1. sudo raspi-config to open the configuration menu then at option 5 enable VNC
2. At option 7 Advanced options, select A5 for Mode 16
3. Exit raspi-config, but do not reboot yet.
4. At the prompt type sudo apt update
5. At the prompt type sudo apt-get upgrade
6. When the commands have all finished type sudo reboot. Exit the SSH app.

**Entering the Raspberry by RealVNC**

1. On the PC, run VNCViewer and use the IP address to enter the Raspberry Desktop.
2. Use the username and password from the previous steps. You will see the graphic appear of an open road. In all aspects the desktop is similar to a Windows desktop.
3. At the top menu bar click on the terminal icon to open a terminal window in the desktop. You are now ready to continue the preparation of the Raspberry for Svxlink.
4. The next steps may require a reboot, but always return to this position to continue the set up.

**Installation of the USB Soundcard**

1. Install the USB Sound Card on the Raspberry then in the terminal screen type the following command at the prompt.
2. sudo nano /etc/modprobe.d/raspi-blacklist.conf
3. and type the following line.
4. blacklist snd\_bcm2835
5. then save the file (cntrl-o) and exit (cntrl-x) then type
6. sudo nano /lib/modprobe.d/aliases.conf
7. locate the following line
8. options snd-usb-audio index=-2 and insert a “#” before it, then save and exit.
9. **reboot**.

**Sound Card Configuration**

1. Re-open RealVNC viewer to the raspberry pi.
2. Re-open a terminal screen and type the following command at the prompt.
3. sudo alsamixer
4. you should now see that your USB sound card is the default sound card. type cntrl-f5 to view all the sliders.
5. set the first slider to around 60% the second to zero, the third to a maximum of 12 and in the fourth type M to mute the amplifier.
6. Type escape then type
7. sudo alsactl store This will save these settings even after re-booting.
8. sudo adduser svxlink (This will prepare the card for the installation of SVXLink.)
9. Make the password svxlink, as you won’t need to use this any further.
10. Test the installation by sudo aplay -l.

**Compiling SVXLink successfully**

apt-get update

apt-get upgrade

1. To insure git is installed, type sudo apt install git, although I have found it installed as standard.
2. At the prompt in the terminal type sudo git clone [http://github.com/sm0svx/svxlink.git](https://github.com/sm0svx/svxlink.git)
3. This will give you a download of the the latest master of svxlink.git including svxreflector, svxserver and svxremote into a folder called svxlink in the current user.
4. When it is safely downloaded, go to the instructions page to add all the necessary additional software. The list is **very important** and is found on the project page of svxlink.org at install.adoc. Some of the software is standard in Raspbian Stretch so you may find the system reporting this fact. The list of -dev packages is more difficult, but I will show you the following pointers.
5. To install the top list of software simply type “sudo apt install gcc g++ make cmake groff gzip doxygen tar git”
6. For the second half of the list install each of the packs individually.
7. sudo apt install libsigc++-2.0-dev
8. sudo apt install libpopt-dev
9. sudo apt install tcl8.6-dev
10. sudo apt install libgcrypt20-dev
11. sudo apt install libasound2-dev
12. sudo apt install libgsm1-dev
13. sudo apt install libspeex-dev
14. sudo apt install librtlsdr-dev
15. sudo apt install libqt4-dev
16. sudo apt install curl libcurl4-openssl-dev libjsoncpp-dev

This last line is new and permits the metarinfo module to access the weather information from an https source.

If the system responds with a y/n query always go for the y and you will have no issues. The final line to type will be sudo apt install alsa-utils, but it may attract the response that the latest tools are already installed.

**The Compilation**  
Now follows the commands below.

1. cd svxlink/src
2. sudo mkdir build
3. cd build

Now we start compiling. Type the following line EXACTLY, including / \. as they are important for the syntax. This command must go on one line. The final two .. after a single space are important too. The command will take several minutes to execute.

1. cd
2. sudo make – This command will take 30-45 minutes to run dependent on processor used.
3. sudo make doc – This prepares all the documentation and manual files.
4. sudo make install – This compiles all the software into the operating system.
5. sudo ldconfig

**Installation of the Sound Files**

The software is now installed, but not yet ready for use. We need the Language files. My preferred method is as follows.

cd /usr/share/svxlink/sounds – We are now in the correct directory. There is a useful command wget that downloads a file from a web source.

sudo wget <http://(websource)> filename  
If you want the french language…

sudo wget <http://jackson.lestroisours.pagesperso-orange.fr/upgrades/fr_FR.tar.gz>

sudo tar xzvf fr\_FR.tar.gz – unwraps all the necessary files into their proper folders.

If you want the en\_US files they are on svxlink.org.  
If you want to make your own then download DSpeech from Dimio and play with that. However the finished .wav files need to be imported into Audacity, trimmed and exported as microsoft .wav files. This changes the twelfth bit from Hex 12 to Hex 10, to enable them to work in svxlink.

**Setting up the GPIO ports**

We need to setup the GPIO ports. The simplest way is to add the following lines to the file rc.local.

1. sudo nano /etc/rc.local

################

#GPIO SCRIPT #

#TO BE INSERTED#

#ON START-UP #

################

# GPIO 17 as PTT to TX2

echo 17 > /sys/class/gpio/export

echo out > /sys/class/gpio/gpio17/direction

sudo chmod 777 /sys/class/gpio/gpio17/value

# GPIO 18 as Squelch to RX2

echo 18 > /sys/class/gpio/export

echo in > /sys/class/gpio/gpio18/direction

sudo chmod 777 /sys/class/gpio/gpio18/value

#GPIO 24 as PTT to TX1

echo 24 > /sys/class/gpio/export

echo out > /sys/class/gpio/gpio24/direction

sudo chmod 777 /sys/class/gpio/gpio24/value

# GPIO 23 as Squelch to RX1

echo 23 > /sys/class/gpio/export

echo in > /sys/class/gpio/gpio23/direction

sudo chmod 777 /sys/class/gpio/gpio23/value

#end of GPIO Section

#Remove # at start of next line for permanent reboot on power loss.

#sudo svxlink –daemon –logfile=/var/log/svxlink.log

These lines go immediately before the final

exit 0

1. Save and exit the file.

**CW.tcl New Paragraph**

In the first version I described some changes relating to increasing the CW characters in the available list. Those changes to the CW character table still apply, but the syntax that sends the characters/callsigns to the audio chain has changed as follows:

CW::play $mycall 95 650 -4;

CW::play “-” 95 650 -4; This command plays the CW character “-” or the assigned call sign at speed 95 with tone 650 Hz at volume -4, which I find quite adequate for most purposes.