SSTV

An Inexpensive System Continues to Grow

A good system gets better...

By Ben Vester, K3BC
4921 Bonnie Branch Rd
Ellicott City, MD 21043
Photos by the author

My software-based SSTV system[1] received a fairly good reception, and a few weeks after its publication there were pictures showing up on 14,230 kHz with the program's characteristic red (or green) "shades of gray" stripe at the top. Figure 1 shows some of these as copied at this location. One advantage of having a sizable body of users is the feedback you get related to problems experienced and additional features desired. A pleasant surprise is the substantial number of people who are now using SSTV on the VHF bands and have organized weekly nets to trade pictures. This move has added a new set of user priorities different from that of the old HF SSTV gang. As this is written, the software is up to revision F, each revision being a step toward filling users' requests. Some of these revisions were described,[2] but a number of them need pictorial support, so they're described here.

Program Structure

The BASIC programming built around a series of machine-language modules (which do all the high-speed work) has proved to be even more flexible than I initially realized. Its greatest virtue is that it allows many users to modify the programs to fit their own operating preferences, but it has also simplified my task of adding features.

One user-instigated feature is the menu screen shown in Figure 2. Dennis, KY1S, decided he wanted a more attractive menu and went to considerable effort to make an attractive personalized screen with icons, using some heavy BASIC programming. Believing other SSTVers would like to do the same thing—but probably wouldn't have Dennis' programming skill—I modified the original *RT* program so it accepts any user-generated SSTV picture file as a menu screen. Now you can feature your favorite landscape, girlfriend—or whatever—on the menu screen. If (or when) you get tired of it, you can change it easily. The *LABEL* program adds the menu words to the picture in a few minutes. This is but one example of a user-initiated addition that I would never have thought of! There are many more described in the oversize system manual file, *TVINFO.TXT*. Please—*read it!*

3-D Pictures

Many of the additions were obvious, such as adding the vertical-interval signal (VIS) code detector, and then expanding to add automatic picture saving (SAVE) so you can leave your computer unattended to detect and save the pictures received while you're away. Noting that a number of people were sending 3-D—or in many cases, *synthetic* 3-D pictures—I took a look at doing that. I was amazed to find that the machine-language module in the *LABEL* program is sufficiently general purpose that with a few judicious POKEs, it generates 3-D pictures, both the synthetic type and those composed of two pictures of the same scene displaced by eye separation. Although no one had asked for this feature, it was too easy not to include it! It's most useful when you're using a TV camera and frame grabber to generate images. You merely feed in a "right eye" and a "left eye" picture and the program outputs a 3-D image. Synthetic generation is probably too

[1]Notes appear on page 24.

Figure 1—Early examples of the Vester SSTV system. The top two pictures are by WB7PAP and the dog by N9EJJ.

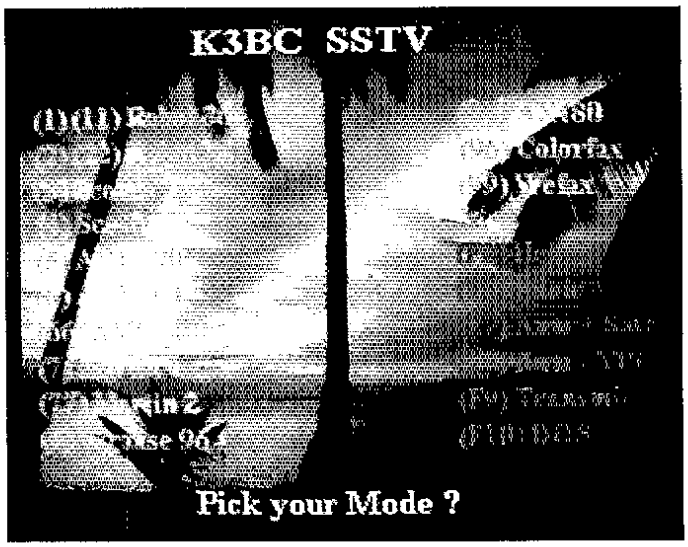

Figure 2—With the latest version of the software, you can customize the Mode menu of your copy of the software.

