr [macr] [counter] .time, ' milliseconds');
```

```
    stop : WRITELN (counter : 3, '  End note on voice ',
    macros.macr [macr] [counter] .channel);
    play : WRITELN (counter : 3, '  Play ',
    macros.macr [macr] [counter] .tone, ' Hz tone on voice number ',
    macros.macr [macr] [counter] .channel);
  END; {case}
END; {list_line}

{List_name displays the name of a specified macro}
PROCEDURE list_name (number : INTEGER);
BEGIN {list_name}
  WRITE (number : 2, ' ', macros.name [number] : 17);
END; {list_name}

{List_data allows a macro to be specified and the whole macro to be displayed
 with screen pausing}
PROCEDURE list_data;
VAR void : CHAR;
  macr, counter : INTEGER;
BEGIN {list_data}
  WRITE ('List which macro? ');
  READLN (macr);
  WRITE ('Macro: ');
  list_name (macr);
  WRITELN;
  counter := 1;
  WHILE ( (macros.macr [macr] [counter] .item <> term) OR (counter = 1) ) DO
      BEGIN {while}
      list_line (macr, counter);
      INC (counter);
      IF (counter MOD 23 = 0) THEN
        BEGIN {if}
        WRITE ('Press any key to continue');
        void := READKEY;
        IF void = #00 THEN
          void := READKEY;
        WRITELN;
        END; {if}
      END; {while}
  WRITE ('Press any key to continue.');
  void := READKEY;
  IF void = #00 THEN
    void := READKEY;
  WRITELN;
END; {list_data}

{Edit_macro allows one to create or edit a tone sequence (macro) }
PROCEDURE edit_macro;
VAR innum, macr, ip, linenum : INTEGER;
  inchar : CHAR;
  mname : STRING [20];
BEGIN {edit_macro}
  WRITELN;
  WRITELN;
  WRITE ('Macro to edit: ');
  READLN (macr);
  WRITE ('Line number to start at: ');
```



CAMOUFLAGE

TECHNIQUES TAKEN TO CONCEAL SELF AND EQUIPMENT FROM ENEMY OBSERVATION)

— A COMBAT TESTED METHODS OF INDIVIDUAL CAMOUFLAGE

DISGUISING THE HELMET
Use leaves or twigs to break up the shape.
Wrap the helmet with burlap or smear it with mud.

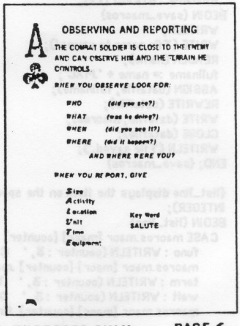
CANVAS EQUIPMENT -
Darken tented web equipment with paint, mud or charcoal.

SKIN -
The face, neck and hands should be toned down with camouflage face paint, mud or burnt cork.

WEAPONS TOREDOWN -
Wrap with strips of burlap or dyed cloth.
Cover shiny parts with cloth, paint or mud.

SHINY OBJECTS -
Windshields, mess kits, and all bright objects reflect light and must be concealed.

CAMOUFLAGE DISCIPLINE -
Make things look like they belong - and not all in one place. Conceal tracks. Keep quiet, don't get other little picked up. Check your camouflage from the enemy standpoint.



OBSERVING AND REPORTING

THE COMBAT SOLDIER IS CLOSE TO THE ENEMY AND CAN OBSERVE HIM AND THE TERRAIN HE CONTROLS.

WHEN YOU OBSERVE LOOK FOR:

WHO (did you see?)
WHAT (was he doing?)
WHEN (did you see it?)
WHERE (did it happen?)

AND WHERE WERE YOU?

WHEN YOU REPORT, GIVE

Size
Activity
Location
Unit
Time
Equipment

Key Word
SALUTE

