Here's a unique program from SV3ORA to permit hams to easily draw schematics and other diagrams and then exchange them over the air in virtually any transmission mode.

Schematix

BY KONSTANTINOS GIANNOPOLOUS,* SV3ORA

During my HF RF experiments, I usually find myself in situations where I need to draw my designed schematics. In fact, at the time I develop my circuits, I draw them block-by-block and then I test each one. If a block fails, I discard it; if it works satisfactorily, I keep it and move ahead.

Schematix is a program I have developed to help me in this aspect of the ham radio hobby[1]. I could use freely available CAD (computer-aided design) programs, but most of them, while quite capable, are also quite complex to learn. My intention was to produce software that would be straightforward to use, even for a child with basic electronics knowledge. I mostly thought about older hams, though, who quite often find it very difficult to learn how to use a complex CAD program. Complexity is a disadvantage if you want to do a simple thing.

When the basic drawing part of the program had been written, I thought it would be nice if one could use this program to send his schematics to his ham friends through radio around the world. So, a quite complex mechanism for being able to do this simply was developed.

The "Schematix" software is intended for use by radio amateurs to draw simple schematics, PCB layouts, and diagrams quickly and easily, and to efficiently send these to other hams via Morse code, voice using a phonetic alphabet, or any text mode.

Apart from drawing, there is a variety of ways in which these schematics can be exported or imported. Schematix has been designed with simplicity in mind, so anyone with basic electronics knowledge can immediately use it, without much effort or documentation reading.

Schematix is written in HTML and Javascript, which has several advantages:

- It can run live from a webpage, without any need to download, or as an offline standalone application when downloaded to your computer.
- It is multi-platform, meaning that it can run on any operating system (Windows, Linux, etc.) as long as there is a Javascript-enabled browser installed (Firefox preferred).
- It is widely open-source, since the code is immediately available to anyone. Any distribution of the application automatically distributes the source code, since it is the actual source that is distributed, no binaries, no executables, no installation to your PC.
- The application can be modified easily using a simple text editor, without the need for special installed tools or compilers on your computer.

Radio amateurs are encouraged to take the code, modify it, and create something even more useful to the ham community. The way Schematix operates, is not limited to schematics, so simple PCB drawings and diagrams can be added.

Why Use Schematix?

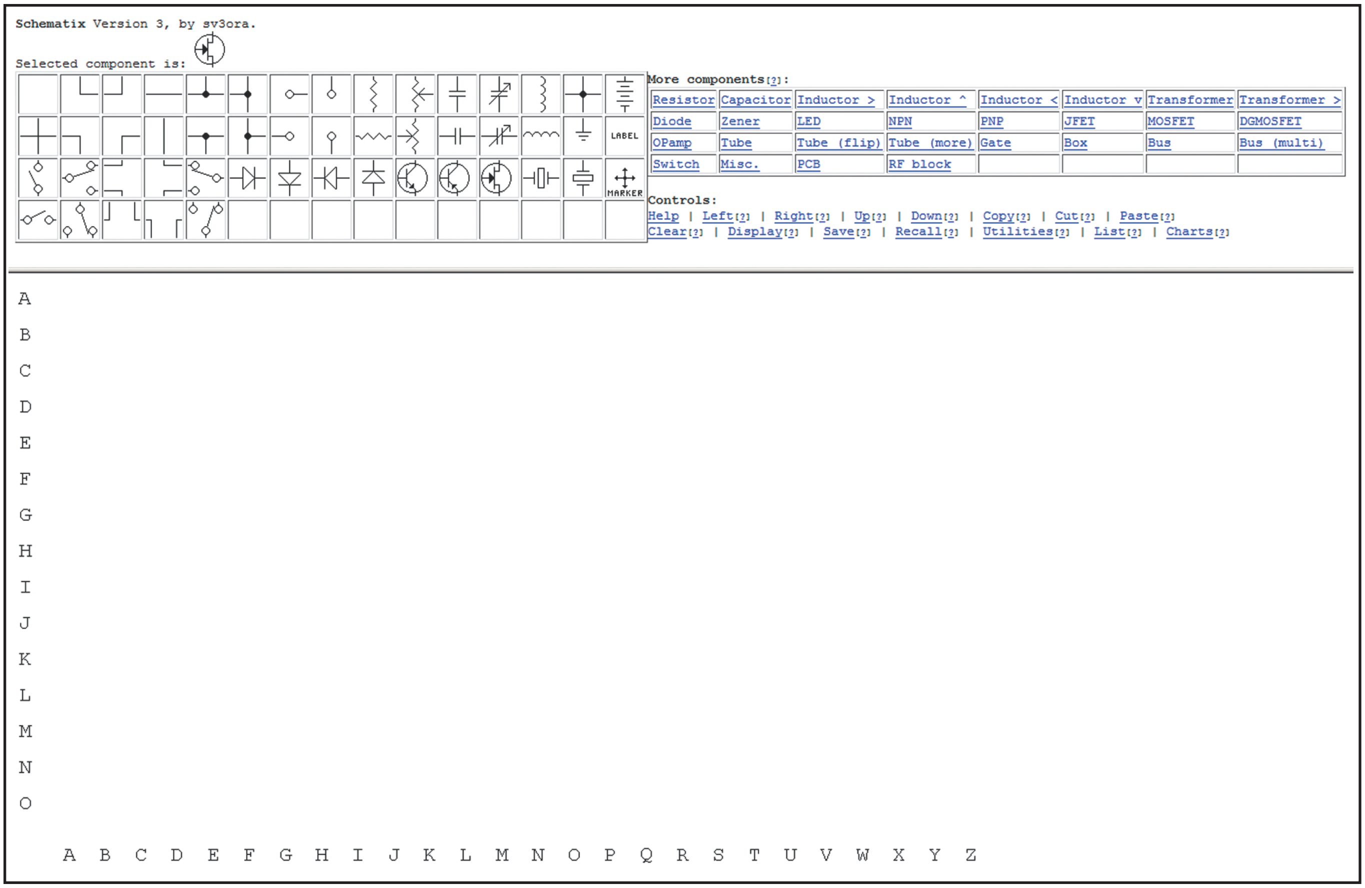
There are not many ways one can send a schematic over the air today, using the low bandwidth required on HF. The most obvious way is to export the schematic generated by a CAD program as an image and then use SSTV, digital SSTV, radio-FAX or another image-based mode that allows for image conversion to audio tones. Then transmit these audio tones using your SSB, FM, or AM transceiver. While this is OK, there are several problems with this approach:

- Unnecessary data is being transmitted, so data efficiency is low. Using such modes, you transmit the same amount of data (so it takes up the same amount of time), whether you transmit a full-color or black & white image, or even a blank image. There is no distinction between such things.
- A voice (SSB/AM/FM) transmitter is required (usually expensive to buy or complex to build). CW or any other on/off keying mode is just out of question.
- Bandwidth efficiency is low, since an SSB/AM/FM transmitter is used and audio tones that occupy most of the bandwidth are sometimes also used (SSTV).
- Data is very prone to errors, since analog tones are all that is sent. Since they are of infinite state, a decoder cannot distinguish between an error and the actual data.
- An error in one area of the schematic image requires a request to retransmit the whole image, for the error to be corrected.
- None of these modes are human-oriented; a computer is always needed for sending and receiving schematics.

Most of these problems arise because an image is supposed to be sent, which can be a picture of a schematic or any other photo. Schematix restricts the type of data to be sent to letters and numbers only; you cannot send an image with it. Although the end result is an image, no actual image is sent. There are also some rules about size and the way you draw your schematic and label the components. However, Schematix has significant advantages over other modes:

- A voice (SSB/AM/FM) transmitter is not required. Any on/off keying mode, including CW, can be used. This allows

** sv3ora@qrp.gr*
This article is adapted from a more detailed version on the author's website, <www.qrp.gr>



Schematix Version 3, by sv3ora.

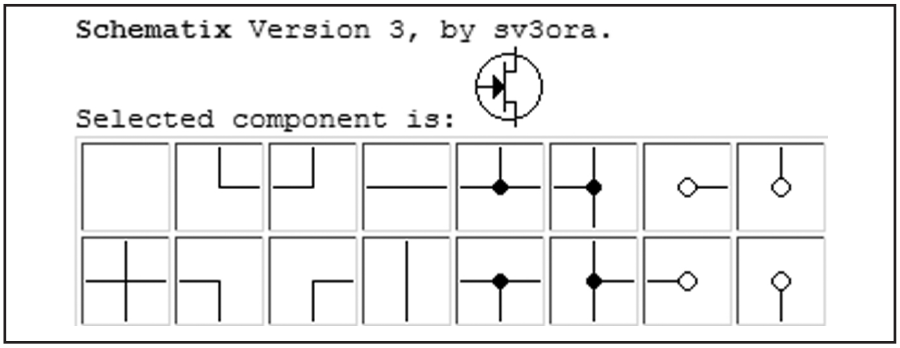
Selected component is:

More components[?]:

Resistor	Capacitor	Inductor >	Inductor ^	Inductor <	Inductor v	Transformer	Transformer >
Diode	Zener	LED	NPN	PNP	JFET	MOSFET	DGMOSFET
OPamp	Tube	Tube (flip)	Tube (more)	Gate	Box	Bus	Bus (multi)
Switch	Misc.	PCB	RF block				

Controls:
Help | Left[?] | Right[?] | Up[?] | Down[?] | Copy[?] | Cut[?] | Paste[?]
Clear[?] | Display[?] | Save[?] | Recall[?] | Utilities[?] | List[?] | Charts[?]