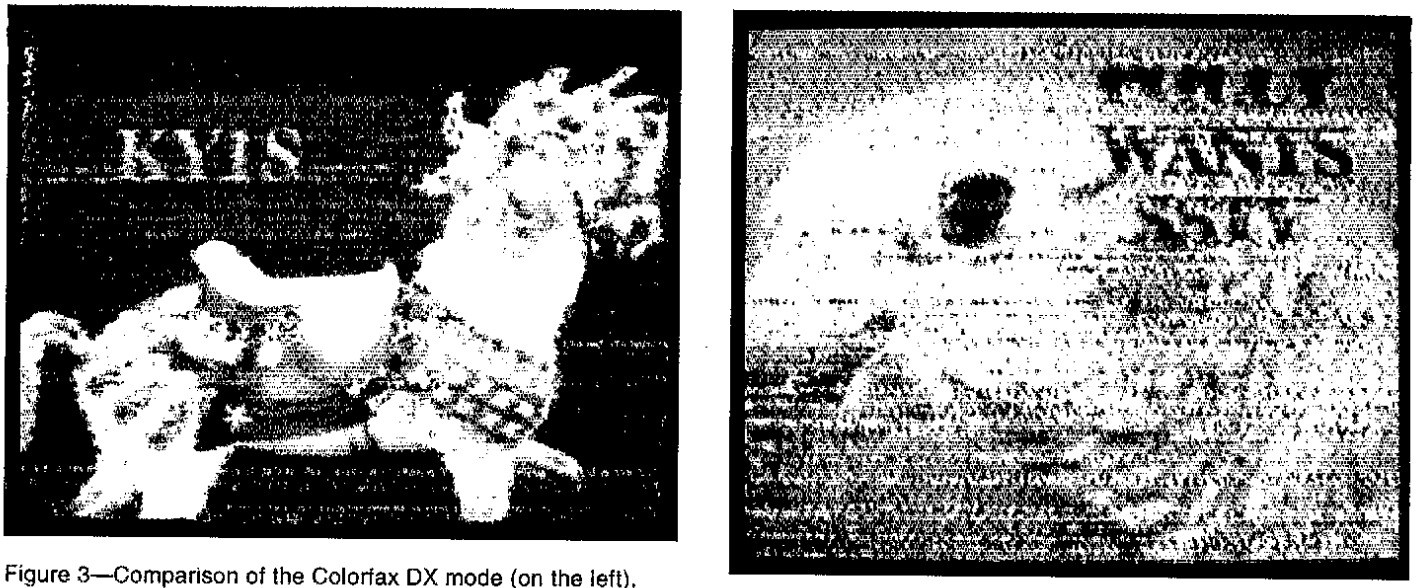

Figure 3—Comparison of the Colorfax DX mode (on the left), with a Scottie 1 mode picture (on the right) received under severe QRM (see the text).

laborious for most, but can be done with this program. Typically, you'll synthetically insert your call sign to be at the picture depth you choose.

File-Handling Convenience

The picture files are rather large, and if not compressed, soon fill up your available disk space. There are also a number of the associated programs that generate picture files that you may choose to save. Each of these has a SAVE routine added now so you can save pictures in full or compressed format right where it is generated. This is particularly convenient when you're using *SLIDESHO* or *TRUSHO* to thumb through a bunch of pictures collected while you were gone. The transmit program, *VT*, is now configured to accept files in full or compressed formats.

Alignment

Because the frequency tolerances of computer clocks vary, line-timing calibration is necessary. In revision A, released January 30, 1994, I included a simple arithmetic formula that allowed users to calculate the line timing for all of the modes once they'd experimentally determined the line timing for any single mode. Months later, there were still people on 14,230 kHz talking about the labor involved in aligning all the modes. Since some users wouldn't (or couldn't) do the simple arithmetic, I created *LINETIME* to do it for them. You crank in a corrected value for Scottie 1 mode and the program spits out all the line timings (LTs) required for the other modes.

Trucolor

More recent additions have taken advantage of the new Trucolor video cards that offer 640×480-pixel resolution with 16.7 million colors and are available for as little as $70! Such cards not only give you much brighter pictures than the 256-color boards, but the increased information content per screen allows you to display *four* SSTV pictures per screen at full resolution and color content. This not only makes for more interesting slide shows, but offers a test vehicle for side-by-side comparison of transmitted files with the original file for calibration of the transmit program.

Also, as described under **New Modes**, such cards make it possible to display higher-resolution color pictures in full color. When the normal SSTV pictures (bounded by 320×240-pixel resolution) are expanded to fill the Trucolor screen, the "square-pixel" effect becomes visible at normal viewing distances. To ameliorate this, we've used fractal techniques in the most recent versions of *TRUSHO*. Although this improves the picture quality, *it can't improve the intrinsic resolution of the picture.* This can *only be done* by using higher-resolution modes to transmit the images.

New Modes

I considered adding the Scottie DX mode, but after studying it, decided it didn't use the redundant pixels as efficiently as it might. Basically, Scottie DX sends the same pixel-byte value three times in a row and then combines these three for a more noise-free single-pixel byte. This is great for reducing white noise, which is not correlated in the three samples. But you only need look at a few pictures to see that many of the noises which pollute our pixel values (QRM, QRN, multipath propagation, carriers, etc) are often correlated over three successive pixels, or more.