```
READLN (ip);
DEC (ip);
IF ip = 0 THEN
  BEGIN {if}
  WRITELN;
  WRITE ('Macro Name : ');
  READLN (mname);
  macros.name [macr] := mname;
  END; {if}
WRITELN;
WRITELN ('(P)lay note              (E)nd note');
WRITELN ('(D)elay                  (C)all macro');
WRITELN ('(T)erminate macro');
WRITELN;
REPEAT
  INC (ip);
  linenum := WHEREY;
  WRITE (ip, ': ');
  inchar := UPCASE (READKEY);
  IF inchar = #0 THEN
    inchar := UPCASE (READKEY);
  CASE inchar OF
      'T' : macros.macr [macr] [ip] .item := term;
      'C' : BEGIN {Call}
      WRITE ('Macro to call: ');
      READLN (innum);
      WRITELN;
      macros.macr [macr] [ip] .item := func;
      macros.macr [macr] [ip] .macnum := innum;
      END; {call}
      'D' : BEGIN {delay}
      WRITE ('Delay in milliseconds: ');
      READLN (innum);
      WRITELN;
      macros.macr [macr] [ip] .item := wait;
      macros.macr [macr] [ip] .time := innum;
      END; {delay}
      'E' : BEGIN {end note}
      WRITE ('Voice number to end: ');
      READLN (innum);
      WRITELN;
      macros.macr [macr] [ip] .item := stop;
      macros.macr [macr] [ip] .channel := innum;
      END; {end note}
      'P' : BEGIN {start note}
      WRITE ('Voice number to play : ');
      READLN (innum);
      macros.macr [macr] [ip] .item := play;
      macros.macr [macr] [ip] .voice := innum;
      WRITE ('Tone to play in Hz : ');
      READLN (innum);
      macros.macr [macr] [ip] .tone := innum;
      WRITELN;
      END; {start note}
  END; {case}
  GOTOXY (1, WHEREY);
  list_line (macr, ip);
```

## *Garbage Channels: Another Approach At Finding Frequencies*

**by Thomas Icom**

You notice this group using handheld radios, and desire to listen in on them. You have no clue as to their identity, and doubt they're even licensed. You can't afford one of those neato frequency counters, and while you have an idea as to what band they're on; a traditional frequency search might take too long. While finding their frequency might seem a difficult to impossible proposition, there is a way to increase your chances of a successful "catch". The entire procedure takes only 10 seconds.

The FCC has set aside certain frequencies for low-power business communications between handhelds. These are called "low power", or "itinerant" frequencies. My friend **The Glitch** refers to them as **"garbage channels"**, as anybody with $200 can pick up a pair of HTs on these frequencies and set up shop. A lot of groups who need inexpensive communications capability of a higher quality than CB often go and pick up a few garbage channel units. They're even available at Radio Shack for $149 a pop.

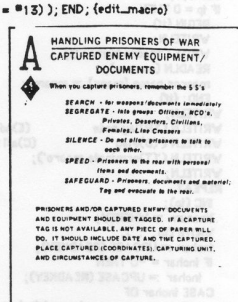
So, there is a good chance that the group you've just encountered might be running on a garbage channel. If you've read the article on Page 10, you should be able to determine what band they're running on. Plug in the frequencies listed below (in Mhz.) that correspond to the right band. You might get lucky.

VHF LOW: 30.84, 33.12, 33.14, 35.02, 35.04, 42.98, 43.04
VHF HIGH: 151.505, **151.625, 154.57, 154.6,** 158.4
UHF: 451.8, 457.525-457.6 (25 Khz. spacing), 464.5, 464.55, 467.75-467.925 (25 Khz. spacing).
800 Mhz.: 851.0125-855.9875

The most commonly used garbage channels are the boldfaced ones in the VHF High Band. If they are on UHF and you have no luck with those frequencies, then try searching 461-465 Mhz. The 12.5 Khz. "splinter" channels are all assigned to "low power" operations. My experience is that of that range, 464-465 Mhz. yields the best results. Also, if they have small cheap looking radios with 2 foot telescoping metal whips try 49.83-50 Mhz. I keep the garbage channels programed into one of my scanner's memory banks. This way if I run into some

**<Continued on Page 10>**

```
  WRITELN;
 UNTIL ( ip = 100) OR (macros.macr [macr] [ip] .item = term) OR (inchar = #13) ); END; {edit_macro}

{List_names lists all the names of all the macros}
PROCEDURE list_names;
VAR count : INTEGER;
 void : CHAR;
BEGIN {list_names}
 FOR count := 1 TO 100 DO
   BEGIN {for}
   IF count MOD (23 * 3) = 0 THEN
     BEGIN {if}
     WRITELN;
     WRITE ('Press any key to continue');
     void := READKEY;
     IF void = #00 THEN
       void := READKEY;
     WRITELN;
     END; {if}
   IF count MOD 3 = 1 THEN
     WRITELN;
   list_name (count);
   END; {for}
 WRITELN;
 WRITE ('Press any key to continue');
 void := READKEY;
 IF void = #00 THEN
   void := READKEY;
 WRITELN;
END; {list_names}