Figure 1. The basic Schematix drawing screen. Commonly used schematic symbols are at the upper left, with options at the upper right for accessing less common symbols. The lower part of the screen is the table in which your schematic will be drawn.



Schematix Version 3, by sv3ora.

Selected component is:

Figure 2. When you click on a component from the menu, it appears at the top of the table. You then click your mouse over the cell where the component should go and it will drop into place.

sending schematics over the air, using cheap homebrew transceivers or powerful, but cheap and power-efficient switching-mode amplifiers (Class-E).

- Bandwidth efficiency can also be high if CW, PSK, or similar modes are used, since these modes occupy minimum bandwidth.
- Data efficiency is high, as only usable symbols are sent, not blank spaces. Also, the fewer components your schematic has, the less data is sent. This is not limited by the total schematic size, but only by the number of components on it. Also, when you use CW, the more common components in the schematic are assigned to shorter Morse code characters, so the efficiency is maximized.
- Data is not so prone to errors, because it is digital and it is presented by finite states of audio tones, carrier-switching or phase-shifting (depending on which mode you use).
- Received errors can be easily detected, since they are presented in the received schematic as blank components, broken components, wrong components, or duplicated components.
- An error in one area of the schematic does not require a request for retransmission of the whole schematic, but just the specific area where the error is.
- Errors can be corrected by the user in some cases, without asking for retransmission, by looking at the received schematic consistency. It is analogous to (but not the same as) the Feld-Hell mode, in which errors can be corrected by the human brain, without the need for retransmission.
- Finally, it is human-oriented. A computer is not necessary for drawing, sending, or receiving schematics. CW or a phonetic alphabet can be easily used, in combination with encoding/decoding printout tables.

Basic Operation

To try Schematix, go to my website, <www.qrp.gr>, click on the Schematix link and select the option to test the program online. Alternatively, you can

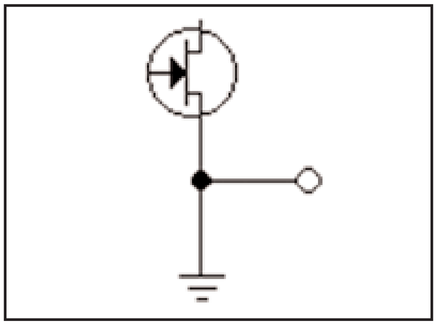


Figure 3. The program has fixed cell sizes and lead locations so that everything lines up with components and wires in adjacent boxes.

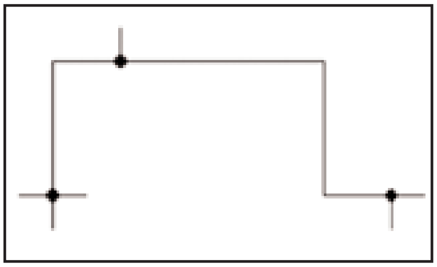


Figure 4. Schematix treats wires and junctions like components. You select and drop in segments of straight lines, bent lines and different types of junctions.

download the program to your PC, unzip it, and run the file "index.html" inside the program folder. This will open up your browser and run the Schematix application (*Figure 1*).

As the program loads, you will see the symbols in the top frame clunk in. Once it has fully loaded, you should see a table of available symbols at the top of the page, with the currently chosen symbol displayed above it. There is also an option for selecting different functions. You will also notice a blank area at the bottom of the page, which is indexed by letters. This is where your diagram will be drawn. If your browser pops-up any messages related to blocked content, you must choose to allow this blocked content in order for Schematix to run correctly.

The first thing to do to draw a schematic is to select the symbol that you want from the top table. Put the mouse pointer over any of the symbols in the table and click it (*Figure 2*). The symbol that's isolated above the table (currently selected component) should change to the symbol in the table that you have clicked.

If it doesn't work, try a few more times. If you can't make it work, you might need to enable Javascript. If it just absolutely refuses to obey, you may need to upgrade your browser.

Now that you have selected the symbol you want, you can place it anywhere in the drawing area below the table of symbols. Put the mouse pointer where you want the symbol to be placed and click. The symbol should drop into that spot.

The drawing area is a table. You can place only one symbol in each cell of the table. For convenience, the drawing area is indexed into lines and columns (A-Z), so that you know in which cell a component is placed. The alignment of the components in the table is automatic and the size of the table cells is fixed. You can't place a symbol halfway between one cell and another. This is why, when you place a symbol, it seems to magically line up with the other symbols in the drawing (*Figure 3*).

To experiment, proceed to place some resistors or capacitors or whatever you want in your circuit. Now comes the process of hooking them up. In other programs, you just start drawing in wires. In Schematix, you have to place the wires as though they were symbols. In the table of symbols, you will find straight wires, elbows, tees, and one kind of cross.

To draw a straight wire that crosses more than one cell, you have to place a straight-line symbol in each cell. When you want to turn a corner, you select the appropriate elbow and place it. If you need a tee, just select it and place it. These wires connect the middle of the sides of the cells (*Figure 4*). The component symbols are also designed so they connect to the middle of the sides of the cells, so the wires and the components automatically line up. This auto-alignment of components is useful for older people with vision problems, but it has also been proven very quick and effective when drawing.

When you place a symbol that you don't want by mistake, you can erase it by selecting the blank symbol in the upper left corner of the table and clicking onto the symbol you don't want. Also, if you want to change a symbol to a different one, just select the desired symbol in the table and click on the previous one to replace it.

Additional symbols appear in row 4 of the symbol table. To get the extra symbols to appear in row 4, click on the appropriate component type in the "More components" table on the right. For example, if you want to draw with vacuum tubes, click on the "Tube" keyword. Symbols for a diode, triode, tetrode, and pentode will appear in row

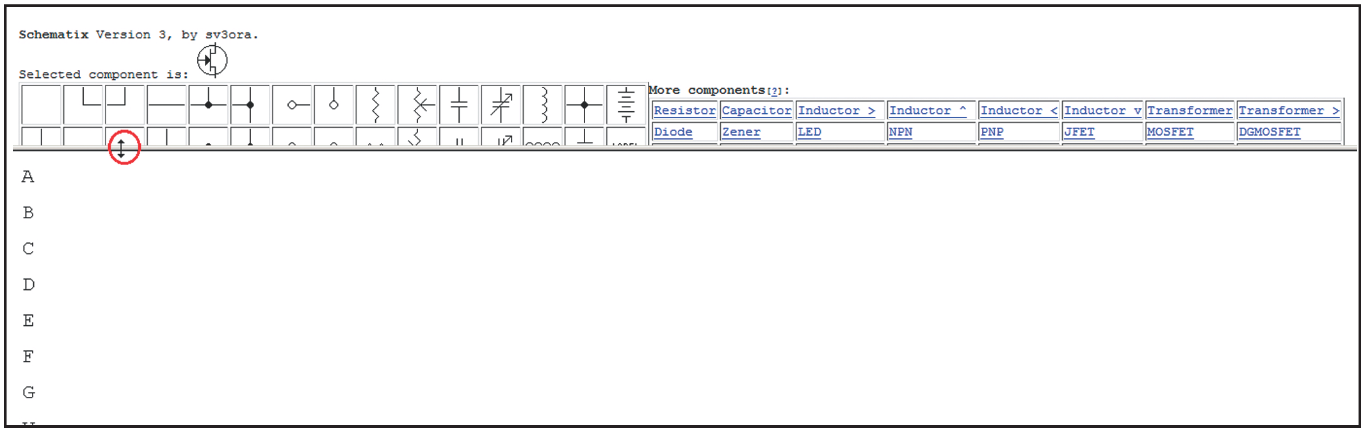
4. You can now select these symbols and place them just as you would any other symbol.

The components labeled "LABEL" and "MARKER" are used for two special functions. The "LABEL" is used for labeling your components (more on this later) and the "MARKER" symbol is used twice between one or more components you have previously drawn, to select them. Then copy/cut and paste them in a different location. Please read the documentation inside the program, if you want to learn how to use the MARKER symbol. This function is useful if you want to copy, move or delete large areas within a schematic.

