# Hellschreiber

## *An old mode still has performance that's hard to beat*

## Part I—History and description

Murray Greenman ZL1BPU

**TIRED of the noise on the low HF bands? Interested in QRP? LF operation? Digital operation on noisy bands, net operation? Digital modes for your ALIVO transceiver? Digital Hellschreiber is the answer!**

**Part I of this series describes the development of Hellschreiber and, in simple terms, the technical details of this uncomplicated but interesting digital mode. Part II describes how *you* can easily get on the air with Hellschreiber—probably with equipment you already own. Part III will have all the technical analysis stuff that digital mode aficionados enjoy.**

### What is Hellschreiber?

Hellschreiber (Hell writing) was invented by Rudolf Hell around 1927, the same year the BBC was founded, as a way of reliably sending text under poor radio conditions. Hellschreiber has nothing to do with the underworld, as the word "Hell" means "bright". At the time, teleprinting as we know it was still in its infancy and, due to the limited capabilities of contemporary electronics, was not usable by radio, was complex and required large heavy equipment. Rudolf Hell's simple system was, therefore, quite different and revolutionary. The Hell system sent text in a similar way to facsimile—it scanned the text and sent it as dots, in fact just like dots sent to a dot matrix printer:


There were military and commercial versions of the equipment, which provided a simple method of sending text, both plain language and ciphers, from portable stations by radio. The equipment was used in the field by the Germans during World War II, including sending "Enigma" coded messages, and it is believed that equipment to intercept transmissions was built in both the UK and the USA. Examples of the original equipment are still around—notably in Holland, although I have heard from hams who saw demonstrations of the Wehrmacht Feld-Hell equipment in England after the war. The Australian Communications Authority still lists an approval for a Siemens Hellschreiber Printer Model 80, permit C70/23 /0080.

Recently I learned of a Silent Key in Napier who had two sets. Where are they now? I understand that both the RNZAF and RNZN ran trials using Hellschreiber on circuits from Waiouru to Europe during the early fifties—does anyone have information about these trials?

From a military point of view the system had great advantages—simple to operate in the field, relatively low power requirement, and best of all, ideal for clandestine operation because no signal was sent until a key was pressed. Unlike CW, skilled operators were not required.

The wartime Siemens and Halske manufactured machines were about the size of a large typewriter, and contained a small motor and a drum with contacts on it; one ring of contacts representing each letter or number. The drum rotated once for every key pressed, then stopped and the contacts keyed a simple CW transmitter. The printer in the unit contained a solenoid, operated by a valve amplifier from the audio tones received from a CW receiver. The solenoid tapped on the back of a strip of paper whenever a dot was received. Above the paper was a rotating spiral, rather like a worm gear, tipped with ink, which scanned the paper strip, the same paper then widely used to print telegrams. This article concentrates on "modern" digital Hellschreiber, but a very good description of the mechanical machines was published in *Ham Radio Magazine* in December 1979. The original article and most of the pictures are available on the Internet.[1] [See also *Radio Bygones*, No 51, February/March 1998—Editor.]

### Technical description

Characters were sent as "dot matrix" dots, one pixel (picture element) at a time. The transmitter scans up each column, and across the character columns, one dot at a time. In Figure 1, the magnified pixels are numbered in the order transmitted. The black pixels are transmitted as Morse key-down, and the white spaces are key-up, like the spaces between Morse code dots. The pixels are higher than they are wide. In the traditional mechanical system, they were printed by a tiny hammer tapping the back of a slowly moving roll of paper above which rotated a small worm gear coated with ink. The worm rotated every 57.12 ms, or 1050 rpm, and the movement of the paper caused the characters to lean forward slightly, like italics.

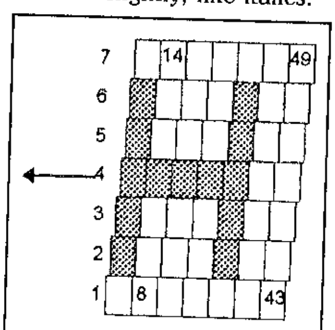
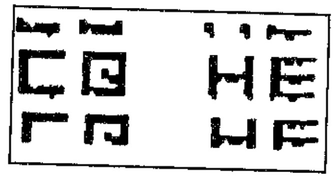



Figure 1. Character slant due to paper movement.

The original character set was very limited, just upper case letters, numbers and very few punctuation characters, but there is no reason why more characters could not be used. Some compromises would need to be made due to the low resolution; lower case descenders, in such characters as p and q would require some ingenuity. The beauty of the system is that no matter what characters are sent, they will be faithfully reproduced, as the receiver has no concept of character set, language or character length. It is possible, for example, to include extra long characters, or use a proportional font.

Many modern digital Hellschreiber systems include a double width transmit system (each column sent twice), which is much easier to read when conditions are very bad, although the character transmission rate is halved. See the example below:




Some programs also include the ability to average the repeated columns of double width printing to reduce the effects of noise. Other systems use double width as "upper case", although the characters are all upper case already!

In the traditional German Feld-Hell system, the characters were sent with a pixel rate of 122.5 Hz, each pixel occupying 8.16 ms, creating a seven by seven dot matrix. This is the standard still most used today. The transmitter carrier was keyed in an on-off fashion, like CW, and the resulting signal was about 400 Hz wide, similar to 45 baud RTTY or CW at 80 wpm! The data throughput was much lower, at 2.5 characters per second, compared with six characters per second for 45 baud RTTY and about 6.5 characters per second for 80 wpm CW. In performance Hellschreiber can be compared with 30 wpm CW, which has a similar data throughput.

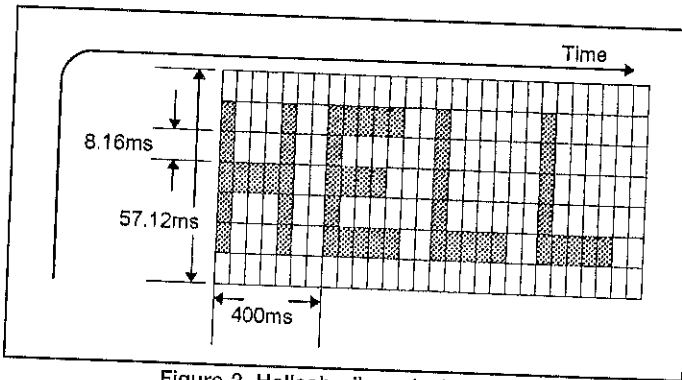



Figure 2. Hellschreiber pixel format.