{Menu displays the menu and returns a true when the user is done}
FUNCTION menu : BOOLEAN;
VAR macnum : INTEGER;
 inchar : CHAR;
BEGIN {menu}
 CLRSCR;
 WRITELN ('AdLib/SoundBlaster Tone Sequencer');
 WRITELN ('Written by Nick Haflinger for Cybertek magazine');
 WRITELN;
 WRITELN;
 WRITELN ('Options:');
 WRITELN ('(L)oad macro file      (S)ave macro file');
 WRITELN ('(E)dit macro           e(X)ecute macro');
 WRITELN ('(I()st macro           (N)ames');
 WRITELN ('(Q)uit');
 WRITELN;
 WRITE ('Enter option: ');
 inchar := UPCASE (READKEY);
 WRITELN;
 IF inchar = #00 THEN
   inchar := UPCASE (READKEY);
 CASE inchar OF
    'L' : load_macros;
    'S' : save_macros;
    'I' : list_data;
    'E' : edit_macro;
```



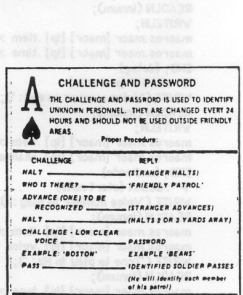
HANDLING PRISONERS OF WAR
CAPTURED ENEMY EQUIPMENT/
DOCUMENTS

When you capture prisoners, remember the 5 S's

*SEARCH - for weapons/documents immediately*
*SEGREGATE - into groups: Officers, NCO's, Privates, Deserters, Civilians, Females, Line Crossers*
*SILENCE - Do not allow prisoners to talk to each other.*
*SPEED - Prisoners to the rear with personal items and documents.*
*SAFEGUARD - Prisoners, documents and material: Tag and evacuate to the rear.*

PRISONERS AND/OR CAPTURED ENEMY DOCUMENTS AND EQUIPMENT SHOULD BE TAGGED. IF A CAPTURE TAG IS NOT AVAILABLE, ANY PIECE OF PAPER WILL DO. IT SHOULD INCLUDE DATE AND TIME CAPTURED, PLACE CAPTURED (COORDINATES), CAPTURING UNIT, AND CIRCUMSTANCES OF CAPTURE.

* U.S. GOVERNMENT PRINTING OFFICE: 1990 O—314 240



CHALLENGE AND PASSWORD

THE CHALLENGE AND PASSWORD IS USED TO IDENTIFY UNKNOWN PERSONNEL. THEY ARE CHANGED EVERY 24 HOURS AND SHOULD NOT BE USED OUTSIDE FRIENDLY AREAS.

Proper Procedure:

| CHALLENGE | REPLY |
|---|---|
| HALT | (STRANGER HALTS) |
| WHO IS THERE? | 'FRIENDLY PATROL' |
| ADVANCE (ONE) TO BE RECOGNIZED | (STRANGER ADVANCES) |
| HALT | (HALTS 2 OR 3 YARDS AWAY) |
| CHALLENGE - LOW CLEAR VOICE | PASSWORD |
| EXAMPLE: 'BOSTON' | EXAMPLE 'BEANS' |
| PASS | IDENTIFIED SOLDIER PASSES (He will identify each member of his patrol) |

REMEMBER:
MEMORIZE CURRENT PASSWORDS AND COUNTER SIGNS. HALT AND IDENTIFY PERSONNEL BEFORE THEY ARE CLOSE ENOUGH TO BE A THREAT.
KEEP THE INDIVIDUAL COVERED AND DO NOT EXPOSE YOURSELF.
SPEAK CLEARLY AND JUST LOUDLY ENOUGH TO BE HEARD.
DO NOT PERMIT PERSON TO PROCEED UNTIL PROPERLY IDENTIFIED.

```
    'N' : list_names;
    'X' : BEGIN
    WRITE ('Macro to execute: ');
    READLN (macnum);
    domacro (macnum);
    END;
  END; {case}
  IF inchar = 'Q' THEN
    menu := TRUE
  ELSE
    menu := FALSE;
END; {menu}

BEGIN {main}
  init_all;              {initializes all macros as blank}
  REPEAT
  UNTIL menu;     {executes until menu says that user is done}
END. {main}
```

## DIALER.PAS

```
{Phone_dialer by Nick Haflinger for CyberTek magazine}
{Phone_dialer will dial the user's phone through the sound board}
{it also doubles as both a red and silver box}
PROGRAM phone_dialer (input, output);

USES crt, driver;

PROCEDURE check_card_installed;
BEGIN {check_card_installed}
  IF NOT (exists_card) THEN
    BEGIN {if}
    CLRSCR;
    WRITELN ('No AdLib card found!');
    HALT (1);
    END; {if}
END; {check_card_installed}

PROCEDURE twotones (tone1, tone2 : INTEGER);
BEGIN {twotones}
  sendnote (0, tone1);
  sendnote (1, tone2);
END; {twotones}

PROCEDURE endtwo;
BEGIN {endtwo}
  endnote (0);
  endnote (1);
END; {endtwo}

PROCEDURE nickel;
BEGIN {nickel}
  twotones (1700, 2200);
  DELAY (66);
  endtwo;
END; {nickel}
```



### A PATROL PLANNING STEPS

The Patrol Leader considers all steps and accomplishes those necessary. The order may vary.

- STUDY THE MISSION
- PLAN USE OF TIME
- STUDY TERRAIN AND SITUATION
- ORGANIZE THE PATROL
- SELECT MEN, WEAPONS, EQUIPMENT
- ISSUE WARNING ORDER
- COORDINATE
- MAKE RECONNAISSANCE
- COMPLETE DETAILED PLANS
- ISSUE ORDER
- SUPERVISE, INSPECT, REHEARSE
- EXECUTE THE MISSION

### PATROL REPORT

Patrols are debriefed at the time of return to the friendly area. The report should include the following items:

- SIZE AND COMPOSITION OF PATROL
- TASK
- TIME OF DEPARTURE
- TIME OF RETURN
- ROUTES (OUT AND BACK)
- TERRAIN
- ENEMY
- MAP CORRECTIONS
- MISCELLANEOUS INFORMATION
- RESULTS OF ENEMY ENCOUNTERS
- CONDITION OF PATROL (DISPOSITION OF DEAD / WOUNDED)
- CONCLUSIONS AND RECOMMENDATIONS

### A COMMUNICATIONS SECURITY TIPS

COMSEC prevents the enemy from gaining useful information from communications. Everyone must practice COMSEC – MAKE IT A HABIT!

*ALWAYS FOLLOW THESE RULES:*

- USE WIRE INSTEAD OF RADIO IF POSSIBLE
- NEVER VIOLATE RADIO SILENCE
- MAKE TRANSMISSIONS BRIEF, NEVER MORE THAN ONE MINUTE OR LESS
- USE PROPER RADIO/TELEPHONE PROCEDURE
- USE MINIMUM POWER REQUIRED
- NEVER TRANSMIT CLASSIFIED INFORMATION IN THE CLEAR
- USE ONLY AUTHORIZED CODES
- USE AUTHENTICATION PROPERLY
- SITE TRANSMITTING ANTENNAS ON REVERSE SLOPE (AWAY FROM ENEMY)
- USE DUMMY ANTENNA TO TUNE OR TEST RADIO
- TRY TO WORK THROUGH JAMMING
- NEVER SAY ANYTHING OVER THE AIR ABOUT BEING JAMMED