So I began looking at inserting the redundancy with time separation between samples—like repeating the same line three times in a row. Each pixel byte would have three samples separated by the time of one line length. The good stability of the crystal-locked receiving system allows combining time-separated pixels quite precisely (systems using re-sync on each line would not combine so precisely!). This would decorrelate the samples against more than white noise and give better results.

The more I thought about it, the more foolish it seemed to use completely redundant pixel bytes. Four pixel bytes grouped together in a square (ie, two adjacent in line 1 and the two directly below these in line 2) could be sent with exactly the same value and give a redundancy improvement against noise in preserving that value. The same four pixel bytes could be used to improve the spatial resolution by a factor of two in each axis. If conditions are good, you see a resolution improvement of *four!* If conditions are bad, your eye averages those same pixels and gets at least as much improvement as if the pixels were all sent with the same value.

Being time separated, the two groups of two give further improvement over the Scottie DX mode. So, if you are going to send more bytes to improve poor-condition reception, it's better to use the bytes for better resolution and let your eyes do the averaging. Signal strength varies and you naturally get the better resolution when the signal strength is high. Also, because most SSTV gatherings have several stations at different distances copying the signal, some get extraordinarily good, high-resolution pictures; stations receiving weaker signals getting less. This process is illustrated daily when people switch modes from Martin 2 to Martin 1. Here, the double resolution is in one axis only, but the difference as to how fast each mode deteriorates under worsening conditions is well known.

I decided to make a full-color DX mode with double resolution in *both* axes. To avoid start-up problems getting everybody locked to the same line timing, I chose to use the same timing as Ralph (WB8DQT) Taggart's FAX480 system.[3] Three successive 546-byte "lines" are used for the red, green, and blue data bytes that form the first full picture line. This is close to a double-size Scottie 1—double in both axes. The Australian street scene in the Up Front section (page 13) shows the typical picture resolution. This double-resolution mode I call Colorfax, and it uses a VIS code of &H6A (the ampersand H

denotes hexadecimal notation).

For evaluating the mode, I enlisted the help of KY1S. To get a relatively constant level of interference and noise, both stations zero-beat the same 40-meter broadcast station. Then, KY1S sent me a Colorfax picture followed immediately by a Scottie 1. The resulting pictures are shown in Figure 3. It's worth mentioning that because of the strong broadcast-station signal, we had trouble communicating verbally during this test. Clearly, a mode that consumes this much time per picture (over 6 minutes) should be used only sparingly on busy frequencies such as 14,230 kHz. With the growing army of slow-scanners, the use of less-busy frequencies is becoming popular, however. And for many VHF bands, Colorfax fills the need for a really high-resolution, full-color mode. It should turn out to be an excellent DX mode.

I replaced the long acquisition header of FAX480 with a short VIS code (&HAA) to make it compatible with all the other SSTV formats. This makes both black and white and high-resolution color available with automatic VIS. The picture files for the B&W fax are interchangeable with FAX480 and can be transmitted with either the VIS or the FAX480 header. This B&W fax mode uses 64 shades of gray on both transmit and receive, unlike the 16 gray shades of FAX480. The format-conversion programs allow generation of pictures in all of these formats from *any* color image.

Format Conversion

Although the original programs were aimed at using a TV camera and frame-grabber for generating most of the picture files, I found that a number of people are using widely available GIF files for pictures. These pictures have only 256 colors, but often do a good job of matching the colors used to the picture color content. Users have located several shareware programs that convert GIF to TIFF files to plug into *TIFCONV* for SSTV conversion, but the procedure is time-consuming. Recently, KY1S located a shareware command-line format-conversion program called *Alchemy*, which is ideally suited to chaining with other programs. With one DOS line command, *Alchemy* converts images from a number of possible formats to any one of another set of formats. Using a simple batch file to chain together the *Alchemy*, *TIFCONV*, and *VT* (transmit) programs, you now can simply type the batch-file name, *TR*, followed by the path and file name for any GIF, PCX, or JPG file and—except for announcing which mode you're going to transmit—no further action on your part is required except to press the **G** key to start transmitting!

CD-ROM Pictures

After surveying the CD ROMs available at the Dayton HamVention this year, I found a number of 24-bit Trucolor pictures available in TGA format. To convert this new source of high-quality images, I created another conversion program, *TGACONV*. Using batch-file chaining of *TGACONV* with the transmit program allows *direct transmission of pictures from the CD ROM in whatever mode you choose for each picture.* Since the PATH, a portion of each file name and its suffix are common to all pictures, you need only type **TE 2905** to get the file E:\TGA\IMG2905.TGA into the transmit buffer, visible on the transmit screen and ready for **G**(o). The manual, *TVINFO.TXT*, gives you enough information to allow you to create your own minimum-keystroke batch files like this.

