

BlogBooker

Low resolution pictures

From Blog to Book.

svxlink.de

Inhaltsverzeichnis

1	2011	9
1.1	Januar	9
	Impressum (2011-01-14 17:27)	9
	Installation (2011-01-14 17:38)	9
	Download (2011-01-14 17:38)	14
	Allgemeines (2011-01-14 17:38)	15
	Links (2011-01-14 17:40)	16
	Bedienungsanleitung (2011-01-14 17:52)	17
	Konfiguration (2011-01-14 18:30)	18
	SvxLink-Features (2011-01-14 19:37)	19
	Details (2011-01-14 19:43)	19
	RepeaterLogic (2011-01-14 19:48)	19
	SimplexLogic (2011-01-14 19:52)	21
	Module (2011-01-14 19:53)	21
	DTMF-Repeater (2011-01-14 19:57)	22
	Parrot (2011-01-14 19:57)	23
	Echolink (2011-01-14 19:58)	24
	METAR (2011-01-14 19:58)	27
	TclVoiceMail (2011-01-14 19:59)	29
	Hilfe (2011-01-14 20:05)	31
	QTel-Client (2011-01-14 20:30)	31
	Tips und Tricks (2011-01-14 20:43)	34
	MicSpkr-Logic (2011-01-14 20:47)	34
	Propagation-Monitor (2011-01-14 21:07)	36
	Selektivrufencoder (2011-01-15 20:49)	38
	Konzept (2011-01-16 15:53)	39
	Inhalt (2011-01-16 16:07)	40
	Einführung (2011-01-16 16:09)	40
	Hilfe-Modul (2011-01-16 16:11)	41
	Parrot-Modul (Sprachpapagei) (2011-01-16 16:13)	42

Echolink-Modul (2011-01-16 16:15)	42
Connect-by-callsign (2011-01-16 16:19)	44
Mailbox-Modul (2011-01-16 16:22)	45
Mailbox: Nachrichten auslesen (2011-01-16 16:24)	45
Mailbox: Nachricht versenden (2011-01-16 16:25)	46
Mailbox: Nachschauen, ob Nachrichten bereitliegen (2011-01-16 16:26)	46
METAR-Modul (2011-01-16 16:27)	46
METAR: Suchfunktion 1 (2011-01-16 16:28)	47
METAR: Suchfunktion 2 (2011-01-16 16:35)	48
Selektivruf-Modul (2011-01-16 16:39)	49
Selektivruf-Dekoder (2011-01-16 16:46)	50
Propagationmonitor-Modul (2011-01-16 16:47)	50
Mehrere Qtel-Clients mit SvxLink (lokal) (2011-01-16 17:08)	51
Probleme und Hilfe (2011-01-16 17:15)	52
Fehler melden (2011-01-16 17:16)	53
Hilfe (2011-01-16 17:33)	54
PropagationMonitor (2011-01-16 18:06)	54
LocationInfo (APRS) (2011-01-16 18:36)	57
RepeaterLogic (2011-01-17 11:11)	58
SimplexLogic (2011-01-17 11:12)	63
remotetrx (2011-01-17 16:52)	66
Macros (2011-01-17 22:40)	68
Link(ing) (2011-01-17 23:03)	69
svxlink.conf (2011-01-18 20:30)	70
Routereinstellungen (2011-01-23 13:52)	79
Tcl-Routinen (2011-01-27 16:06)	80
Interfaces (2011-01-28 08:44)	81
Cradle (2011-01-29 12:45)	82
Analoges Voicemodem (2011-01-29 14:43)	84
GSM-Interface (2011-01-30 12:49)	85
1.2 Februar	86
Forum (2011-02-02 18:36)	86
1.3 Oktober	87
Internetstreaming vor Linux Kernel Version 2.6.37 (2011-10-17 20:42)	87