```
PROCEDURE dime;
BEGIN {dime}
 nickel;
 DELAY (33);
 nickel;
END; {dime}

PROCEDURE quarter;
VAR count : INTEGER;
BEGIN {quarter}
 FOR count := 1 TO 5 DO
    BEGIN {for}
    twotones (1700, 2200);
    DELAY (33);
    endtwo;
    DELAY (33);
    END; {for}
END; {quarter}

PROCEDURE number (num : CHAR);
BEGIN {number}
 CASE UPCASE (num) OF

    {white box/silver box tones (keypad)}
    '1' : twotones (697, 1209);
    '2' : twotones (697, 1336);
    '3' : twotones (697, 1477);
    'A' : twotones (697, 1633);
    '4' : twotones (770, 1209);
    '5' : twotones (770, 1336);
    '6' : twotones (770, 1477);
    'B' : twotones (770, 1633);
    '7' : twotones (852, 1209);
    '8' : twotones (852, 1336);
    '9' : twotones (852, 1477);
    'C' : twotones (852, 1633);
    '*' : twotones (941, 1209);
    '0' : twotones (941, 1336);
    '#' : twotones (941, 1477);
    'D' : twotones (941, 1633);

    {red box tones}
    'N' : nickel;
    'I' : dime;
    'Q' : quarter;

 END; {case}
 DELAY (100);
 endtwo;
END; {number}

FUNCTION readchar : BOOLEAN;
VAR option : CHAR;
BEGIN {readchar}
 option := READKEY;
 number (option);
```

## *Quick guide to finding band of a portable by looking at the antenna*

These measurements are for the actual antenna, not including the mount between the portable and the antenna, which is usually either a BNC connector or a screw in connector, which can be a bigger diameter at the base and add to the overall length.

If the antenna is longer than about 12 inches, and about 1/2 inch diameter, it is probably VHF-LO band, usually in the 30-50 MHz range.

If the antenna is about 6 inches long, and about 1/4 inch in diameter, it is probably VHF-HI band, in the 150-165 MHz range.

If the antenna is either 2 inches long and about 1/4 inch in diameter, or about 6 inches long and 1/8 inch in diameter, it is probably in the UHF band, in the 450-470 MHz range.

If the antenna is about 2 inches long and about 1/8 inch in diameter (sometimes mounted at the top of another antenna that is about 5 inches long and about 1/4 inch in diameter), then it is probably in the 800 MHz band, or possibly even the new 900 MHz band, and the range is possibly in the 840-860 MHz range, if the portable is using a repeater system.

These ranges should be fairly accurate, with some exceptions like if the radio is licensed by the federal government in which case the frequencies tend to be in different ranges than those frequencies set aside for business/public safety.

**<Continued From Page 7>**

people with HTs, I bring that bank up to do a quick check. In many cases that's all I have to do to get their frequency. I keep the garbage channel active when I'm travelling in an area I have no frequencies for. The wide use of these frequencies everywhere ensures that I always have something to listen to. Usually it also turns out to be something interesting, as a lot of "security" forces use the garbage channels.

## *SIMPLE TV JAMMER*

### SYMBOLS

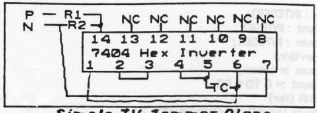
TC means trimmer capacitor. Use any trimmer (variable) capacitor from 3-50 microfarads. Or just experiment! Radio Shack has a few different kinds.

**<Continued Page 11>**

```
  IF UPCASE (option) = 'X'THEN
    readchar := FALSE
  ELSE
    readchar := TRUE;
END;

PROCEDURE menu;
BEGIN {menu}
  CLRSCR;
  WRITELN ('Phone Dialer  by Nick Haflinger for Cybertek magazine'); WRITELN;
  WRITELN ('keys: 0-9, #, *  = normal phone keypad');
  WRITELN ('     A, B, C, D  = silver box tones');
  WRITELN ('     N, I, Q     = (N)ickel, d(I)me, (Q)uarter red box tones'); WRITELN;
  WRITELN ('X = e(X)it');
  WRITELN;
  WRITELN;
END; {menu}

BEGIN {main}
  check_card_installed;
  menu;
  REPEAT
  UNTIL readchar = FALSE;
END. {main}
```

## DRIVER.PAS

```
{AdLib Driver by Nick Haflinger for Cybertek magazine}
UNIT driver;  {AdLib driver for Tone Sequencer}

INTERFACE

FUNCTION  exists_card : BOOLEAN;          {is there a sound card?}
PROCEDURE init_adlib;                     {Inits the AdLib board}
PROCEDURE endnote (voice : INTEGER);      {Ends a note directly}
PROCEDURE sendnote (voice, freq : INTEGER);  {Sends a note}

IMPLEMENTATION

TYPE tenbitbin = ARRAY [0..9] OF
  0..1;
TYPE internal_instchr = ARRAY [1..6, 1..2] OF
  BYTE;