It is important to note that markers are not symbols that must be exported or saved. They exist only to be used by the copy/cut/paste functions and they disappear after these functions have been correctly used. Please ensure there are no markers left in your schematic prior to clicking "Utilities" or any save function. You may erase a marker just as you would any other component.

Hints on Schematic Drawing

Note that if your screen resolution is not very high, not all lines or columns (A-Z) may appear in the drawing area. If you draw small schematics, this may not be a problem, but if you draw larger ones, it might be. If not all columns appear in the drawing area, you can make them show by horizontally resizing your browser window. If not all lines appear in the drawing area, you can make them show by resizing the



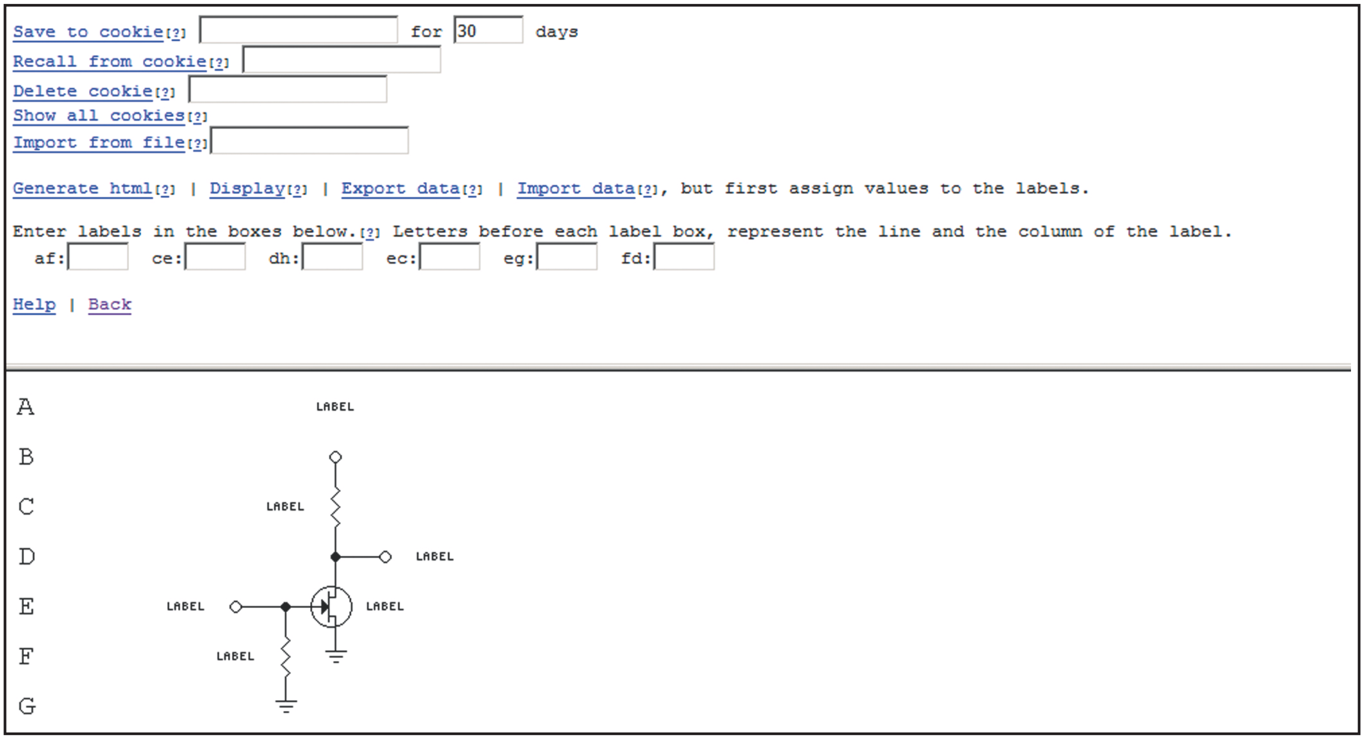
Schematix Version 3, by sv3ora.

Selected component is:

More components[?]:

Resistor	Capacitor	Inductor >	Inductor ^	Inductor <	Inductor v	Transformer	Transformer >
Diode	Zener	LED	NPN	PNP	JFET	MOSFET	DGMOSFET

Figure 5. If your screen resolution is low, it's possible to increase the display area of the schematic table by dragging up the boundary with the top menu area (which will then be smaller). Note the double arrow circled in red that appears when you are changing a window boundary.



Save to cookie[?] for 30 days

Recall from cookie[?]

Delete cookie[?]

Show all cookies[?]

Import from file[?]

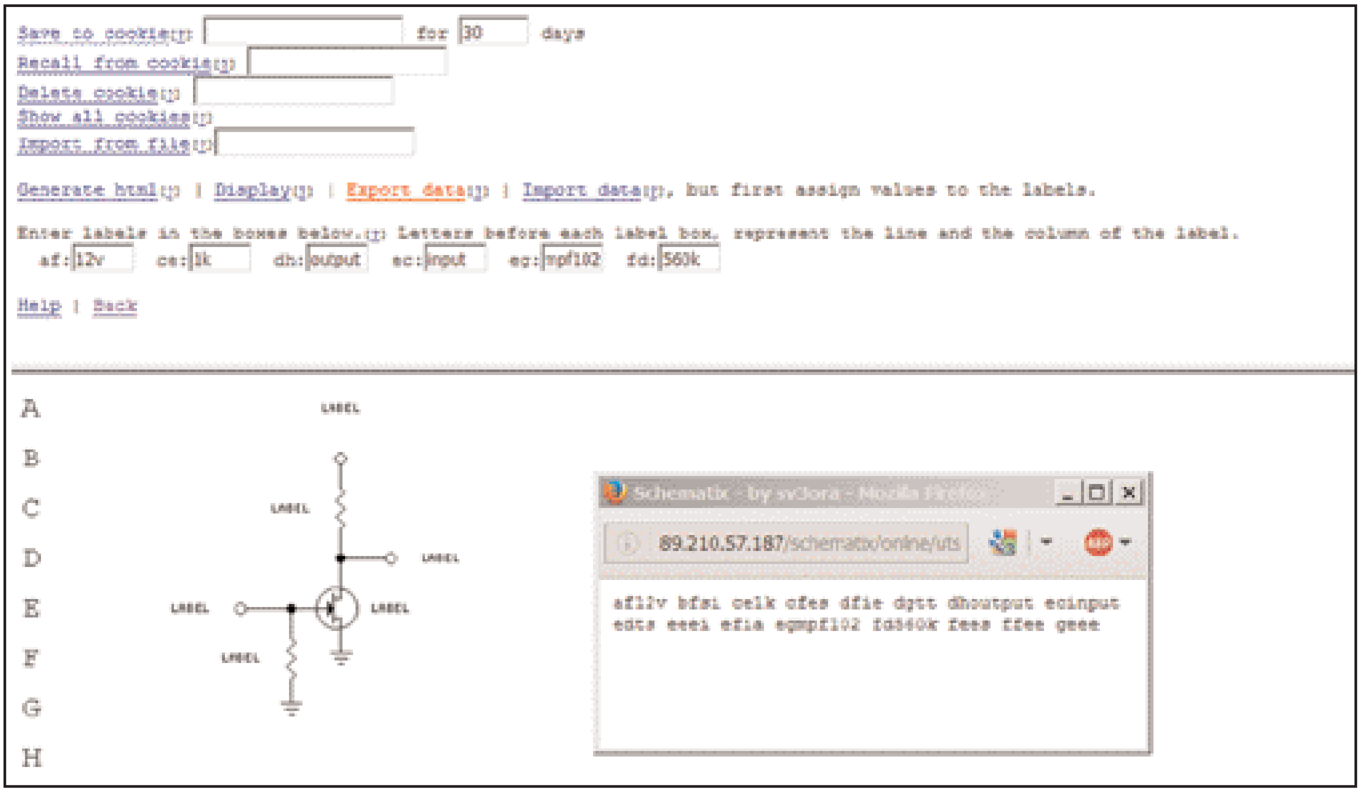
Generate html[?] | Display[?] | Export data[?] | Import data[?], but first assign values to the labels.

Enter labels in the boxes below.[?] Letters before each label box, represent the line and the column of the label.

af: ce: dh: ec: eg: fd:

Help | Back

*Figure 6. After you've drawn your circuit and inserted the word "LABEL" next to each component, you then use the boxes at the top of the **UTILITIES** screen to enter the value for each label. The boxes are identified by the location on the grid of each LABEL that you've inserted.*



Save to cookie[?] for 30 days
Recall from cookie[?]
Delete cookie[?]
Show all cookies[?]
Import from file[?]

Generate html[?] | Display[?] | Export data[?] | Import data[?], but first assign values to the labels.