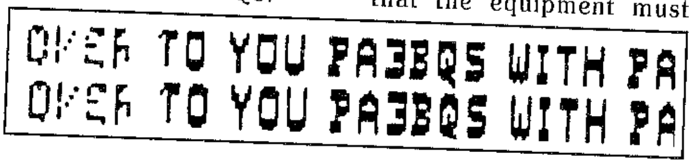
The Feld-Hell system had no synchronization mechanism, simply relying on the pixel rate being similar at each station. (The old machines had a rheostat for adjustment of motor speed.) With seven pixels per character column, of which five printed, and seven columns per character, including the inter-character space (five columns print), each character therefore too $8.16 \times 7 \times 7$ ms $= 400$ ms to transmit. Put another way, $122.5$ Hz $\div 7 = 17.5$ columns/second, and $17.5$ cols/sec $\div 7 = 2.5$ characters/second.

Because there was no synchronism, if there was a difference between the sampling rate at the receiver and the pixel scanning rate at the transmitter, the text would slant up or down, and therefore tend to cut the characters in two. The solution was very simple—the Feld-Hell system transmitted each character once, but the receiver printed it twice, with one image above the other, but using a two-turn inked helix. In this way, if the synchronism was out, the characters could still be read. If the rates were very different, the text would slope uphill or downhill, but still be readable. You can see this in the examples.

There are other newer but less frequently used systems such as Hell-GL and Hell-80, which use synchronism. Hell-GL, developed commercially, uses synchronization bits at the start of each character, while Hell-80 is an amateur adaptation using asynchronous data start and stop bits. These systems are much more prone to errors, and neither of these systems has any advantage over RTTY on HF, although the Europeans use them on VHF. The original Feld-Hell system is today still the favourite for HF.

## Digital Hellschreiber

Modern computer generated Hellschreiber is typically transmitted by keying an audio tone on and off, and controlled by computer. This tone is then fed to an SSB transmitter. For transmission and reception a very simple interface is all that is required. For (sampling theory) reasons that experts will understand, incoming pixels are sampled for display at least twice the pixel rate. The best systems average several samples per pixel (a technique called oversampling) and display the result as varying levels of grey, depending on the confidence of whether the pixel was a dot or not. This makes the text much easier to read because it tends to suppress noise. the following example is from software by Henk PA3BQS.




The transmitter duty cycle is very low. In the drawing in Figure 2, the world "HELL" consists of $7 \times 7 \times 4 = 196$ pixels. Only 44 of these are actually transmitted—22.5 per cent duty cycle. Most SSB transmitters will send forever under these circumstances, as the duty cycle is similar to voice. I estimate the CW duty cycle for 1:1 dot/space weighted characters to be about 40–50 per cent.

While Hell might appear to have very poor throughput compared with other modes, it is outstanding in its most suited application—low power and portable communications using simple equipment on noisy bands, which is exactly what the system was developed for in the first place. Because reading the received text is left to the eye and brain, which are superior pattern detectors, significant numbers of errors are easily tolerated without affecting the readability of the text. In fact, up to about 20 per cent of the pixels can be wrong before the eye and brain refuse to recognize familiar characters.

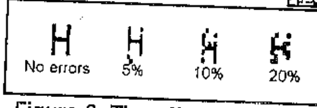


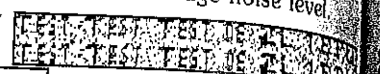

Figure 3. The effect of errors on Hellschreiber.

Within the context of the word and sentence, severe distortion of some letters in a phrase still does not impair the understanding of the phrase. In addition, timing errors in the arrival of individual pixels have limited effect on the readability.

With RTTY even one wrong or delayed data bit will destroy a letter or cause loss of synchronism and then loss of many characters. RTTY must make decisions on each received character in real time, while the reader can take the time to evaluate Hell writing in the context of complete sentences.

It has always been a problem with automatic reception, that the equipment must decide in real time *when* a data bit occurs, and *what* the data bit is. Both of these properties are subject to transmission path errors, and consequently the equipment is prone to making errors. Improvements can be made by techniques such as repeating characters, sending check information, using slow data rates and narrow bandwidth, but in the end the equipment makes wrong decisions. CW and Hell are unique among digital transmission modes in that no decision need be made by the equipment—it is eye, ear and brain which do this.

The eye, ear and brain have excellent powers of pattern detection and error rejection. However, it does take an expert to read noisy Morse code at 30 wpm (equivalent Hellschreiber speed), but any literate person can read Hellschreiber. The following example was received on 80 m using an SSB receiver (2.4 kHz filter) with the signal not moving the S-meter above the average noise level.



Despite the noise, the signal is still very readable, even when received with the simplest software. It is hardly possible to tune in a signal this weak by ear, because it is not possible to distinguish the patterns of the signal in the noise by ear. Fortunately, tuning by eye is easy.

Hellschreiber is also ideally suited to languages based on other character sets—Arabic, Hebrew, even Chinese. The Chinese have apparently been running a higher speed system equivalent to Hellschreiber on HF for many years—even times on our amateur bands. I have recently heard commercial FSK Hellschreiber in the area of 3.4 MHz, but the pixel rate is different to Feld-Hell.

It is possible to operate Feld-Hell under similar conditions and in similar bandwidths to CW. In other words, Hell is one of few simple modes available to the amateur and can be used down into the noise and still be fully readable. Hellschreiber should attract the attention of:

- QRP and portable operators—it's compatible with your new ALIVO transceiver
- MF and low HF operators—avoid the noise, have round table digital QSOs
- LF (170 kHz or lower) enthusiasts—what better way to key that transmitter?

Now that every amateur shack is equipped with a personal computer (yours isn't?) the equipment requirements for operating Hellschreiber are minimal. You probably have everything except the software. All you need is:

- DOS-based PC, preferably 286 or better

# Hellschreiber

## *An old mode still has performance that's hard to beat*

## Part II—getting on air

Murray Greenman ZL1BPU

**PART I described the development of Hellschreiber and gave technical details of this simple but interesting digital mode. Part II of the series describes how *you* can easily get on the air with Hellschreiber—probably with equipment you already possess.**

### Software

There are several programs available, but I would recommend getting started with the simplest to use software, from Sigfus Jonsson LAØBX. This will work with a very simple Hamcomm style modem, and needs no more than a DOS-based PC. Like many communications applications, it will not work in a Windows 95 DOS box, but often does work in full screen mode, although the timing may not be reliable. However, the LAØBX software will work in DOS from Windows 3.1.

The LAØBX program has minimal instructions, but few are needed, and the configuration file is easy to understand and alter to suit your station. Most examples in this article were transmitted and received by the LAØBX software. The software will display multiple lines of Hell text on the screen and allows the operator to send standard Feld-Hell signals and CW ID. The transceiver is fully controlled from the keyboard. The latest version of the LAØBX software, called HS_C9709.zip, is only about 85 k bytes and can be downloaded from the SARTG FTP site,(1), or via the LA9IHA Hellschreiber page.(2) If you find it difficult to obtain the software, send me an SASE containing a PC formatted blank 3.5" disc. Alternatively, email me at coombedn@ihug.co.nz

The software is free for non-profit use.