2	2012	91
2.1	Januar	91
	SvxLink für 16k-Sounds selber kompilieren (obsolete) (2012-01-09 18:39)	91
	Locale-Konzept (2012-01-18 13:09)	91
	Audioeinstellungen (2012-01-23 14:38)	92
2.2	Februar	94
	Unterschiedliche Roger-Beep's (2012-02-09 21:33)	94
	Trennung EchoLink-Verbindung verhindern (2012-02-14 12:46)	95
2.3	März	97
	Projekte (2012-03-10 20:37)	97
	SvxLink auf Pandaboard (2012-03-10 20:40)	97
2.4	April	99
	XTS3000 an SvxLink (2012-04-06 09:41)	99
2.5	Juni	101
	Parameter zur Laufzeit ändern (2012-06-06 13:41)	101
	Internetstreaming ab Linux Kernel Version 2.6.37 (2012-06-08 00:00)	103
	Linking RemoteTrx und SvxLink (2012-06-20 20:32)	105
	Internetstreaming (2012-06-28 12:50)	111
2.6	August	111
	(Unwetterwarnung) (2012-08-16 18:42)	111
	(Verkehrsinformation) (2012-08-17 07:37)	112
	(eMail-2-Rf) (2012-08-17 08:35)	114
	Persönliche Begrüßung eingehender EchoLink-Stationen (2012-08-19 16:53)	114
	SvxLink fernsteuern (2012-08-20 08:48)	115
	SvxLink Streaming & GNU Radio (2012-08-21 21:55)	115
	Selektivruf bei eingehender EchoLink-Verbindung auslösen (2012-08-22 21:26)	117
	Informationsansagen zeitlich gesteuert mit cron (2012-08-23 18:26)	118
2.7	September	121
	SvxLink auf Raspberry-Pi (2012-09-11 10:09)	121
2.8	Oktober	134
	Startskripte (2012-10-04 20:39)	134
3	2013	137
3.1	Mai	137
	Compileroptionen für ARM, Geode (und andere Prozessoren) (2013-05-29 14:53)	137
3.2	Oktober	137
	SvxLink mit Tetra-DMO-Repeater (Tetra im Amateurfunk) (2013-10-29 12:41)	137
	Audioprobleme beim Raspberry Pi (2013-10-19 23:14)	146
	HowTo (2013-10-22 07:37)	146

4	2014	149
4.1	Januar	149
	Einfaches EchoLink-Logbuch (2014-01-16 13:55)	149
4.2	Februar	150
	Bootfähiges Image Voyage-Linux (incl. SvxLink) für ThinClients (2014-02-28 15:19)	150
4.3	März	152
	Beeps bei EchoLink-Verbindung unterdrücken (z.B. I-Phone) (2014-03-02 08:16)	152
	SvxLink @ DBOHAL/DBOPET auf dem Petersberg (Stand: 21.8.14) (2014-03-23 09:46)	153
4.4	Juni	162
	Qso-Recorder (2014-06-24 19:19)	162
4.5	August	164
	Direkter SvxLink-2-SvxLink Verbund (SvxServer) (2014-08-12 08:35)	164
4.6	Dezember	167
	Zeitlich gesteuerte Ausgaben generieren (2014-12-10 09:37)	167
5	2015	173
5.1	Januar	173
	Lokaler Simplex-Link mit EchoLink (2015-01-16 08:45)	173
5.2	April	177
	GPIO-Konfiguration (2015-04-17 07:43)	177
	SvxLink-Adapter für BananaPi (2015-04-21 13:25)	178
6	2011	181
6.1	Januar	181
	/etc/rc.d/svxlink start (2011-01-20 17:27)	181
	PhoneLogic (2011-01-14 18:52)	181
	DTMF-Digit "DProblem (2011-01-30 10:19)	182
6.2	Februar	183
	Neuer Rauschsperrtyp EVDEV (2011-02-25 08:27)	183
6.3	Mai	183
	Release 11.05 veröffentlicht (2011-05-15 20:32)	183
6.4	August	184
	Automatische Installation von SvxLink (2011-08-25 16:53)	184
6.5	September	184
	SvxLink-Abend in Weißenfels (2011-09-19 13:03)	184
6.6	November	185
	Neues SvxLink-Release, Version 11.11 (2011-11-06 19:51)	185
	Neue SvxLink-Version 11.11.1 (2011-11-29 08:37)	185