{POWER returns the 2 to the nth power, n>=1}
FUNCTION power (n : INTEGER) : INTEGER;
VAR accum, count : INTEGER;
BEGIN {power}
  accum := 1;
  FOR count := 1 TO n DO
    accum := accum * 2;
  power := accum;
END; {power}

{CVTBIN converts a number to a ten bit binary number}
PROCEDURE cvtbin (VAR outnum : tenbitbin;
```




**Simple TV Jammer Plans**

**<Continued from Page 10>**

IC means integrated circuit. Use a TTL 7404 or 74LS04 Hex Inverter. Can get this at any respectable electronic shop. Radio Shack part number 276-1802 for about 99 cents.

R1 and R2 are resistors. TTL chips require a supply voltage of 5 volts (DC). Since we are using a standard 9 volt battery, we need a voltage divider network. Choose R1 and R2 such that R1/R2=0.8. Example, if R2=1000 ohms (1K- ohm) then choose R1=800 ohms.

Vo(Pin 14)=Vi(P)*R2/(R1+R2)

P means the positive terminal of the 9 volt battery and N means the negative terminal of the 9 volt battery.

Silence
is the voice
of complicity.

### CONNECTIONS

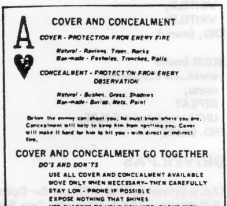
Connect Pin 2 to Pin 3.
Connect Pin 4 to Pin 5.
Connect Pin 5 to one side of trimmer capacitor (TC).
Connect Pin 6 to other side of trimmer capacitor (TC).
Connect Pin 7 to negative side of 9 volt battery.
Connect Pin 1 to Pin 6. * DO NOT CONNECT TO PIN 8.7
NC means No Connections

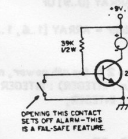
### OPERATION

Use a breadboard for testing and/or solder the finished project on a small piece of

**<Continued Page 17>**

```
 innum : INTEGER);
VAR count : INTEGER;
  worknum : INTEGER;
BEGIN {cvtbin}
  worknum := innum;
  FOR count := 0 TO 9 DO
     BEGIN {for}
     outnum [count] := worknum MOD 2;
     worknum := worknum DIV 2;
     END; {for}
END; {cvtbin}

{SENDREG sends a specified value direct to the AdLib register specified} PROCEDURE sendreg (portreg, outvalue
: BYTE);
VAR temp, count : BYTE;
BEGIN {sendreg}
  port [$388] := portreg;
  FOR count := 1 TO 6 DO
     temp := port [$388];
  port [$389] := outvalue;
  FOR count := 1 TO 35 DO
     temp := port [$388];
END; {sendreg}

{EXISTS_CARD checks for the presence of an AdLib or compatible card}
VAR count, t1 , t2, void : INTEGER;
FUNCTION exists_card : BOOLEAN;
BEGIN {exists_card}
  sendreg (4, $60);                {resets timers}
  sendreg (4, $80);                {enables timers}
  t1 := port [$388];                 {get current timer}
  sendreg (2, $FF);                {reset timer 1}
  sendreg (4, $21);                 {start timer 1}
  FOR count := 1 TO 123 DO
     t2 := port [$388];    {wait for timer (80ms)}
  exists_card := (( (t1 AND $E0) = 0) AND ( (t2 AND $E0) = $C0) );
END; {exists_card}

{init_voice sends the characteristics for a pure tone directly to the card} PROCEDURE init_voice (voice :
BYTE);
BEGIN {init_voice}
  sendreg ($20 + voice, $21);
  sendreg ($40 + voice, $1C);
  sendreg ($60 + voice, $F3);
  sendreg ($80 + voice, $0F);
  sendreg ($C0 + voice, $01);
  sendreg ($E0 + voice, $00);
  sendreg ($23 + voice, $01);
  sendreg ($43 + voice, $00);
  sendreg ($63 + voice, $03);
  sendreg ($83 + voice, $0F);
  sendreg ($C0 + voice, $01);
  sendreg ($E3 + voice, $00);
END; {init_voice}

 {INIT_ADLIB is a quick and dirty init for the AdLib card}
 PROCEDURE init_adlib;
```



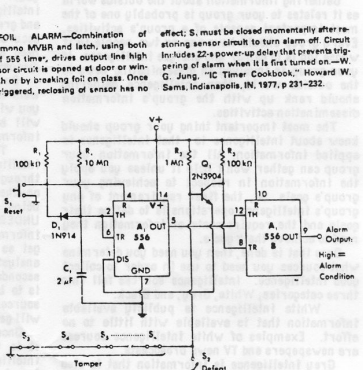

CIRCUIT-BREAKING ALARM—Operates from small 9-V battery, making it independent of AC power failure. Opening of switch or equivalent breaking of foil conductor removes ground from base of transistor, to energize alarm.—Circuits, *73 Magazine*, April 1973, p 132.



FOR EDUCATIONAL PURPOSES ONLY

```
VAR count : BYTE;
BEGIN {init_adlib}
  FOR count := 0 TO $F5 DO
    sendreg (count, 0);
  sendreg (1, 32);
  FOR count := 0 TO 8 DO
    init_voice (count);
END; {init_adlib}