Enter labels in the boxes below.[?] Letters before each label box, represent the line and the column of the label.
af: 12v ce: 1k dh: output ec: input eg: mpf102 fd: 560k

Help | Back

Schematix - by yo3ora - Mozilla Firefox
89.210.57.187/schematix/online/uts

```
af12v bfsi celk cfes dfie dgtt dhoutput ecinput
edts eeei efia egmpf102 fd560k fees ffes geee
```

Figure 7. When the schematic is finished and you click on ***Export data****, a window will appear that either shows you the encoded data or highlights errors and tells you how to fix them.*

bottom drawing area frame. Note that this will reduce the size of the top symbols frame (*Figure 5*).

If you still can't make all lines and columns appear, you can zoom out your browser's window view. In many browsers, a quick way to zoom in/out is to hold down the left CTRL key on the keyboard while simultaneously scrolling the middle wheel of the mouse. If this can't change the zoom level, you can zoom in/out from the browser preferences. See your web browser's documentation on how to change the zoom level.

Controls

The controls area contains different functions which are useful when drawing your schematic. I will only briefly describe their operation here. For more extensive information, please read the documentation inside Schematix.

The **Left**, **Up**, **Right**, and **Down** functions move the whole schematic in these directions. Take care not to move the schematic to positions exceeding the boundaries of the drawing area, because you may lose drawn components. The **Copy**, **Cut**, and **Paste** functions are used in conjunction with the MARKER symbol explained above. The **Clear** function deletes the whole schematic. The **Save** and **Recall** functions constitute a simple way of temporarily saving your schematic (in a cookie) or recalling a previously saved one. You can think of this as an "undo" function, but you have to manually save your schematic when you want it. The **List** function allows you to save or recall your schematic in a text list, which is useful for saving it as a text file. The **Display** function simply displays your current schematic in a new window. Then you can print-screen it or save it as an HTML file. The **Charts** link leads you to useful tables and documents, which you can use to draw/encode/decode schematics without the use of a computer. For those of you who are interested in this capability, please read the documentation, where it is explained more thoroughly. The **Utilities** function leads you to another page, where you can enter components' values, import/export your schematics for exchanging them over the air, and more.

For your convenience, there is a little help symbol "[?]" right next to each function. If you click that, you are redirected to the relevant section of the documentation that explains the current function.

Labeling Components

Drawing a schematic in Schematix is a two-step process. First, you complete the drawing of your full schematic, as described above, and only then do you enter the components' labels. You cannot enter labels as you draw your schematic, only when you fully complete its drawing. Also, you cannot come back and edit a schematic once you have entered component labels.

If you click the "LABEL" icon in the components table, you should be able to place the word LABEL in the drawing area next to the component you want. You must do this for each component in your schematic that you want to label.

The actual labels (component values) will be added later on, using the "UTILITIES" function. This function replaces the LABEL words next to your components with the actual values that you will enter.

Utilities

After you have finished editing your schematic, click **Save** or save it to a list, then click the UTILITIES function. This will cause the symbols table to disappear and the utilities frame to appear in the top frame. The schematic should still be in the bottom frame.

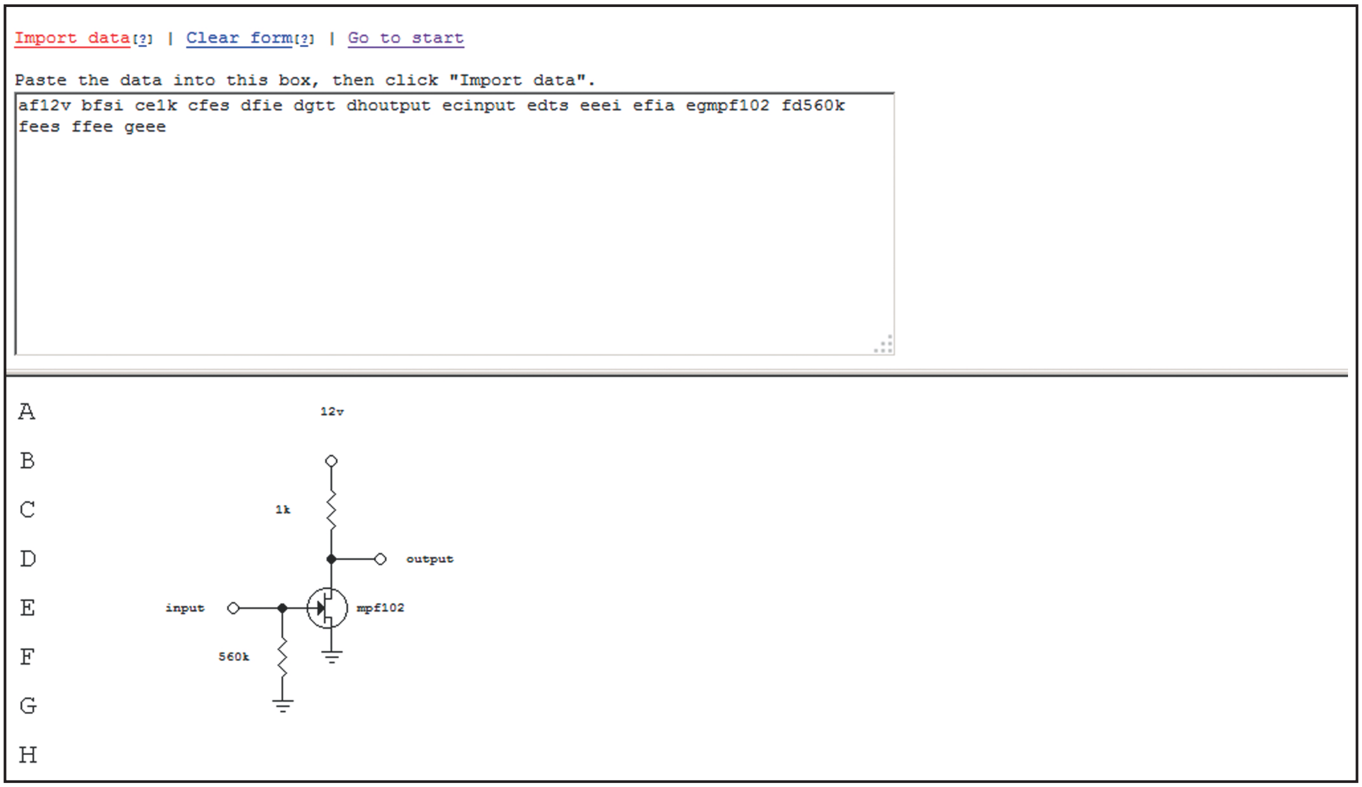
The most useful functions on the Utilities page are: **Import data**, **Export data**, **Display**, **Generate html**, and **component labeling**, which is described below.

The labels (component values) are entered in the form fields provided in the top utilities frame (*Figure 6*). Again, labels may be assigned to components only after completion of a schematic drawing and before using the "**Generate html**," "**Display**," or "**Export data**" functions.

Schematix restricts the way in which labels are entered in the form fields. This is done for compatibility with the "Export data" function. The rules are simple, but must be followed. Labels must be:

- Greater than 2 and no more than 6 characters; or if they are only two characters, at least one of them must be a digit.
- Single character labels or empty labels are not allowed.
- Special characters (such as "+") are not allowed; only letters or numbers without spaces between them.

Also, to be compatible with the **Export data** function, labels


Import data[?] | Clear form[?] | Go to start

Paste the data into this box, then click "Import data".

```
af12v bfsi ce1k cfes dfie dgtt dhoutput ecinput edts eeei efia egmpf102 fd560k
fees ffee geee
```

*Figure 8. The schematic recreated at the receiving end, after pasting the received text into the box at the top of the screen and clicking on **Import data**.*