7	2012	187
7.1	März	187
	SvxLink-Server unter Ubuntu auf einem Pandaboard (2012-03-10 15:56)	187
7.2	Juni	187
	Internetstreaming ab Kernel Version 2.6.37 (2012-06-08 07:56)	187
	Neuer Artikel Parameter zur Laufzeit ändern”(2012-06-03 08:32)	187
	SvxLink @ Hamradio FDH (2012-06-19 23:10)	187
7.3	August	188
	Neuer Branch "digitalmodes”(2012-08-08 13:01)	188
	SvxLink Streaming & GNU Radio (2012-08-22 11:34)	188
	Informationsansagen zeitlich gesteuert mit cron (2012-08-23 18:37)	189
7.4	September	189
	APRS Statistik im neuen Trunk (2012-09-07 10:29)	189
	Artikel SvxLink auf Raspberry-Pi (2012-09-24 10:47)	190
8	2013	191
8.1	Juni	191
	2. SvxLink-Treffen in Halle/Saale 15./16.6.2013 - abgesagt (2013-06-05 14:28)	191
9	2012	193
9.1	Dezember	193
	Frohe Weihnachten (2012-12-20 17:57)	193
10	2013	195
10.1	März	195
	Happy Birthday, SvxLink (2013-03-03 11:20)	195
	Professionelle Stimmen/Ansagen für SvxLink (2013-03-17 21:16)	195
	Neues im Trunk (2013-03-29 08:27)	196
10.2	Juni	196
	Neues Release 13.06 herausgegeben (2013-06-06 23:22)	196
	HAMRADIO 2013 (2013-06-10 20:49)	197
10.3	Juli	198
	SvxLink-Anfragen und mehr... (2013-07-18 18:10)	198
	Neues Release 13.07 (2013-07-30 07:19)	198
10.4	August	199
	SvxLink-Webbook 08/2013 (2013-08-28 09:31)	199
10.5	September	199
	make release”(2013-09-16 08:30)	199
10.6	Oktober	199

Unterstützung des Opus-Codecs im Trunk (2013-10-13 11:48)	199
SvxLink @ INTERRADIO 2013 (2013-10-25 08:36)	200
10.7 November	200
SvxLink @ INTERRADIO 2013 -Update- (2013-11-11 20:02)	200
TetraLogic für DMO-Repeater (2013-11-17 18:03)	201
Vorträge INTERRADIO 2013 (2013-11-30 17:49)	201
10.8 Dezember	201
Release 13.12 (2013-12-01 17:03)	201
Frohe Weihnachten und alles Gute für 2014 (2013-12-24 18:59)	202
11 2014	203
11.1 Januar	203
Änderungen (2014-01-30 10:28)	203
Freier Satz für Italienisch (2014-01-30 10:32)	203
11.2 Februar	203
HAMRADIO 2014 - Vorträge (2014-02-26 23:03)	203
11.3 März	204
Erweitertes Konzept für SQL,DTMF und PTT (2014-03-05 17:30)	204
11.4 Februar	205
Bootfähiges Image Voyage-Linux (incl. SvxLink) für ThinClients (2014-02-28 15:50)	205
11.5 Mai	205
Erweiterung um einen Control-Port (2014-05-31 19:58)	205
11.6 Juni	205
HAMRADIO-Vorträge von DK5HH und DL1HRC (2014-06-29 12:41)	205
11.7 Juli	205
APRS-SysOp Treffen Auersberg (2014-07-10 12:29)	205
11.8 September	207
SvxLink-Relaisverbund (2014-09-24 11:37)	207
11.9 November	207
Interradio 2014 Hannover -Vorträge- (2014-11-09 09:30)	207
11.10 Dezember	208
Eine frohe Weihnachtszeit (2014-12-10 15:56)	208
12 2015	209
12.1 März	209
Workaround bei EchoLink-Problemen (2015-03-23 06:21)	209

1. 2011

1.1 Januar

Impressum (2011-01-14 17:27)

Dies ist eine private Webseite. Informationen und Daten von dieser Webseite können ohne weitere Anfrage für non-commercial/GNU Projekte genutzt werden.

Alle Angaben und Informationen ohne jegliche Garantie, Umbauten und Installationen/Konfigurationen erfolgen auf eigenes Risiko!

Weitere Informationen: dl1hrc { at } svxlink . de, df1amb { at } svxlink . de oder DENIC

Haftungsausschluß:

Inhalte von Internetseiten, die von dieser Seite aus verlinkt wurden, können sich ändern, ohne dass der Webseitenbetreiber davon Kenntnis erhält. Wir distanzieren uns ausdrücklich von Inhalten und Gestaltungen von über Links zu erreichenden Seiten, die nach unserer Linksetzung verändert wurden und lehnen jede Verantwortung ab, sollten diese gegen geltende Gesetze, Rechtsvorschriften oder ähnliches verstoßen.

Installation (2011-01-14 17:38)

1) durch bereitgestellte Binärpakete

Einige Distributionen bieten fertig kompillierte Pakete für SvXLink an. Diese müssen Sie dann nur installieren.