{ENDNOTE ends a note on the voice channel specified}
PROCEDURE endnote (voice : INTEGER);
BEGIN {endnote}
  sendreg ($b0 + voice, 0);
END; {endnote}

{sendnote sends a note directly to the sound card with the specifed vox/freq} PROCEDURE sendnote (voice, freq
: INTEGER);
VAR count, fnum, block, outnum1, outnum2 : INTEGER;
  converted : tenbitbin;
BEGIN {sendnote}
  IF freq < 97 THEN
    block := 1
  ELSE
    IF freq < 195  THEN
      block := 2
    ELSE
      IF freq < 389  THEN
        block := 3
      ELSE
        IF freq < 780  THEN
          block := 4
        ELSE
          IF freq < 1561  THEN
            block := 5
          ELSE
            IF freq < 3122  THEN
              block := 6
            ELSE
              IF freq < 6244  THEN
                block := 7
              ELSE
                IF freq < 12487 THEN
                  block := 8;

  CASE block OF
    1 : fnum := ROUND (freq / 0.095);
    2 : fnum := ROUND (freq / 0.191);
    3 : fnum := ROUND (freq / 0.381);
    4 : fnum := ROUND (freq / 0.763);
    5 : fnum := ROUND (freq / 1.526);
    6 : fnum := ROUND (freq / 3.052);
    7 : fnum := ROUND (freq / 6.104);
    8 : fnum := ROUND (freq / 12.207);
  END; {case}

  cvtbin (converted, fnum);
  outnum1 := (32 + (4 * block) + (converted [9] * 2) + converted [8]);  outnum2 := 0;
```

WINDOW-FOIL ALARM—Combination of power-up mono MVBR and latch, using both sections of 555 timer, drives output fine high when sensor circuit is opened at door or window switch or by breaking foil on glass. Once alarm is triggered, reclosing of sensor has no effect; S, must be closed momentarily after restoring sensor circuit to turn alarm off. Circuit includes 22-s power-up delay that prevents triggering of alarm when it is first turned on.—W. G. Jung, "IC Timer Cookbook," Howard W. Sams, Indianapolis, IN, 1977, p 231–232.






FOR EDUCATIONAL PURPOSES ONLY

```
  FOR count := 0 TO 7 DO
    outnum2 := outnum2 + (converted [count] * power (count));
  sendreg ($a0 + voice, outnum2);
  sendreg ($b0 + voice, outnum1);
END; {sendnote}