### Computer

The computer requirements are minimal. You need a PC—and that's about it! Sigfus recommends a 286 or better, and a serial port or parallel port is required, plus CGA display or better. I have used the LAØBX software successfully on the following computers:

- No-name Pentium II 233 MHz with 3 MB Se SVGA
- IBM PS/Note 486/25 laptop with mono VGA
- HP200LX palmtop (10 MHz 80186 processor, 640×320 mono CGA)

The palmtop and an ALIVO transceiver would make the ultimate portable digital station. The HP200LX must run the LAØBX software without the graphical interface—in native DOS mode.

### Connecting up for the LAØBX software

Sigfus intentionally kept his manual as short as possible (to make sure you would read it!), but there are one or two things that should have been mentioned that affect amateurs interested in building their own interfaces. The software can use a standard Hamcomm or JVFAX interface, provides power for the interface from the computer serial port, and expects to use the signals shown in Table 1.

**Table 1.**

| Signal | DB9 | DB25 | Use |
|---|---|---|---|
| DSR | 6 | 6 | Receive data from Hamcomm limiter. |
| RI | 9 | 22 | Alternative Hamcomm input (used by WEFAX and SSTV, etc). |
| TXD | 3 | 2 | Transmit audio (in TXD2 mode), Transmit keying (TXDØ mode). |
| RTS | 7 | 4 | Press-to-talk signal to transceiver, interface power (–9 V). |
| DTR | 4 | 20 | Interface power (+9 V). |

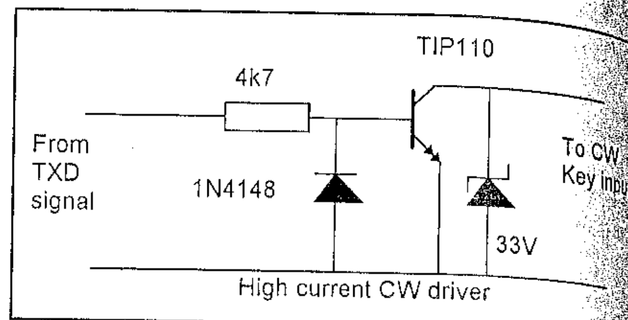



Figure 1.

To use your interface for HFFAX, EXSSTV, Pasokon TV Lite, MSCAN, JVFAX and so on, you need also to connect the receiving limiter output to the R1 signal. LAØBX expects this signal on DSR, so for compatibility, connect it to both pins.

If you plan to use your interface and LAØBX software with an SSB transceiver, no modifications are required. If a CW transmitter is the target, the keying input can be driven by the Hamcomm PTT circuit, by moving the anode of the diode that goes to the base of the PTT transistor from RTS to TXD. Use the VOX in the rig to turn the transmitter on for you. Alternatively, build a copy of the PTT circuit especially for CW, and connect it to the TXD signal. In Figure 1 is a design capable of keying a 1A transmitter. It is possible to key a high-speed relay using the Hamcomm PTT circuit. The pull-in and drop-out times of the relay should be similar or the dots will be distorted; look for pull-in times of 5 ms or better.

To operate the interface with an ALIVO transceiver, you *will* need to provide a high current keying transistor like the Darlington device in Figure 1. You will need to use manual transmit/receive changeover, as the standard ALIVO has no relay power (now there's a good idea!). My elderly five transistor 1W QRP rig has relay control and full break-in, and required no modifications at all to send and receiver Hellschreiber.

### Setting up the LAØBX software

To set up the software on your PC, start by "unzipping" the LAØBX program into a new directory, and then edit the configuration and menu files to suit your installation. Table 2 shows the files to be found in your directory.

**Table 2. LAØBX files**

| | |
|---|---|
| HS.CFG | Station configuration file. |
| HS.ESE | LAØBX DOS executable (the program). |
| HSMAN.ENG | English language manual. |
| MESS | Canned messages: bag messages, station ID, etc. |
| CONTEST.TXT | Information about the annual Hell contest. |
| HSMAN.NOR | You never know when you might need the manual in Norwegian ... |
| LPT_INTF.TXT | The LA9ZO modem and use of the LPT port. |
| MODEM.ENG | Simple LAØBX modem and COM port information in English. |
| MODEM.NOR | Simple LAØBX modem and COM port information in Norwegian. |
| REF.TXT | Bibliography of reference information available about Hellschreiber. |

**Table 3. Software configuration in HS.CFG**

| Parameter | Comment |
|---|---|
| >INT4 | Interrupt for COM1 (use INT3 for COM2). |
| >COM1 | Selected COM port (use COM2 for COM2). |
| >BONW | Black on white display—probably more readable than white on black on most PCs. |
| >CW14 | Sets CW ID speed. |
| >UF1100 | Suitable upper limit of software bandpass filter. |
| >LF800 | Suitable lower limit of software bandpass filter. |
| >NOPI | Detector sensitivity. |
| >SPON | Speaker on. |
| >SPF1000 | Speaker tone frequency (near Tx frequency). |
| >TXD2 | 980 Hz Tx tone on TXD pin. |

You need not worry about the LA∅BX or LA9ZO modems unless you are interested in trying something different—the Hamcomm interface works quite well enough!

The software is configured in the file HS.CFG. Edit this file with a DOS text editor such as EDIT (don't use a word processor). Most of the information in the file is comment. The important lines (recognized by the software) start with a ">" symbol. The list of parameters in Table 3 assumes that you are using the PC's COM1 serial port, an SSB transmitter, and can set your narrow CW filter for use at 980 Hz received tones.

If you plan to use a CW transmitter, change to >TXD∅ and modify your Hamcomm interface as described in the previous chapter (the transmitter won't appreciate trying to key audio tones at 980 Hz), but you may find it difficult to tune a 1715 Hz received signal with your CW filter. This won't matter if you don't mind using the RIT to offset the received signal.

Before you try transmitting, edit the file MESS with a text editor, and add all your standard station information and other memory requirements. I find it wise to leave ALT F8 and ALT F9 blank, as they can be accidentally sent when you really intend to change to CW or receive, but get confused in the heat of the QSO. You cannot use the function keys by embedding them in these messages, unfortunately, although you can embed them in the transmitting keyboard buffer. An experimental version of the LA∅BX software which I am evaluating does provide embedded commands in the MESS file and allows you to set up a "beacon" mode with CW and Hell ID.

The commands F8, F9 and F10 are for use from the keyboard, during a QSO. These keys embed commands in the transmit buffer, so you can (for example) complete your Hell over, then switch to CW to send your call-sign, then switch to receive; for example:

... so, over to you Fred. G∅PFG de ZL1BPU AR K <F8> de ZL1BPU <F9>

The commands ALT F8, ALT F9 and ALT F10 are *immediate* equivalents of these commands. While the other station is transmitting, you can start filling your buffer, and when it is your turn, press ALT F10 to transmit at the start of the buffer. This is just the same as typical RTTY operation.