Für OpenSuSE werden Binärpakete im hamradio-Repository bereitgestellt. An dieser Stelle herzlichen Dank an OM Walter (DL8FCL), der sich dieser Sache angenommen hat. Um SvXLink unter OpenSuSE 11.4 bis 13.2 zu installieren, fügen Sie einfach das entsprechende hamradio-Repository hinzu:

Repo für 11.4 http://download.opensuse.org/repositories/hamradio/openSUSE_11.4/

Repo für 12.1 http://download.opensuse.org/repositories/hamradio/openSUSE_12.1/

Repo für 12.2 http://download.opensuse.org/repositories/hamradio/openSUSE_12.2/

Repo für 12.3 http://download.opensuse.org/repositories/hamradio/openSUSE_12.3/

Repo für 13.1 http://download.opensuse.org/repositories/hamradio/openSUSE_13.1/

Repo für 13.2 http://download.opensuse.org/repositories/hamradio/openSUSE_13.2/

Dort finden sich auch noch mehr Programme für Funkamateure.

Für Ubuntu stellt Felix Lechner Pakete für verschiedene Ubuntu-Versionen zur Verfügung, siehe: <https://launchpad.net/felix.lechner/+archive/hamradio>

Um dieses Repository hinzuzufügen führen Sie aus:

```
sudo add-apt-repository ppa:felix.lechner/hamradio
```

Detailliertere Informationen zur Installation von Software über ppa's erhalten Sie auf der Ubuntu-Webseite: http://wiki.ubuntuusers.de/Paketquellen_freischaalten/PPA

2) oder für den Freak: ßu Fuß" durch Selberkompilieren

Achtung: Das Projekt ist zu Github gewechselt. Deshalb stehen nur da die neusten Versionen zur Verfügung!

Bitte auch hier unterscheiden, ob Sie ein Release installieren wollen oder einen Trunk. Weiterführende Informationen zu den Begriffen Release und Trunk finden Sie auf der Wiki-Seite zur Versionsverwaltung.

Für beide Varianten gilt prinzipiell der selbe Ablauf, es unterscheidet sich nur die Herkunft der Quellen.

Herunterladen können Sie sowohl den letzten Trunk als auch das Release der Einfachheit halber per svn. Installieren Sie zunächst git auf Ihrem PC. Unter openSUSE per Yast oder sudo zypper install git. Unter Ubuntu/Debian geben Sie ein sudo apt-get install git oder nutzen Sie das Software-Center.

- Herunterladen des Trunks

```
git clone https://github.com/sm0svx/svxlink.git
```

Es sind weitere Pakete erforderlich, unter Debian kann man diese mittels:

```
apt-get install gcc g++ make cmake libgcrypt-dev libgsm1-dev libsigc++-2.0-dev tcl-dev libspeex-dev libasound2-dev libpopt-dev libssl-dev libopus-dev groff
```

installieren. Eventuell gibt es Fehlermeldungen, falls einige Pakete bereits auf dem System installiert sind. Unter openSUSE kann die Installation mit yast oder zypper erfolgen, z.B.: zypper install gcc gcc-c++ make cmake libgcrypt-devel libgsm-devel alsa-devel libsigc++2-devel tcl-devel speex-devel libopus-devel libasound2-devel openssl-devel

Unter src finden Sie die c++-Dateien. Des Weiteren sind die folgenden Schritte erforderlich:

```
sudo useradd -g users -G tty,audio -c "SvxLink User" svxlink
sudo passwd svxlink
cd svxlink/src
mkdir build
cd build
cmake -DUSE_QT=OFF -DCMAKE_INSTALL_PREFIX=/usr
-Dsysconf_install_dir=/etc -DLOCAL_STATE_DIR=/var
-DCMAKE_BUILD_TYPE=Release ..
make
sudo make install
```

Voraussetzung für die o.g. ist natürlich, dass sich auf Ihrem PC eine c++-Entwicklungsumgebung befindet. Probleme und Fehlermeldungen werden mit Sicherheit auftreten, da Sie am Anfang sicherlich noch nicht alle erforderlichen Pakete installiert haben.