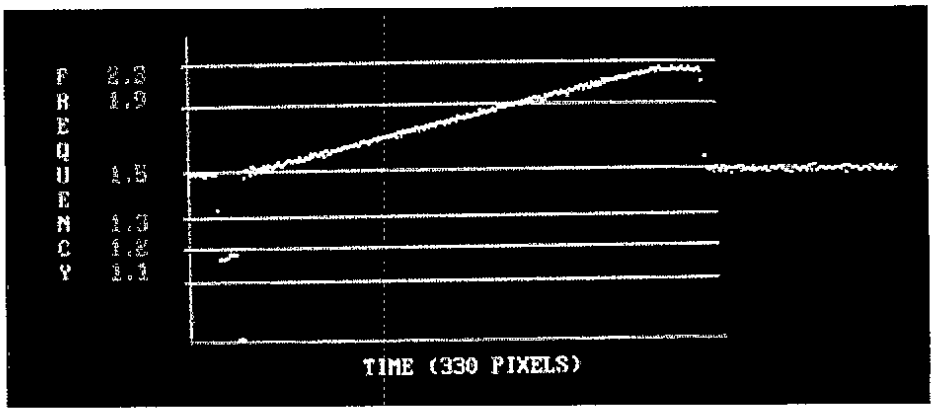

Figure 4—The frequency versus time display is used to look at incoming signals. Here you can see that the receiver is mistuned about 15 to 20 Hz from the transmitted (incoming) frequency. If perfectly tuned, the sync pulse would be located at the 1200-Hz mark instead of at 1180 Hz as shown. This frequency difference is small enough to still receive pictures well. This display is from the gray-shades portion of the incoming picture.

Real Time

One of the most-requested features was a line-by-line paint of the picture as it came in. This is easy to do with a line-synched system (which has more jitter than the crystal-locked system) and where you don't want to eliminate the picture gap, which occurs on the occasional picture that is copied out of sync. Cramming the screen paint at the end of each line just is too large a gap. Interleaving the screen paint with the measurement of each pixel byte's frequency can be done, but it requires a fast computer. Since the population of 80286 computer users seems to be decreasing rapidly, I put this in as a switchable option for the people with faster machines.

The newer Trucolor video cards don't require VESA drivers, are faster, and require a minimum of code to write to, so they work better. I also added a real-time option for 256-color boards. One negative aspect of this is I no longer manage to read my daily newspaper while pictures are coming in!

Tuning Aids

I resisted adding any frequency displays to the software for some time since the *HAMCOM* software is freely available, can use the same interface as our programs and has an excellent spectrum-analyzer display that can serve this purpose. The display I added differs in that it is a frequency versus time presentation. It displays in sequence the frequency of every byte sample received and looks much like the oscilloscope display you get looking at a TV video signal. With the same scroll routines as used in picture viewing, you can scroll the frequency-versus-time viewing window through a complete picture file. Figure 4 shows a typical shades-of-gray picture line. This display is useful for looking at VIS codes, AVT and FAX480 headers and more.

As another aid to get everyone on the same operating frequency, I added a pulsing 1200-Hz tone to the transmit program, *VT*. This tone can be keyed on when needed by holding down the T(one) key.

Comment Line

In most SSTV modes, there are 16 picture lines devoted to shades of gray, and many systems overlay your call sign here. In the latest revision, I added the ability to include a short comment in this area on each picture as it is transmitted. Or, you can default to your call sign and/or name, or whatever you choose to put in the SYSTEM CONFIGURATION list on any particular day. You can have a little fun with the Robot users, since (I'm told) that system doesn't display these lines—although the lines often do show up on the Robot's *next transmission.*

Summary

"I haven't had so much fun in years!" is by far the most common comment I've received about the system. And, I've heard at least one young man describe the system as "not user-friendly." Both comments apply. They're the natural reactions to the software's "sweat factor" that is *designed in specifically* to get some experimental juices flowing again!

The latest software revision is available on the ARRL BBS: 203-666-0578 as VESTER_F.ZIP. A number of other BBSs—and Internet—have picked it up, so you may find it on your local BBS. Let's see you on the air!

Notes

[1]B. Vester, "An Inexpensive SSTV System," *QST*, Jan 1994, pp 27-29.
[2]B. Vester, "Vester SSTV/FAX480/Fax System Upgrades," Technical Correspondence, *QST*, Jun 1994, pp 77-78.
[3]R. Taggart, "A New Standard for Amateur Radio Facsimile," *QST*, Feb 1993, pp 31-36.