## Interface

The simple Hamcomm or JVFAX modem (it has one op-amp and one transistor) will allow the software to detect on/off keying of the transmitted signal because the LA∅BX software uses a clever digital signal processing technique. That the LA∅BX Hell program uses the Hamcomm modem is good news, because this same hardware permits operation on RTTY and AMTOR (using Hamcomm software), as well as FAX (using PC HFFAX or JVFAX) and SSTV (using MSCAN, EZSSTV, Pasokon TV Lite or JVFAX). If you operate any of these modes using a Hamcomm interface, you now have no excuse not to run Hellschreiber!

For miniaturization addicts, the Hamcomm interface can

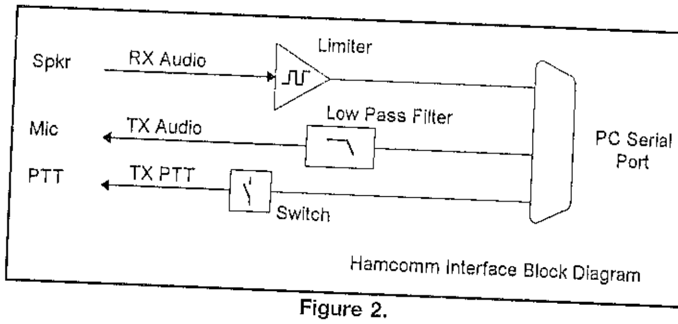



**Figure 2.**

easily be built inside a DB25 connector, but it is equally useful in a small desktop box, which gives you the opportunity to include extra features. A good schematic and excellent layout program for the interface can be found at http://www.accoeeson.com/~tmayhan/schem.htm

My Hamcomm -like interface is built into an existing modem, which already has the power supply, the limiter circuit, some useful band-pass filtering and a very good transmitter low-pass filter. It also has tuning meters, and allows audio to an external speaker to be turned on and off, so you can hear what you are doing, when you need to.

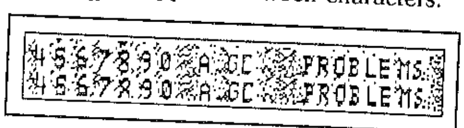
Many other amateur modems, such as RTTY units (Maplin TU1000, BARTG Multyterm, AF4Z Multi-modem and others) have PC compatible signals, and may be modified to provide both receive and transmit operation on Hamcomm compatible programs. Take the output of the slicer to the PC COM port R1 and DSR inputs, and connect the COM port TXD line to the transmit tone filter in place of the existing tone generator, via a switch, so you can still operate RTTY if the need arises.

There is also a fairly simple specialized Hell modem designed by LA∅BX.[3] I have the schematic of some very high performance grey-scale modem designs by PA∅KDF. SASE if you are interested. If you wish to use an SCS PTC-II Pactor controller for high performance Hell without making any modifications to the unit, contact Wilbert ZL2BSJ for information.

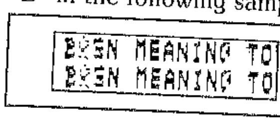
## Receiving Hellschreiber

Connect your Hamcomm modem to the PC and the radio, and after setting up the software, as described above, run the file HS.EXE from DOS.

To adjust the receiver, set the audio gain until occasional noise dots appear on the screen. You will find that very little audio is required. If you can't find any Hellschreiber signals (no surprise!), practise by tuning in a CW signal (CW signals print what looks like bar codes). Hellschreiber signals are easily identified—they have a purr-purr sound like bursts of high speed Morse, and are distinctly different to other digital modes, largely because Hell uses a single tone system. Once a signal is received, tune across it carefully for best (brightest) copy, and if necessary adjust the audio and RF gains for best printing. If the signal is strong, it is best to use very slow AGC or wind back the RF gain to defeat the AGC, which may otherwise cause noise to appear between characters.



On weak signals, experiment with RF and audio gain for best printing. If the signal is subject to rapid fading, you may need to adjust the gain occasionally during an over to achieve best results. Hell is subject to some interesting multi-path and fading distortion effects. For example, it is not uncommon for occasional characters to appear to have their pixels splashed all around where they should be, as they arrive by different paths and therefore at different times. Observe the first "E" in the following sample.



Rapid fades can cause the loss of one or more characters as the AGC catches up.

Observe the loss of two characters, and nearly a third, in the next sample.


The ideal AGC would have about 500 ms attack and 500 ms decay. On 80 m it is normal for copy to deteriorate as the evening progresses, due to multi-path timing distortion. This effect is well-known to RTTY and AMTOR operators. On 40, 30 and 20 m copy is generally much less affected—the biggest problem is that occasionally letters print on top of themselves due to long path and short path reception. Another interesting effect on the higher bands is a "wave" effect, as a weak signal fades in and out over a few seconds.

The performance of your receiving equipment will be considerably enhanced if you use a 500 Hz bandwidth filter. The digital filter in the software has very steep skirts and performs very well, but if you set the bandwidth too narrow, it will make the signals difficult to tune. I operate most of the time in this mode, ie, with SSB filter in the transceiver and 500 Hz wide software filter, because it allows me to listen to adjacent signals. The disadvantage of the software filter or an external audio filter is that a strong signal within the receiver pass-band will pump the AGC and seriously affect copy of a weaker Hell signal. The only answer to this is a narrow IF filter.

Few transceivers allow the CW filter to be used in SSB mode. If yours does not, you will need to switch back and forth from SSB to CW on transmit and receive. There is a simple modification to the Kenwood TS-430S which places the CW filter in the SSB Narrow position—allowing 800 Hz centre frequency to be used on upper sideband and 2.1 kHz centre frequency on lower sideband—ideal for Hellschreiber and RTTY respectively. The filter centre frequency must match the filter in the LA0BX software. Adjusting the tuning becomes critical or the characters run together. If you don't have, or can't use a CW filter, experiment with pass-band tuning.

Because of its narrow bandwidth and on-off keying, it is legitimate to operate Hellschreiber in the CW section of most bands, but in keeping with its "data" designation, it is probably best to keep the data segments, at least on 80 m and 40 m. The European calling frequencies are 3580 kHz and 7035 kHz. There is operation in Europe on Sundays at 1230Z on 80 m, 1530Z on 40 m, and VK2DSG works Europeans at 0200Z on 14061 kHz, although these times and frequencies are changing all the time. A good place to look for local, ZL and VK, Hellschreiber operation is just above the CW segment on 80 m, say 3558 or 3630 kHz. "Look for" is the operative term, because the signals are difficult to identify if they are weak. A weekly Hell net is under discussion and is likely to be 3558 kHz Fridays at 0800–0900Z.