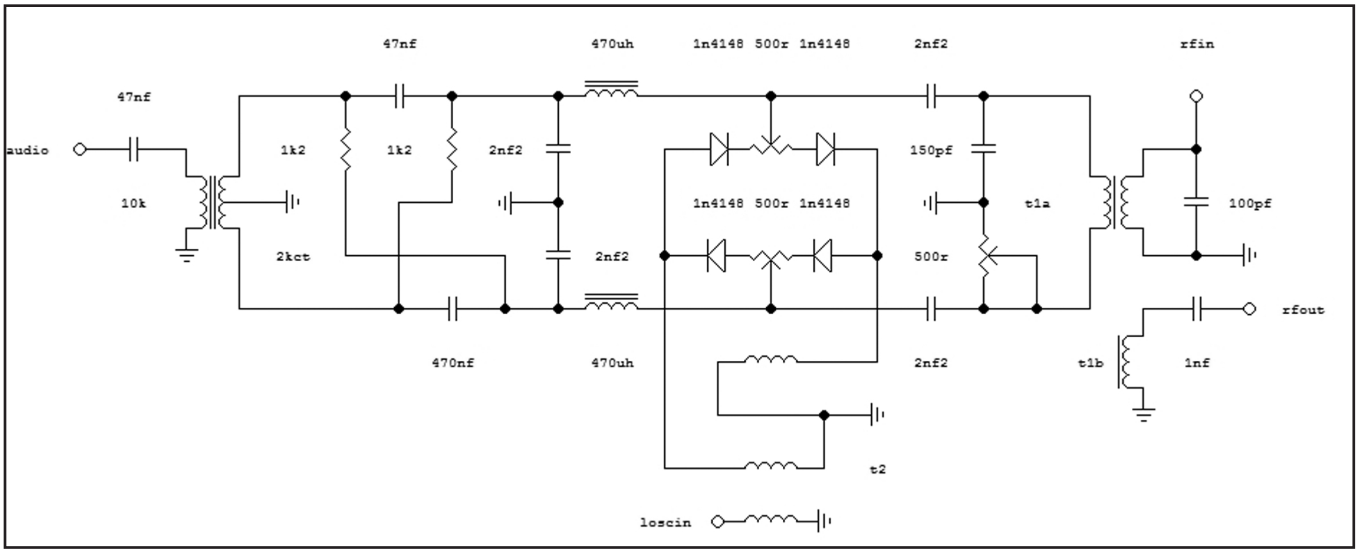
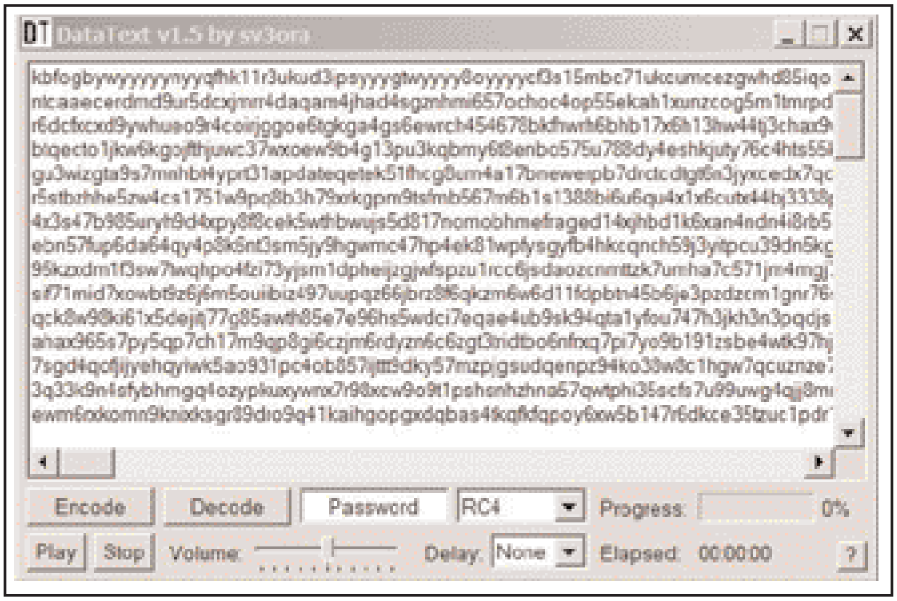


Figure 9. Here is a schematic diagram drawn using Schematix that will be transmitted using various modes and speeds in order to compare results.



```
kbfogbywyyyyynyyqfhk11r3ukud3ipsyyygtwyyyy8oyyyycf3s15mbc71ukcumcezgwhd85iqo
nlcaaecerdmd9ur5dcxjmrr4daqam4jhad4sgmhmi657ochoc4op55ekah1xunzcog5m1tmrpd
r6dcfxcxd9ywhueo9r4coirjggoe6tgkga4gs6ewrch454678bkfhwrh6bhb17x6h13hw44tj3chax9
btqecto1jkw6kgojfthjuwc37wxoew9b4g13pu3kqbmy6t8enbo575u788dy4eshkquty76c4hts55
gu3wizgta9s7mnhbt4yprt31apdateqetek51fhcg8um4a17bneweipb7drctcdtgt6n3jyxcedx7qc
r5stbrhhe5zw4cs1751w9pq8b3h79xrkgpm9tsfmb567m6b1s1388bi6u6qu4x1x6cutx44bj3338
4x3s47b985uryh9d4xpy8f8cek5wthbwujs5d817nomobhmefraged14xjhbd1k6xan4ndn4i8rb5
ebn57fup6da64qy4p8k6nt3sm5jy9hgwmc47hp4ek81wpfysgyfb4hkcqnch59j3ytpcu39dn5kc
95kzxdm1f3sw7iwqhpo4fzi73yjsm1dpheijzgjwfspzu1rcc6jsdaozcnmttzk7umha7c571jm4mgj
sif71mid7xowbt9z6j6m5ouiibiz497uupqz66jbrz8f6qkzm6w6d11fdpbtn45b6je3pzdzcm1gnr76
qck8w98ki61x5dejtj77g85awth85e7e96hs5wdci7eqae4ub9sk94qta1yfou747h3jkh3n3pqdjs
anax965s7py5qp7ch17m9qp8gi6czjm6rdyzn6c6zgt3ndtbo6nfxq7pi7yo9b191zsbe4wtk97hj
7sgd4qofjjyehqyiwk5ao931pc4ob857jjtt9dky57mzpjgsudqenpz94ko38w8c1hgw7qcuznze
3q33k9n4sfybhmgq4ozypkuxywnx7r98xcw9o9t1pshsnhzhna67qwtphi35scfs7u99uwg4qjj8m
ewm6xkomn9knixksgr89dro9q41kaihgopgxdqbas4kqfkfqpoy6xw5b147r6dkce35tzuc1pdr
```

Figure 10. The circuit in Figure 9 encoded to text using the author's DataText program. It generates more than 75,000 characters and would take hours to days to transmit if sent with CW.

with decimal values are automatically converted to non-decimal values. For example 4.7K is automatically converted to 4k7. So keep these things in mind when assigning label values, but do not worry too much, because the program automatically warns you of any errors, and allows you to correct them.

The number of label input fields, which are displayed for you to enter data, is automatically adjusted to match the number of LABEL images in your schematic. Before each label input field, its line and column letters are displayed, so that you know which label in the schematic you are editing. For example, a letter pair "fm" before a label input field would place the text you type into that input field in the cell at the junction of line "F" and column "M" in the lower table.

For your convenience, note that the schematic is scanned for LABEL symbols, starting at the top left of each line and proceeding to the right. When it gets to the right margin it moves down and does the next line of symbols. Thus, the order of the label input fields follows this pattern.

Display Labeled Schematic

The **Display** function displays your schematic, including your labels, in a new window, to check if everything is OK, before exporting it.

Generate HTML (Including Labels)

The **Generate HTML** function generates HTML code for your schematic, including its labels. It can be used as an alternative method to save your schematics, but your saved schematics must be inside the Schematix folder in HTML form.

Exporting Schematics (Export Data Function)