BEGIN {main}
  {Unit initialization}
  init_adlib;                          {Initialize card}
END. {main}
{End unit initilization}
```

## *"Where do we start?"*
## *Resistance Op's in the 90's*
## *Part II, Intelligence Operations*

by **Thomas Icom**

Gathering information about the outside world as it relates to your group is probably one of the most important aspects of a group's activities. Every group should have one or two people in it whose job is to gather and analyze intelligence. If your group is involved in disseminating news to the outside world, then intelligence gathering should rank up with the group's information dissemination activities.

The most important thing your group should know about intelligence is that intelligence is applied information. All the information your group can gather won't help it unless you apply the information in respect to achieving your group's goals. So, the first requirement of any group's intelligence operations is to define their goals and the requirements of information they need to achieve their goals.

Once that is done, then you need you determine what sources you need to use in order to collect good intelligence. Intelligence sources fall into three categories; White, Grey, and Black.

White intelligence is publicly available information that is available with little to no effort. Examples of white intelligence sources are newspapers and TV news broadcasts.

Grey intelligence is information that is also publicly available, but requires research effort to acquire. Examples are non-commercial radio communications (ie. public safety communications you hear on a scanner), certain trade publications which have limited distribution, and social engineering select people who have expertise in the field you are gathering intelligence on.

Black intelligence is information which is not publicly available, and thus can only be generally attained by illegal means. Examples are the breaking and entering of a target's property to duplicate certain pieces of information, the use of surveillance equipment, and the blackmail of a target's employee who has an exploitable weakness. In my opinion, the gathering of black intelligence is, for the most part, a waste of time and too risky for a group to engage in. Should your group have a real need for black intelligence, either a good job of collecting white and grey intelligence should be adequate for you to accurately guess any proprietary information you need, or your group is way beyond anything I'll be discussing here.

Once you defined the intelligence sources you want to use, you can then gather the information you will need for good intelligence. This process will be constantly occurring, as the flow of information will never stop.

The prime requirement for your group's intelligence gathering operatives is that they are through. There will often be an urge to do on-the-spot analysis of the information as it's being gathered. THIS SHOULD NOT BE DONE UNDER ANY CIRCUMSTANCES! When gathering information to be turned into intelligence, just get as much as possible, and give it to the analysts. It's their job to figure it out. A secondary requirement for intelligence gatherers is to be on the lookout for new intelligence sources, as old ones go dry. This skill is one that will get better with time.

Once you have gathered your information, then it is the job of the analyst to turn it into intelligence. Intelligence analysts are a rare breed, as the job requires not only top expertise in their respective field, but also an uncanny knack of figuring out complex patterns, and putting together many different seemingly unrelated pieces of information into a coherent picture which tells how the intelligence affects the group's goals. Many of these types of people will seem a little flakey, but if you find one, hold onto him, as they are a rare and valuable breed. It should be noted that the military's A.A.T., or Analytical Aptitude Test; which is used to determine a person's analytical capacity, is said

to have a 90% "failure" rate.

Usually, due to a group's size limitations, one person will be assigned to the task of both gathering and analyzing intelligence. If this is the case, then it is very important that the group's intelligence specialist have a "split personality" when it comes to the different tasks of gathering and analyzing intelligence. From personal experience I can say there will be a strong urge to do a quick analysis of information as it is being gathered. If you do this, you won't be half as successful as if you just simply collected the information and then looked it over later. Keep the jobs separate.

Once the information has been analyzed and turned into intelligence, then it should be quickly sent to the group's command staff for action. Even the picture is incomplete, it should still be turned over to the policymakers; who should take into account that the information is incomplete. What's important here is that the group's leader's be constantly informed as to upcoming and current developments. This way, they can make informed decisions in regards to group activities and policy.

This is also the point where it is decided whether or not the intelligence is disseminated to the outside world, if your group is involved in information/news dissemination. Working on the basis of incomplete intelligence can often result in a bogus story. However, waiting until you have a complete picture will often result in the story losing its time value.

Once a decision has been made to release the intelligence to the public as news, then steps must occasionally be made to sanitize the intelligence in order to disguise the source. This is often needed to preserve the integrity of an intelligence source so you can continue to use it. However, sanitzing a story too much can often make it lose its credibility. So thus, a good balance must be found.

I hope everyone found this article informative, and that it accomplished it's objective in giving everyone a basic primer on intelligence operations. Intell ops skills are best acquired by going out and doing it, so get to it! Any questions can be referred to me via my V.M.B. @ 10288-0700-751-2600, Box 4266; via Uncensored BBS @ 914-761-6877, or by writing me at the magazine.

***FIGHT DISINFORMATION***
***WITH***
***THIS INFORMATION***

## *Reflections*

by **Beltane**

***Ed. Note:*** This article was originally a message taken off of **Black Crawling Systems**, a Cyberpunk/Survivalist BBS in Boston. **(617-482-6356)** Since the author showed some good ideas, permission was acquired to reprint it here. With that, we'd like to thank the author for allowing us to reprint it.

Some ideas that I myself have concluded are that the technology of survival should of necessity range from stone age to high tech. Here's what I mean, any rifle becomes a paperweight once the ammunition is gone, I myself prefer the compound bow. No licensing (anonymity) soundless, the projectiles can be manufactured by hand, and they are extremely effective in the hands of an expert. A backpack properly outfitted is essential, there should be enough freeze dried food for a minimum of 14 days. All equipment should be simple the highest quality and reliable. A modified programmable scanner is important also, when martial law is declared they aren't going to want people wandering around. Waterproof maps (detailed topo, etc) of the escape route are essential. Doesn't hurt to do a little camping also to check everything out a couple times a year. Don't count on being able to use the roads either, checkpoints, weird people, etc. Stick to remote locations and use a dirt/street bike. Practice with your equipment, an emergency is no time to be reading the manuals for anything, or to be practicing your shooting skills. Forget about expensive cumbersome junk, there will be more than enough toys to be had at anytime thereafter, simply take what you need. Prepare to be as well outfitted as the military, buy the same equipment. I prefer Brigade Quartermester. Computers will be worthless without phone lines. If you survive the initial disaster you can come back and get all of that junk that you want anyway at anytime, so a pocket PC is all that's needed when there may be the occasion to use a payphone. They fit in a backpack no sweat, I use mine camping all the time. Don't forget a solar recharger. Where do you recharge it when the utilities are gone? All this stuff is cheap and will work..... guaranteed. Don't forget what they taught you in the military, keep it simple. I don't intend to allow the lunatic political behavior of a collection of oxymorons get me, you shouldn't either.