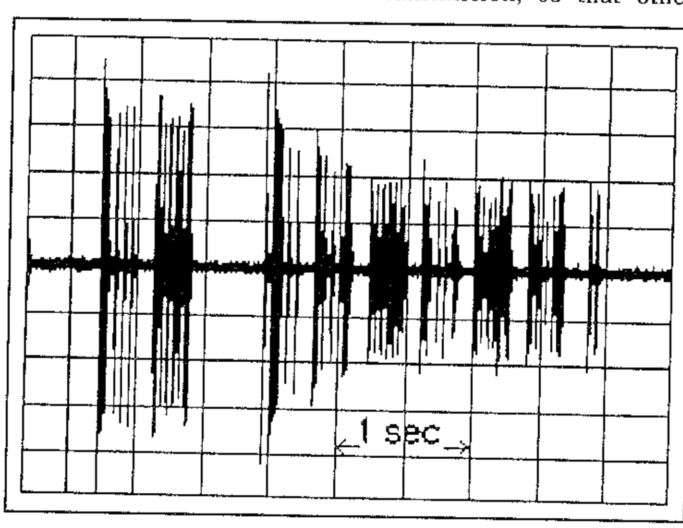
## Transmitting Hellschreiber

Before you transmit, tune up the transmitter with carrier, then make final adjustment of audio gain (or transmitter drive) by sending CW from the LA0BX software. This is important because the duty cycle of Hellschreiber is low and it is tempting to overdrive the transmitter. The oscilloscope picture of the off-air received signal (below) clearly shows the low duty cycle of the transmitter. Two short words (about 10 seconds) are shown, and the signal was 59+ copy at a range of about 40 km.

If you are using a CW or homebrew transmitter, check the keying waveform of the transmitter. If the transmitter uses VOX or is capable of full break-in, it is not so important if the first character after sending starts is lost, but the transmitter must send each following dot with a pulse width of at least 3 ms. There will be delay between the keying signal and the RF output, but the RF signal should be similar in dot pulse to the keying signal. The ARRL-accepted standard of 3 ms rise and fall time for CW transmitters (to avoid key clicks) may be too slow for Hellschreiber—around 1 ms would be better. Some CW transmitters may need modification for high speed transmission, but there should be no such trouble with SSB transmitters. Remember that the keying speed is equivalent to 80 wpm Morse. With the transmitter on dummy load, tune across the signal and check for clicks and excessive bandwidth.

Once you have set the audio level correctly by sending CW, you can change to Hell and, without changing the settings, you are ready to put out a call. When you do, observe that the transmitter average power level is much lower, and the transmitter barely gets warm. If you use a CW transmitter, perhaps an old valve one, tune it up for maximum CW (even get the plates to glow!), and it will be happy sending Hell without distress.

When you call CQ or make a test transmission, always include a CW ID after the transmission, so that other amateurs will know what the "Hellschreiber you are up to" "CQ HELL de ZL1XYZ" "TEST HELL de ZL1XYZ" would be suitable. Preferably use a Voice ID as well, that way you will have maximum opportunity to pass on to others who listen the news about this fascinating old/new mode.




While it is possible to operate Hell with good copy underneath an SSB QSO, it is not something to do on purpose. Choose a clear frequency in the data segment (I suggest 3610–3630 kHz on 80 m). Carriers or CW signals within a few hundred Hertz of the received frequency will seriously effect copy if they are strong enough to capture the limiter. It won't matter if you have local noise on the frequency, but it will make hearing signals difficult. Call for Hell for at least a minute, and remember, when you are not sending characters, nobody will know that you are there! Use conventional CW and RTTY shortcuts and operating procedure. It should be possible to operate full break-in Hell, but I have not yet seen software that supports the ability to listen between words. You software experts—get writing!

## Advanced techniques

A number of amateurs I have contacted, both on air and via email, are using DSP (Digital Signal Processing) techniques to receive and transmit Hell. One good example is a program called EVMHell, written for the Motorola 56002 Evaluation Module by Doug Braun SV/N1OWU. Another is an excellent technique being developed by PA3BQS, which uses a PIC processor to digitize and average the signal at high speed, while taking all the timing problems away from the PC. Both of these systems use grey scale display and proportional transmit fonts. The weak signal performance of these system is better than the LA0BX software. I am hoping for someone to quickly develop a system using the PC sound card to demodulate the received signal.

In a subsequent article I hope to discuss the technical

# Hellschreiber

## *An old mode still has performance that's hard to beat*

## Part III—Signal analysis and digital philosophy

Murray Greenman ZL1BPU

**PART I described the development of Hellschreiber and gave technical details of this simple but interesting digital mode. Part II of the series described how to get on the air with Hellschreiber. Now we will look at some properties of the signal and discuss some important recent developments.**

### Bandwidth

It is widely believed that CW occupies no bandwidth, although this is not actually so since the signal occupies a bandwidth related to the signalling speed. In this respect, Feld-Hell is like high-speed CW, 80 wpm CW in fact. The greatest instantaneous bandwidth requirement is for alternate black and white pixels, at 122.5 bits/sec, or 61.25 Hz, the same as a string of dots at 80 wpm. This causes sidebands spaced 61.25 Hz down from the carrier, and will be about 40 dB down at the tenth sideband. The 30 dB bandwidth (5 S points down on the carrier) will be 612.5 Hz, as shown in Figure 1.

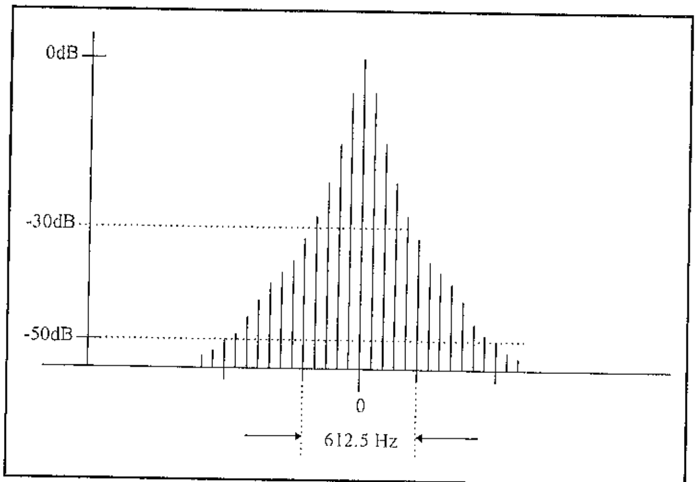



Figure 1.

In a *Break-In* article many years ago,[1] an excellent series of drawings showed the bandwidth occupied by various signals, and showed that a CW signal displays a peaked spectrum with sidebands at the keying rate, just like Figure 5. Most of the other modes portrayed in the article were FSK, and exhibit a flat top or double peaked characteristic.