The **Export data** function is used to export a schematic and its labels in a

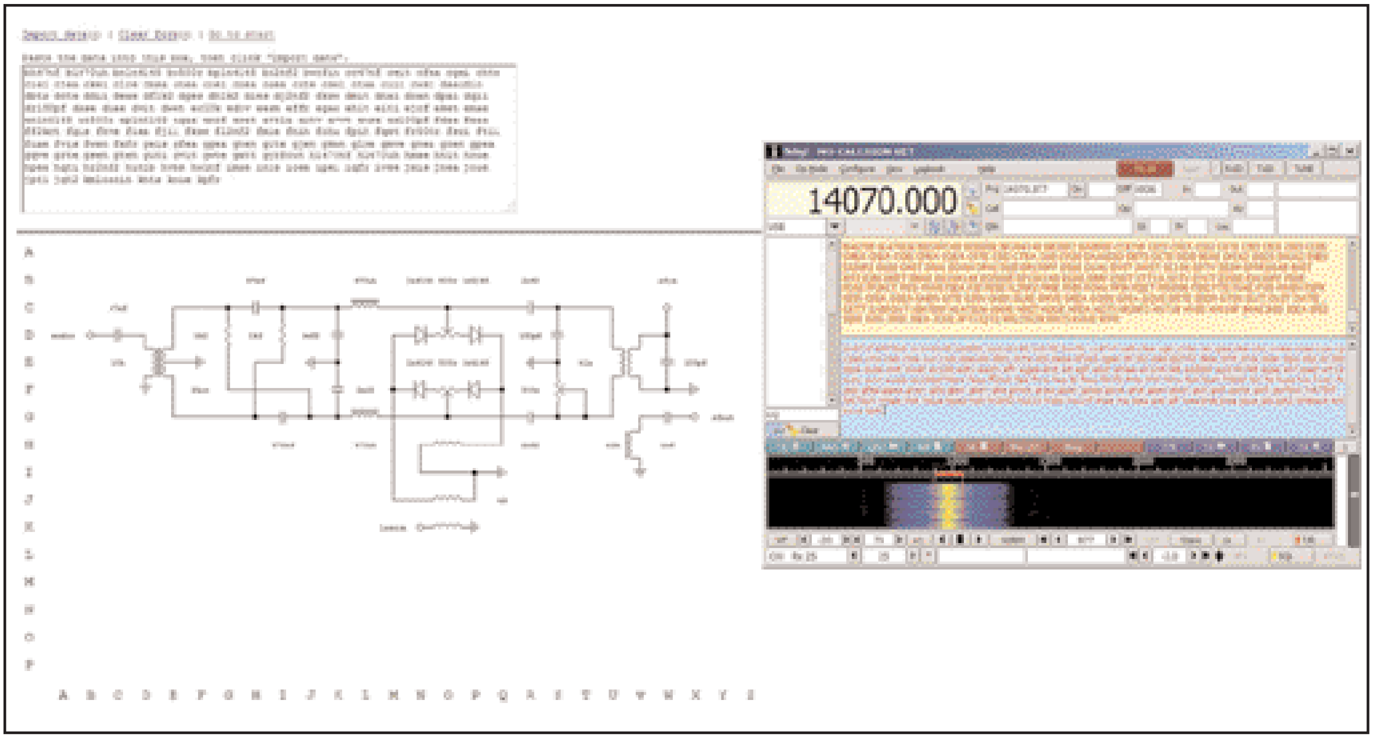
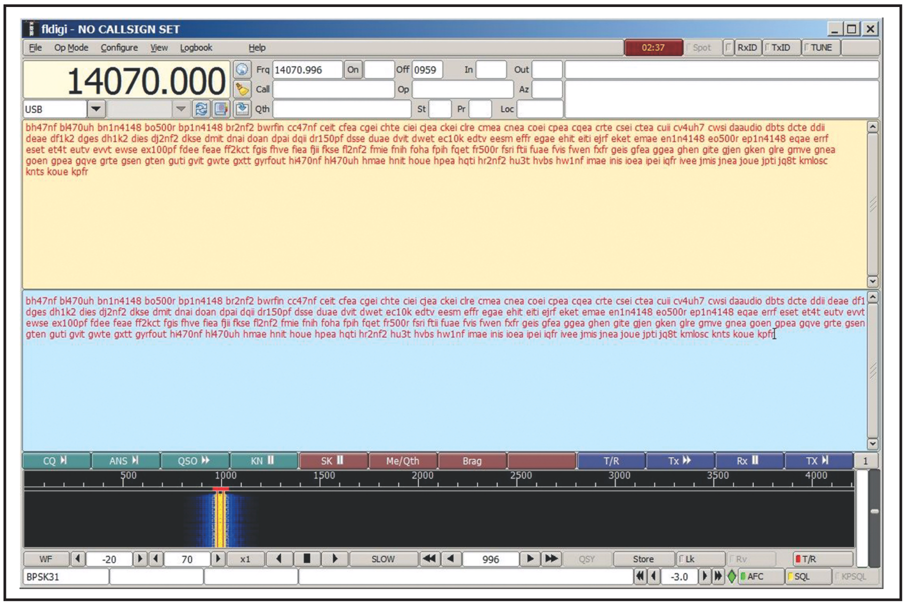


Figure 11. Here, the same circuit has been encoded using the Schematix program and has only 750 characters, making it much more practical to transmit over the air. In this screen shot, the data is being sent via 25 word-per-minute CW and would take a little over 5 minutes to transmit.

form that is efficient for sending this schematic over the air to other hams. With the schematic displayed in the bottom frame and all its labels filled in on the form fields in the top frame, click **Export data** to export the schematic (*Figure 7*). If there is an error in the labels, a pop-up window will be displayed, telling you where the error is and how to correct it. If there are no label errors, a new window will open, containing the exported data text. (This is text, not a diagram.)

Select all text in that window and copy it. Then paste the copied text inside your favorite sending program, to send it over the air, using your favorite mode. *Figure 7* shows the Utilities page with the form filled in, and the popup window with the code generated, after clicking the **Export data** function.



```
bh47nf bl470uh bn1n4148 bo500r bp1n4148 br2nf2 bwrfin cc47nf ceit cfea cgei chte ciei cjea ckei clre cmea cnea coei cpea cqea crte csei ctea cuii cv4uh7 cwsi daaudio dbts dcte ddii
deae df1k2 dges dh1k2 dies dj2nf2 dkse dmit dnai doan dpai dqii dr150pf dsse duae dvit dwet ec10k edtv eesm effr egae ehit eiti ejrf eket emae en1n4148 eo500r ep1n4148 eqae errf
eset et4t eutv evvt ewse ex100pf fdee feae ff2kct fgis fhve fiea fjii fkse fl2nf2 fmie fnih foha fpih fqet fr500r fsri ftii fuae fvis fwen fxfr geis gfea ggea ghen gite gjen gken glre gmve gnea
goen gpea gqve grte gsen gten guti gvit gwte gxtt gyrfout hi470nf hl470uh hmae hnit houe hpea hqti hr2nf2 hu3t hvbs hw1nf imae inis ioea ipei iqfr ivee jmis jnea joue jpti jq8t kmlosc
knts koue kpfr
```

```
bh47nf bl470uh bn1n4148 bo500r bp1n4148 br2nf2 bwrfin cc47nf ceit cfea cgei chte ciei cjea ckei clre cmea cnea coei cpea cqea crte csei ctea cuii cv4uh7 cwsi daaudio dbts dcte ddii deae df1
dges dh1k2 dies dj2nf2 dkse dmit dnai doan dpai dqii dr150pf dsse duae dvit dwet ec10k edtv eesm effr egae ehit eiti ejrf eket emae en1n4148 eo500r ep1n4148 eqae errf eset et4t eutv evvt
ewse ex100pf fdee feae ff2kct fgis fhve fiea fjii fkse fl2nf2 fmie fnih foha fpih fqet fr500r fsri ftii fuae fvis fwen fxfr geis gfea ggea ghen gite gjen gken glre gmve gnea goen gpea gqve grte gsen
gten guti gvit gwte gxtt gyrfout hi470nf hl470uh hmae hnit houe hpea hqti hr2nf2 hu3t hvbs hw1nf imae inis ioea ipei iqfr ivee jmis jnea joue jpti jq8t kmlosc knts koue kpfr
```

Figure 12. Sending the same data using BPSK-31 (with only a tiny bit more bandwidth) reduces transmission time to just over 2-1/2 minutes. Using BPSK-63 cuts the transmit time to less than a minute and a half, while still using minimal bandwidth.

Importing Schematics (Import Data Function)

The **Import data** function is used to import a schematic and its labels, that has previously been transmitted to you over the air by another ham.

By clicking **Import data** (*Figure 8*), a text box is displayed in the top frame. Copy the text you have received from your favorite receiving program and paste it into this text box. Then click **Import data** on that page. The imported schematic will appear in the bottom frame.

You can repeat this process as many times as you like, by clearing the text box using the **Clear form** link and pasting your data text into the box again. *Figure 8* shows the Import page and some data that has been pasted into the form. This data "reveals" the schematic at the bottom frame when the **Import data** link is clicked.

Error Detection and Correction

In Schematix, some form of error detection — and in some cases, correction — can be performed without data retransmission or additional data overhead. Humans can be used to correct the data to some extent. Error detection is done automatically, as well as by humans. The error detection is partially due to the rules of component labeling. Additionally, it depends on the available components in the Schematix library, as well as schematic inconsistency that can only be noticed by the user. I will use some examples below to illustrate.

If a word is less than four characters long, the program knows an error has occurred, but it does not know if the error is in the part of the word that refers to the position of the component or the part that refers to the component/label, so the symbol is ignored. These errors are shown as blank blocks in the schematic and in some cases they can be spotted by the user when looking at the overall schematic for inconsistencies (e.g., blank blocks where they shouldn't exist).

In some cases, the user can guess the component behind the error and correct it without asking for a resend. For example, if a continuous straight wire is broken at some point, the user can easily guess that this point should be a wire connected to the rest of the wire. Another example may be a label with a value of "1k" and no nearby component. It's reasonable to assume that a resistor should be somewhere nearby, and that the blank cell is where it should be.

In another example, when the first two characters of a word appear to contain at least a number, the program knows an error has occurred and the current component/label is ignored. This error is also shown as a blank space in the schematic and it is again up to the user to spot and correct it.

Let us now consider an example in which a word that has been received, is four or more characters long and its first two characters do not contain numbers. Based on the Schematix labeling rules, the word consistency is correct. In this case, the first two letters of the word represent the line and the column of the component/label in the table and the remaining characters represent either the component or the label value. However, there is no way to check whether an error has occurred in the actual component types or label values. Despite this, there are two things that can be done to spot an error.

One is by the user doing manual error checking/correcting, by looking at the overall schematic for inconsistencies, as described above. The other is done automatically by the program and it refers to the special case in which a component is received which does not exist in the Schematix library. In this case, it is certain that an error has been received and the missing component is shown in the table as a broken image.