## *Wireless Reception of Cable TV*

by **Thomas Icom**

The difference between cable TV and regular TV can be seen in cable TV's "official" name, Community Antenna Television (CATV). Basicelly, with CATV, the cable company installs a TV antenna, and a satellite dish at a good receiving location somewhere (they call this place a "head- end"), and then re-transmits the TV signals over cable to customers' homes and businesses. CATV systems use frequencies between 108-400 Mhz. to send their extra channels through the system. The mid-band (channels 14-22) is between 108-174 Mhz., super-band (23-36) is between 200-290 and hyper-band (37-53) is between 290-400 Mh. (These ranges are approximate.) To keep the CATV signals from interfering with the regular services on those frequencies. the cable company must keep leakage from their system down to an absolute minimum. This is easier said than done, and no matter how hard they try, CATV systems still leak to some extent, however minimal it may be.

With equipment available from your local Radio Shack along with your TV, you can pick up low-level CATV signal leakage and get free CATV service without having to pay an installation fee or monthly service charges. This is all perfectly legal because you (currently) have the right to receive radio signals that come onto your domain, and in all reality the cable TV company shouldn't be letting these signals leak out of their system to begin with. When attempting this you should be aware that if your cable TV company is competent, you might not find an area in your neighborhood's outside plant that has enough leakage to get a good signal. Also, the way the government has been acting lately, one never knows when they might decide people who experiment with low-level RF signal reception are a threat to society. If such a thing does occur, make the necessary preparations before one of those UH-60 Blackhawks full of men with MP5SD3s lands on your lawn to bring you into "protective custody".

Your wireless CATV reception station will need the following:

- Fringe reception TV antenna
- 10-20 db gain TV reception amplifier. Preferably one with an amplifier module that mounts near the antenna and a remote power supply. The two piece models are better beause they amplify the signal before it gets noise from going through the coax.
- Cable TV converter box
- Cable descrambler for your system, if you want to receive "premium" channels (optional)
- television (obviously)
- Good quality RG-6 (or better) coax cable with connectors and 75-300 Ohm transformer if your TV doesn't have a 75 Ohm antenna input
- TV antenna mast and other materials for mounting the antenna on your roof

The system is put together as shown in Figure 1. If your CATV descrambler is also a converter, then put it where the CABLE CONV. goes. If it is one of those types that descrambles channel 3 from a separate converter than hook it up as shown in Figure 2.

Once you have your system put together you have to find a leak to receive off of. The cable companies find leaks by driving down the road with a field strength meter. You can do this by walking under the cable and stopping where you see a peak on the meter. You can get a field strength meter for under $50 at Radio Shack. However, unless you use a tuned field strength meter you will also register any strong RF source. If you have a portable TV such as a Sony Watchmen, you can tune to a channel that you normally won't receive and walk around until you get reception.

On a side note, many of those portable TVs that use an analog tuning dial (as opposed to a channel selector knob) often are able to receive mid-band cable channels without a converter box. (The Sony Watchman Model FD-10A can.) To check if your portable TV can do this, examine the TV's VHF (Ch. 2-13) tuning selector. The mid-band capable TV's have only 1 band for VHF selection. To get mid-band with these TV's, simply tune between channels 6 and 7. You will also be able to pick up various types of voice commo from 88-174 Mhz, although the selectivity and sensitivity on portable TV's suck for picking up narrow-band FM voice. You might fare better getting in some of the stronger FM broadcast stations on 88-108 though.

When looking around for a good CATV leak,

**Figure 1**

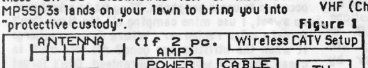
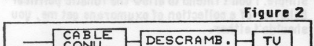



**Figure 2**




Hook-up when using add-on descrambler.

check under junction boxes where the main line is tapped to run cable drops to subscribers. They often degrade from being exposed to the elements. Also check around the homes of neighbors who you feel might have added extra CATV extensions to their house's wiring plant. If they have an antenna on their roof, then chances are they took the wiring plant that was hooked up to their antenna, and hooked it up to the cable system. The cable company generally uses good quality (expensive) materials when installing cable in someone's house. Most people when extending their system go to a department store or Radio Shack, and buy the cheapest (low quality) stuff they can find. This often leads to CATV leakage. Usually, the CATV company doesn't notice the leakage from a home unless it is really severe, as their detection equipment consists of a field strength meter and a 1/4 wave antenna. And since their readings are taken while going 40 MPH down a road, the signal from inside a house set back XXX feet from the road is usually not strong enough for them to take notice. You on the other hand, are using an antenna that has 100 times more gain than theirs, and are amplifying the signal your antenna picks up even more.

Once you've found a leak, aim your antenna at it. You should then be receiving free cable TV. The reception may not be perfect, but what do you expect for nothing? To get better reception, try one or more of the following:

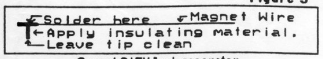
1. Position your antenna closer to the source of the leak
2. Use a higher gain (bigger) antenna
3. Use a more powerful amplifier

If your leak still isn't adequate or you haven't found one at all, then use the following technique. Be careful with this one, as it will royally piss off the cable co., FCC, FBI, FAA, NSA, PUC, and your local chamber of commerce if you're cable co. happens to a member. Not to mention that if you get caught, you could be charged with vandalism, theft of services and possibly sedition. You'll need the following:

- straight pin
- 8-10 foot length of magnet wire (length not critical)
- "liquid black tape", Newskin (liquid bandage), or similar spray-on insulator material
- soldering iron and solder
- crazy glue

To start, solder one end of the length of magnet wire to the blunt end of the straight pin. Then

**Figure 3**




Covert CATV leak generator.

apply the insulator material to the straight pin, except for the pointed end. Let the insulator dry. The insulation on the pin is the important part, as it is needed to prevent you from accidently shorting out your neighbors cable line. This will look something like Figure 3.

When that's done take assembly and crazy glue over to your neighbor that has cable TV. Find a nice run of coax and push the pin in. Make sure the tip has good contact with the center conductor of the coax. Then unroll the magnet wire and crazy glue it to the coax in a way that it won't be noticeable.

What you just did was add an antenna to your neighbors cable. You will now be able to enjoy the use of your wireless cable TV setup. The main problem with this setup is that the resulting "leak" will most assuredly be noticed by the cable co. who will come over and inspect everything. Your best bet is to install this device during the early evening and remove it the next morning before the repair crews hit the road. This way, it is less likely to be detected, unless your cable co.'s repair crews work at night.

**<Continued From Page 11>**

perfboard. Don't forget to get something to hook the battery to the board.

Hook up the battery and turn on the TV to channel 3. SLOWLY turn the TC and watch the TV. You should see the picture go crazy. If not, check all your connections. If everything is OK and still nothing happens, check your battery, it may have given up the ghost. Get a new battery. Try different channels. All I could get on channel 11 were a bunch of squiggly lines. But the most important thing to remember is to experiment!



**FOR EDUCATIONAL PURPOSES ONLY**