We can update that article for Hellschreiber related modes. Table 1 shows the 6 dB and 30 dB bandwidth of various common and some uncommon modes.

**Table 1.**

| **Mode** | **4 dB** | **30 dB** |
|---|---|---|
| 32 wpm CW | 24.5 Hz | 245 Hz |
| Feld-Hell | 61.25 Hz | 612.5 Hz |
| MT-Hell (see later) | 100–200 Hz | 200–400 Hz |
| RTTY (50 baud) | 200 Hz | 500 Hz |
| AMTOR (100 baud) | 200 Hz | 700 Hz |
| HF Packet (300 baud) | 400 Hz | 800 Hz |

These days it is relatively easy to measure keying bandwidth using a PC, even if each individual spur cannot be seen due to lack of spectrum analyzer resolution. The spectrum analyzer I use is Spectrograph 4.2.2a, which uses the PC Sound Card for input.[2] Figure 2 is a 256 point FFT (Fast Fourier Transform), averaged over 32 measurements to smooth the noise, and shows a 32 wpm string of CW dots.

The spectrum analyzer was fed from a Collins 32S-1 receiver on LSB with a 2.4 kHz filter, via a transformer-coupled high-pass filter with 300 Hz cut-off to reduce the hum. The receiver RF gain was backed off to disable the AGC and limit background noise. The transmitter was a homebrew, VFO-controlled, QRP rig on dummy load at 3.6 MHz. The equipment was adjusted to 0 dB on a continuous carrier prior to high speed keying ...

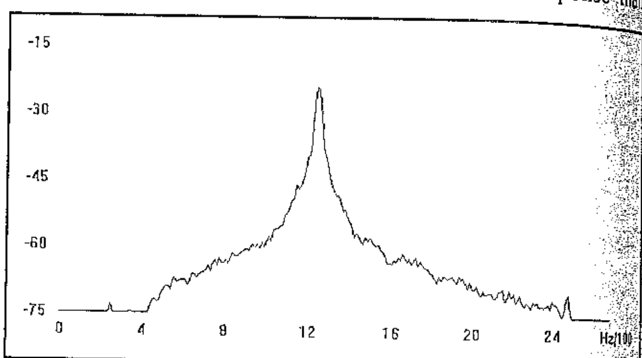



Figure 2.

Various inter-modulation problems in the receiver, and limits to the dynamic range of the sound card and FFT function means that the classic shape is lost at around 30 dB below the peak. Anyway, the result (Figure 1) is sufficient to show that CW certainly has bandwidth!

Using the same setup to assess Hellschreiber showed that the carrier peak is lower, because of reduced duty cycle. The classic shape is retained, but the signal is predictably wider, since the signalling rate is much higher.

### Key clicks

Chirp and keying rate are not the only factors that influence signal bandwidth. Key clicks are generated whenever the key is pressed or lifted; to transmit CW or Feld-Hell with data elements as short as 8 ms, the transmitter keying speed must be high. Most commercial transmitters have softened keying edges to minimize clicks in CW mode, and provide creditably clean CW, but may not be suitable for transmitting Hellschreiber. The ARRL recommend 3 ms rise time and fall time for normal speed CW which is a significant portion of the Hell pixel time! Slower response than 3 ms is not appropriate for Feld-Hell, because, as the Figure 3 shows, the effective transmitted pulse is significantly shortened, and reduced in height, markedly reducing the readability of the signal.

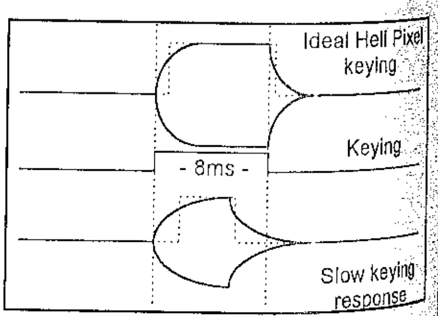



Figure 3.

The best way to get around this is to use an SSB transmitter, partly because the rise time of the transmitter is faster (under 1 ms) and partly because the key clicks (which will be generated) can be limited to a 2.4 kHz bandpass. This is how the LAØBX software operates. Before you operate Feld-Hell with your transmitter, it is a good idea to check or adjust its keying waveform with an oscilloscope, comparing the transmitter with Figure 3.

There is a cunning technique employed in the traditional mechanical Feld-Hell machine which limits the minimum pulse width to 8 ms while providing 4 ms resolution! I omitted this from the description in Part I to limit the reader's confusion. The font is designed so that the pulses have 4 ms resolution, rather than 8 ms, for improved character shape, but the 4 ms pulses are never transmitted alone, for example on the right of a "B" there are two 12 ms pulses (each 1.5 dots). The purpose of this restriction is not only to reduce key clicks, it also restricts the minimum size of pulse that the mechanical reception equipment need respond to, and helps the transmitter generate reasonable pulses despite slow keying.

## The waterfall plot

The spectral display (for example, in Figure 2), is an excellent tool which displays signal strength vertically, and frequency horizontally. However, it does not tell the whole story, as well as clicks and chirps, there are many things that are hard to see with a conventional spectral display. Now is the time to roll out the secret weapon, the *waterfall display*, which portrays signal frequency vertically, time horizontally, and signal strength in brightness or colour. The PC sound card waterfall plot or real-time *spectrogram* is arguably the most important Amateur Radio test tool developed since the Grid Dip Meter. With a waterfall display, FSK keying, key clicks, chirp, hum and distortion are easily demonstrated. Figure 4 is a waterfall plot display, using GRAM,[2] and the display is of an off-air signal from ZL2BSJ in Wellington. It represents about 13 seconds of a Feld-Hell transmission (horizontal scale) and 0 to 2576 Hz (vertical scale). The calibration marks have been omitted for clarity.

Without intending to criticize the equipment in use, this plot illustrates some features of the signal, and the equipment used to send it and capture it.

- ★ The CW-like appearance of the signal (1400 Hz single tone) is obvious as a dark band across the middle of the picture. It is slightly furry due to key clicks.
- ★ The key-down signal is fairly clean, with no noticeable chirp, no drift and no hum on the 1400 Hz tones. Each of the small black bands is in fact a series of dozens of dots, not discernible at this sampling rate (hence the visible sidebands).
- ★ The transmitter has some hum and buzz at 400 and 500 Hz (mains harmonics) seen as lines along the bottom of the screen. The bottom two lines, 100 Hz and 200 Hz, are from the receiver, and since they are not accompanied by background noise, you can see that they are outside the receiver IF pass-band.
- ★ When the transmitter is silent between words, the receiver AGC winds up and additional buzz on the signal (which was S9+60 dB with key down) can be seen throughout the audio as a "ladder" of lines between segments of the signal. The probable cause is hum in the audio circuitry of the transmitter.