Another example is if, due to an error in location data, two components are placed in the same cell. The program will put them both there, leaving it up to the user to try to correct, the error. In some cases, by looking at the schematic, the user might be able to decide which of the two components better matches the current cell. (*You will also need to figure out where the other component belongs, but there's a good chance that cell will be blank. – ed.*)

Keep in mind that error *detection* is different from error *correction*. In the current version of Schematix, there is no automatic data error correction implemented. This has been done on purpose, to minimize the data that has to be transferred over the air. However, as noted above, there may be cases in

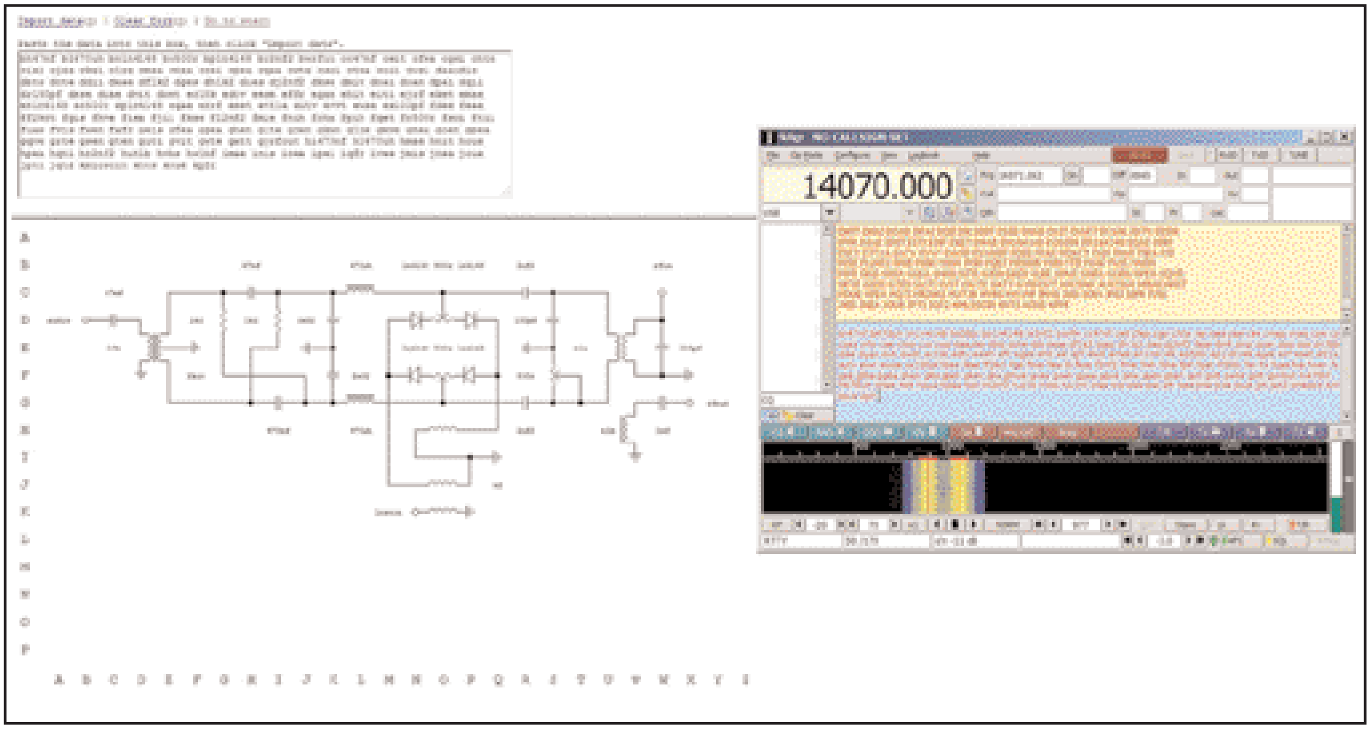


Figure 13. Using 50-baud RTTY to send the schematic would take 2 minutes and 21 seconds.

which you might be able to correct an error by simply looking at the schematic for inconsistencies.

In case the user (or the program) spots an error in an imported schematic that cannot be corrected by either of them, you can ask for retransmission of data from your ham friend. However, you do not need to ask for retransmission of the full schematic. Due to the way Schematix is made, you can instead request retransmission of data for specific cells. Just let your ham friend know which cells you have not received correctly (line and column letter of the cell on the grid), in order to retransmit them to you.

Drawing and Exchanging Schematics Without a Computer

There may be various reasons why one may not have a computer available. For example, some might consider the cost, although this is not a big issue for computers today, as nearly everyone already has one. Another reason may be the extra weight and size, but more importantly the power requirements, if one has to carry a computer along when operating outdoors. While a simple homemade transceiver might draw a few tens or hundreds of milliwatts, a laptop would draw a few watts. Another reason may be the need for weatherproof equipment, and a computer usually isn't.

Schematix allows drawing, sending and receiving schematics, PCBs and diagrams even without the help of a computer, as it has been designed to be human-oriented. This is a unique feature that is possible because of the design of the drawing and the exchanging mechanisms. Although drawing and exchanging schematics without a computer is more time-consuming, it can be done with a little patience. Note that you can only use CW or phonetic alphabet

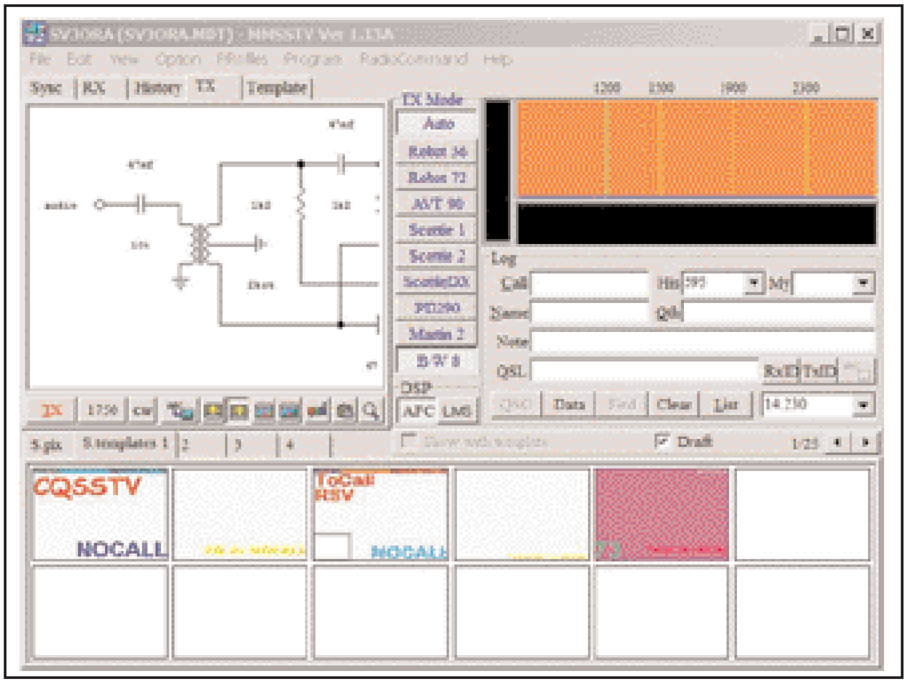


Figure 14. Sending the schematic as an image using slow-scan TV is problematic. In the B/W8 mode shown here, transmission time is only 8 seconds, but the resolution is very low and the image is severely cropped.

modes, if you do not have a computer with you, since these are the only human-oriented operating modes. If you are interested in this nice feature, please read the details in the Schematix documentation.

A Note About Using Schematix On-Air

Schematix is intended for use by radio amateurs to efficiently send their previously drawn schematics to other hams via Morse code or any other mode. The text that is being sent over the air is encoded, but not encrypted. This is an important distinction.

An operator may listen to the CW stream and write down the decoded letters, but they will mean nothing to him, so they may appear to be encrypted. But this is not the case. Unless the operator knows how to interpret this data to have meaning for him (i.e., translated to a schematic), he does not know whether this data is encrypted or encoded. However, by just looking at the data, he may notice that some of it makes sense (for example 2n2222). This is a way to notice that this data is Schematix data.

However, to be compliant with FCC rules[2] and "just in case," I would advise you to include some text at the beginning or/and the end of your transmissions along the lines of "the data can be decoded using Schematix" or even including a link to my website, <www.qrp.gr>. This will also help new hams to correctly identify and decode this data.

Comparison to Other Programs/Modes

The easiest way to see why using Schematix may be superior when sending schematics over the air is to consider an example and compare it with other modes/programs. Let's consider a schematic that has been drawn using the Schematix editor (*Figure 9*).

This schematic has been exported as an image, using the print screen button and pasting into Paint, then saving as a low-quality JPG file. The size of the image file is 53 Kb.