GRAM has many uses, and was well described in a recent *Break-In* article by Gary ZL1AN.[1] We will meet GRAM and waterfall plots again when discussing multi-tone Hellschreiber.

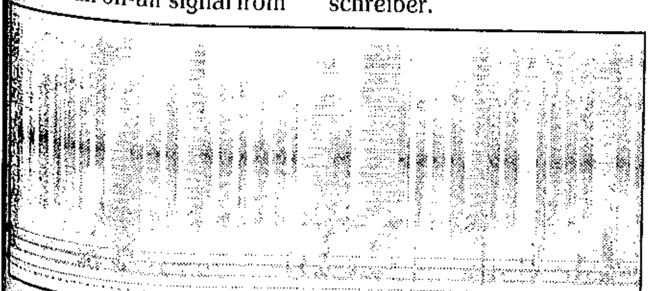


Figure 4.

## Decisions, decisions

In Part I of this series I touched on the problem that is common to all digital modes; *when* is there a data bit, and *what* is it? To refresh your memory, I quote:

> "It has always been a problem with automatic reception, that the equipment must decide in real time *when* a data bit occurs, and *what* the data bit is. Both of these properties are subject to transmission path errors, and consequently the equipment is prone to making errors."

A good digital communications system should either:

- ★ make highly accurate decision (TYPE A), or
- ★ avoid making decisions at all! (TYPE B).

In a noisy HF environment, particularly where simple equipment is an important factor, the latter approach—I call it TYPE B—is a much simpler approach to take, and quite appropriate for casual QSOs and nets.

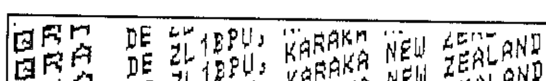
This decision problem applies to all digital modes, and most modes attempt to take the first approach. Making highly accurate decisions is important where the data transmission controls equipment without human operators, for example operating radio bulletin boards or repeater controllers. However, for person-to-person random QSOs, the alternative (no decisions) approach is entirely appropriate and much simpler. Of course, CW addicts have known this for years. It is also recognized that error correction and detection processes occupy a significant porportion of useful transmission bandwidth that could well be used to provide redundancy to enhance the noise tolerance of the radio circuit, and so avoid needing error correction! CW and Hell are unique among digital modes in taking the TYPE B approach, by avoiding making decisions in electronics. FAX and SSTV have some similar properties, but are considered to be analogue modes.

Decision making has two facets. *When* refers to synchronism—deciding *when* there is a data bit, which is no simple matter when the radio path length can vary by as much as the length of many bits. *What* refers to whether the information in the radio channel at the time a decision needs to be made is data 1, data 0, or noise. Almost all data systems are *synchronous* or *asynchronous*. In synchronous systems the data is sampled at fixed times relative to a reference (which may be transmitted ahead of the data, as it is in Packet radio, or at the start of a frame, as it is in AMTOR and PACTOR). In an asynchronous system, data is sampled at fixed times relative to the start of the data word (as it is in RTTY). All of these modes experience the *what* and *when* problem, because each data bit transmitted is subject to the vagaries of the ionosphere. For human readability, a *non-synchronous* mode would be ideal.

Feld-Hell is a step in this direction. First of all, Feld-Hell is uniquely *quasi-synchronous*. This means that the receiving equipment should try to be in step with the sending equipment, but the information is not lost if it isn't quite in step. Rudolf Hell achieved this with two clever mechanisms: printing data when it arrives, and providing immunity to long term timing errors.

First, the pixel is printed *when it arrives*, so it isn't sampled as such. Both mechanical RTTY machines and computers used for RTTY have to sample the incoming data bit at a fixed time (typically in the middle of the expected bit), whereas the old Feld-Hell machine simply prints a dot *when* it gets the data—even if it is late or early. No time decision is made. Secondly, the timing is relatively unimportant, because if the equipment gets out of step, the effect is to cause the characters to move up or down, or ultimately, to slant up or down.

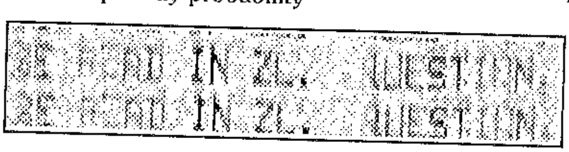
In the example below, the timing is out by more than two per cent. If individual bits of



data arrives seriously out of time, which they will if the propagation path changes due to ionospheric multi-path effects, pixels will be scattered, the quasi-synchronous nature of Feld-Hell causing the print to be not easily readable. This is what happened to the "L" in my call-sign in the next example.


A mechanism related to the *what* process, which should be employed by Feld-Hell, is the use of "fuzzy" logic. In other words pass on information to the reader based on the *likelihood* of there being a data bit, rather than make a firm yes/no decision. This technique is used in multi-tone and greyscale versions of Hell. Probability, over-sampling and averaging are the techniques which can make the most spectacular improvement to reception. The following samples were both generated from a tape recording of OH/DK4ZC transmitting Feld-Hell on 20 m. First, as received by the LA0BX software, with its fixed bit decisions and quasi-synchronous timing:

Second, a sample made by Rob ZL2AKM by post-processing the data from the same recording, using over-sampling, averaging, and greyscale to portray probability




The audio source and text in the two samples is exactly the same, but the advantages of fuzzy decisions are immediately obvious. The text goes from unreadable to completely readable. This sample shows all the features of a good TYPE B system; it leaves *all* the decisions to the reader's eye and brain. No timing decisions are made (because of over-sampling), and no data bit decisions are made, because of fuzzy (greyscale) presentation. The equipment is simply reproducing a visible form of what arrived on the radio channel.

The advanced Feld-Hell software methods I have encountered, G3PLX's EVMHell, G3PPT's Feldhell and PA3BQS's PIC system, all use fuzzy decisions and over-sampling. These techniques are used together to display pixels of varying brightness, based on the confidence of there being a pixel present in each of a number of samples. The averaging used to display each pixel significantly counters the effects of noise—remember that when averaged, data samples tend to add, whereas noise samples tend to zero.

Perhaps it is time to summarize what we can achieve using digital equipment to decode Hellschreiber (see Table 2).

**Table 2.**

| Technique | Mechanical Feld-Hell | Simple Program (LA0BX) | DSP Software (EVMHell) | MT-Hell Frequency Domain |
|---|---|---|---|---|
| Ionospheric timing immunity | No | No | No | Yes |
| QRM | Good | Good | Very good | Excellent |
| Synchronism required | Quasi | Quasi | Quasi | None |
| Show bit when it arrives | Yes | Maybe | Yes | Yes |
| Use fuzzy logic | No | No | Yes | Yes |
| Over-sample | No | No | Yes | Yes |

In a couple of places in this article, you will have spotted references to "MT-HELL". This is the secret weapon of the Hellschreiber world—Multi-tone Hellschreiber. Now that you have been introduced to the spectrogram, the next issue will discuss MT-Hell, sending uncoded text in the frequency domain and the advantages this has. New Zealand now has a score of Hell users, and several already operate frequency domain Hell!