If this file was to be encoded to text (*Figure 10*) and then sent using the DataText program I have developed (available on my website), 75093 characters would have to be sent on the air. It would probably take hours or even days to send this via CW and even the slightest error would make the data corrupt, so that retransmission would be needed.

However, if Schematix is used (*Figure 11*), the number of characters is reduced to about 750, including the spaces between words. An error anywhere in the characters would result in corruption of a specific part of the schematic, not the whole schematic. Sending the 750 characters at 25 wpm CW would take about 5 minutes and 18 seconds. Of course, the time will vary based on your sending speed.

If we send the same data using BPSK-31 (*Figure 12*), without sacrificing any bandwidth compared to CW, it will take 2 minutes and 37 seconds. Using BPSK-63, again without sacri-

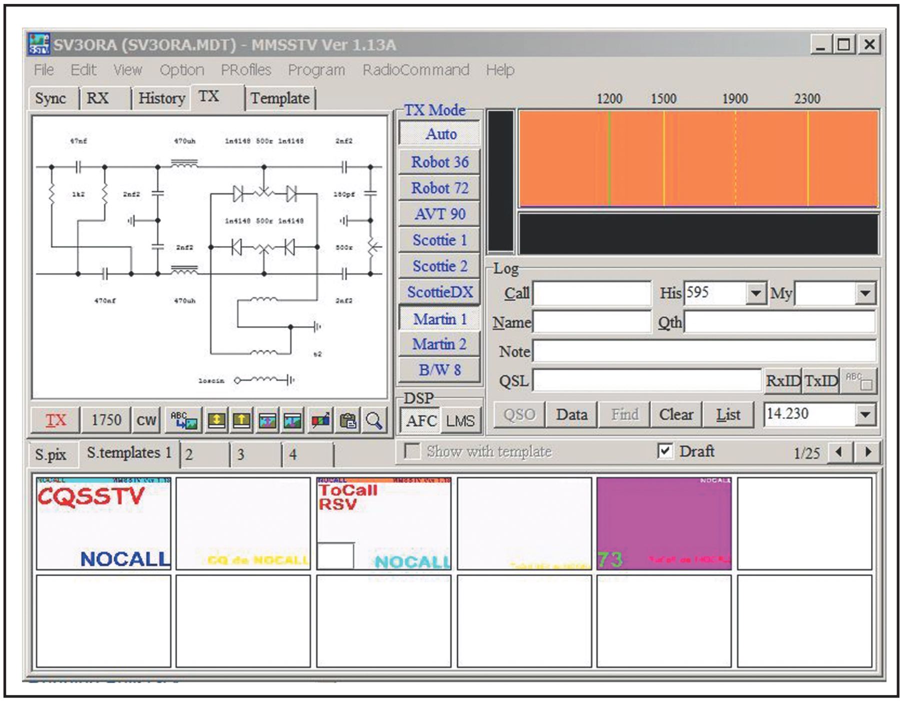


Figure 15. Sending the image with the commonly-used Martin1 SSTV mode takes just under two minutes but still results in a low-resolution cropped image.

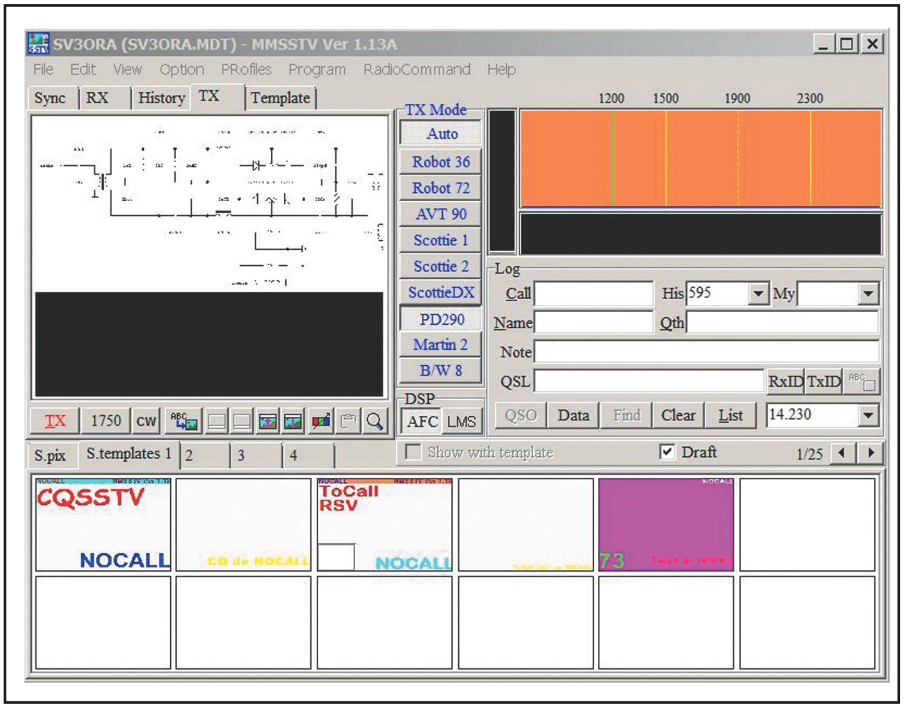


Figure 16. The higher-resolution PD290 mode takes nearly five minutes to send the image and it still needs to be scaled and cropped.

ficing any bandwidth compared to CW, will take 1 minute and 19 seconds.

If we sacrifice the bandwidth a bit and use RTTY-50 (*Figure 13*), the same number of characters can be send in about 2 minutes and 21 seconds. Again, time relates to the speed of sending, so if using RTTY-75, this can be even faster.

Comparing Schematix to SSTV is also interesting. Of course, I have not tested all the SSTV modes, but I have chosen between the extreme ends, the fastest but lower-resolution mode (B/W8), a commonly used one (Martin1) and the slowest but higher-resolution FD290.

Using B/W8 to send the same image (*Figure 14*) would take only 8 seconds, but the resolution is only 160x120 and the image is severely cropped. Trying to scale the image down to 160x120 results in severe loss of schematic information.

Using Martin1 to send the image (*Figure 15*) would take 1 minute and 54 seconds, but the resolution is only 320x256 and the image is still cropped. Trying to scale the image down to 320x256 again results in loss of the schematic information.

Using PD290 to send the same image (*Figure 16*) would take 4 minutes and 50 seconds, and the 800x616 resolution is still not adequate to fit in the whole image without scaling, so the image is cropped again. Trying to scale the image down to 800x616 results in an image that is distorted, but the schematic information can be distinguished. However, given the noise on reception, which directly affects the received image quality, all the image information might not be able to be recovered.

Comparing Schematix to Wefax (most commonly used for sending weather faxes) is interesting as well. Using WEFAX576, the same image has been sent in 6 minutes and 24 seconds. This is a relatively broad signal and is still susceptible to corruption from noise, QRM, or changes in propagation.

To summarize, the same schematic has been sent in different ways with the following results (listing mode, transmission time and notes):

- Datatext: Days, very bandwidth efficient (CW/PSK), not tolerant of errors, not practical.
- Schematix via CW 25 wpm: 5:18, very bandwidth efficient, quite tolerant of errors.
- Schematix via BPSK31: 2:37, very bandwidth efficient, quite tolerant of errors.
- Schematix via BPSK63: 1:19, very bandwidth efficient, quite tolerant of errors.
- Schematix via RTTY-50: 2:21, bandwidth efficient, quite tolerant of errors.
- SSTV B/W8: 8 sec, bandwidth inefficient, cropped image, not practical, not very tolerant of errors.
- SSTV Martin1: 1:54, bandwidth inefficient, cropped image, not practical, not very tolerant of errors.
- SSTV PD290: 4:50, bandwidth inefficient, cropped image, short of OK when scaling, not very tolerant of errors.
- WEFAX576: 6:24, bandwidth inefficient, not very tolerant of errors.

"Schematix is intended to add a new feature that had been missing from the radio amateur community until now."

Conclusions

This idea is something never tried before, as far as I am aware, at least not in this way. In that sense, I would greatly appreciate your comments, positive or negative, so as to improve the program and make it more usable.

Apart from the documentation and the current article, a short video has been created at <http://users.sch.gr/giannopk/Schematix.mp4> where the basic operation of the program is demonstrated.

I would love to see hams use the program on air and comment to me. I would encourage you to try it with members of your local clubs as well. Either way, Schematix is intended to add a new feature that had been missing from the radio amateur community until now.

Notes:

1. Schematix has been developed by the author, based on a previous program written by Jim Osburn, WD9EYB.

2. FCC rules restrict transmission of messages "in codes and ciphers intended to obscure the meaning thereof…" [§97.113(a)(3)]. Since there is no intention to obscure the meaning of the data, this restriction does not apply in this case. However, it is advisable to include some plain text as the author suggests, in order to remove any doubts. – *ed.*