If you are interested in experimenting with DSP, want to talk digital, have any questions, or want to hear MT-Hell, send me a note. I recommend that you join the irregular net on 3560 LSB at around 0830–0930z week nights. Friday night is the ZL Hell Raisers' net, and a good opportunity for a Hell contact with VK. All the software you will ever need for Hell is available on my web site, or failing that, by sending me a floppy disk and an SASE.

For more information visit my web site at:

http://members.xoom.com/zl1bpu/contents.html

## References


1. "HF Radio Data Communication", Bill Henry and Ray Petit, *Break-In*, March 1993.

2. Spectrogram 4.2.2a for Winn95 or WinNT, is Freeware by R. S. Horne rshorne@mnsinc.com and is available for download at http://www.monumental.com/rshorne/gram.html Highly recommended for any radio signal analysis.

3. "The Morseman", Gary Bold ZL1AN, *Break-In*, September 1998, page 14.

# Hellschreiber

## *An old mode still has performance that's hard to beat*

## Part IV—Frequency domain

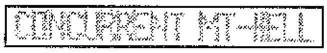
Murray Greenman ZL1BPU

**PART I described the development of Hellschreiber and gave technical details of this simple but interesting digital mode. Part II of the series described how to get on the air with Hellschreiber. Part III dealt with signal analysis and digital philosophy. Now we will discuss some important recent developments.**

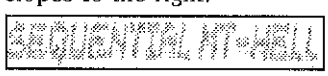
At the risk of being suspected of an "April Fool's" joke, I would like to conclude this article with an introduction to sending text in the frequency domain. Up until recently, all Amateur transmissions, including Hellschreiber, took place in the time domain. Even if different frequencies were used (FSK), the transmissions remained in the time domain, and were decoded in the time domain. However, it is quite possible to transmit signals in the frequency domain. Digital HF users may be aware of frequency domain modes such as Piccolo, and there are phase domain examples as well (PSK31, Clover). These phase and frequency modes are (according to my definition) TYPE A communications systems. A few amateurs, G3PLX, G3PPT and myself) are working with sending text as a frequency domain TYPE B transmission. The aim is to free the text transmission from *all* time domain-related effects, of course including interference and most importantly, freedom from ionospheric propagation timing distortion, which affects Feld-Hell and has such a disastrous effect on RTTY, even when signals are strong.

The secret to analyzing text signals in the frequency domain is good Fast Fourier Transform software, with a display of frequency against time—the waterfall display again! Frequency domain transmission can be likened to the chirping of birds or the noises of whales—they make no sense to humans, but when displayed on a spectrograph, which displays frequency against time, recognizable patterns appear. The same applies to sending text as tones. If you have a spectrogram program, try whistling different notes, with some practise, you might be able to whistle your name! (Jack is difficult, but Jill is easy.)

This is exactly the technique used to generate frequency domain text, which we call Multi-Tone Hellschreiber, or MT-Hell, because of its historical association with Feld-Hell. The MT-Hell signals are generated as a series of chirps, and may contain more than one tone (frequency) at a time. There is no amplitude information, except that when there is no tone, there is no dot. We have evolved two systems—concurrent C/MT-Hell, and sequential tone S/MT-Hell, which are largely compatible with the same receiving software. Most of the work so far has been done using sophisticated DSP hardware and up to 16 concurrent tones. I have been working with seven sequential tones, a system I call MOSIAC II, which sends text using a simple DOS PC program. I use GRAM and a PC sound card to receive.

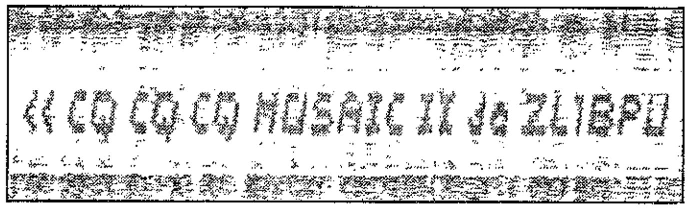



This example shows C/MT-Hell, generated using sound card software by G3PPT. It has upright characters and sharp, thin text. The next sample is S/MT-Hell generated by my MOSIAC II, and has slightly furry, fatter text, which slopes to the right.




Both systems use seven tones with the same tone spacing, and were recorded using the same GRAM settings. The images have not been enhanced, enlarged or touched up in any way. Some furriness and noise surrounding the text in the above examples can be attributed to keying rate sidebands and clicks. There are also mathematical shortcomings in the FFT technique which cause similar effects. With a fast enough PC, and the right windowing technique, these artefacts can, to a large extent, be controlled by choosing the correct settings.

Most received noise is in the amplitude domain, which contributes significantly to the noise rejection properties of frequency domain reception. In addition, in-band interference such as other signals, splatter, and lightning crashes have very little effect on the readability of MT-Hell text. I have demonstrated this by sending a QRP MT-Hell signal between the tones of an amateur RTTY news bulletin transmission on 80 m—with good readability! The following picture demonstrates this. The ragged black marks top and bottom are the 170 Hz shift RTTY signal, and the MT-Hell text in the centre is clearly visible. With the ear, all that could be heard was the RTTY station. The MOSIAC II transmission is a mere 50 Hz wide, transmitted at 10 pixels per second, or about one character per second.




Because MT-Hell operates in the frequency domain, there is no notion of synchronism, no need to match transmit and receive tone frequencies or even pixel rates at each end of the communications path. Thus the timing requirements are very relaxed, as can be seen in the following example, where two completely different transmission techniques, at different speeds, are decoded at the same time!

Relaxed timing also frees MT-Hell from time-of-arrival errors caused by the ionosphere. Even variations in timing caused by the computer operating system has negligible effect.

It has been found that the pixel tone spacing (Hz) should approximately match the pixel rate (pixels/sec) for the best compromise of readability. This places the first keying sideband under the adjacent pixel and minimizes the pixel structure. (Helps make the characters look smooth rather than dotty.) This approximate one-to-0ne aspect ratio is really all that defines the timing of the characters.

Unlike Feld—Hell (amplitude keyed Hellschreiber), there is no need to display each character twice to combat bad character synchronism—because there is no concept of synchronism! MT-Hell is truly *non-synchronous* (look back at the table of properties). Tuning is uncritical, and although the transmitted signal is generally less than 200 Hz wide, there is no need for a narrow filter. Interfering signals have minimal effect on copy, unless they are that much stronger than the wanted signal that the receive gain is affected by AGC action.