Regelmäßig Updates der SvxLink-Quellen einspielen und installieren

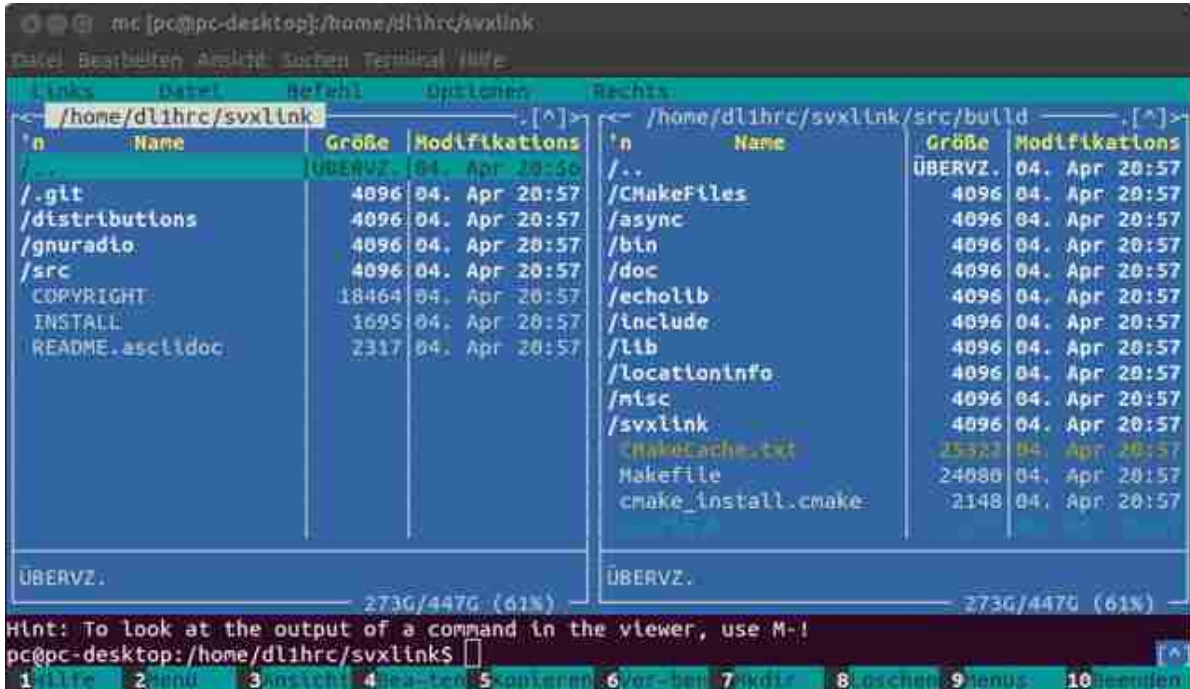
Möchte man auf die neueste Version updaten so wechseln Sie einfach in das Verzeichnis svxlink (also i. A. /home/Rufzeichen/svxlink) und geben den Befehl

```
git pull
```

ein. Danach führen Sie folgende Befehle aus

```
cd src/build
make clean
make
sudo make install
```

Nachfolgend mal eine typische Struktur für die Svxmlink-Entwicklung im mc dargestellt:



3) Installationskript für openSuSE

Ich habe mal ein Skript für openSuSE gebastelt, allerdings übernehme ich keine Garantie, dass es immer und überall problemlos funktioniert. Wer ein (möglichst neues) openSuSE laufen hat, kann gerne das Folgende ausprobieren, bitte als root ausführen:

```
wget http://svxmlink.de/download/svx-install.sh
chmod 750 svx-install.sh
./svx-install.sh
```

Das o.g. Skript installiert eine Entwicklungsumgebung (C++, Compiler, ...) sowie alle für die Installation benötigten Programme bzw. Bibliotheken. Weiterhin wird die deutsche Sprachumgebung installiert (bitte Lizenzierungsbedingungen beachten!). Nach Beendigung der Routine steht ein lauffähiges Svxmlink (Trunk-Version) auf dem aktuellen Stand zur Verfügung, Sie müssen nur noch Ihre Parameter wie Rufzeichen, Soundkarte, Beschreibungen etc. anpassen. Das Skript habe ich mehrfach auf dem BananaPi ausprobiert. Sie sollten auf Ihrem System allerdings noch ca 300MByte freien Speicherplatz besitzen.

Bitte nicht ungeduldig werden, je nach System und Performance kann die Installation 1h und mehr in Anspruch nehmen. Es gibt auch keinen bunten Fortschrittsbalken. Also einfach machen lassen!

Svxmlink wurde von Tobias (SMOSVX) unter Fedora Linux entwickelt. Es sollte sich einfach unter Fedora in-

stallieren lassen, da hierfür Binärpakete zur Verfügung stehen. Da es unter Fedora entwickelt wurde ist es auch unter dieser Distribution am besten getestet worden.

SvxLink hat einige wenige abhängige Pakete. Die meisten sollten bereits auf einer normalen Linux-Arbeitsstation installiert sein. Qtel benötigt X/Qt aber der SvxLink-Server ist eine reine Konsolenapplikation.

SvxLink liefert einige RPM-Pakete aus:

libasync

notwendig. Hauptbestandteil

echolib

benötigt für EchoLink

svxlink-server

der SvxLink-Server

qtel

Das Qt EchoLink-Client-Programm

xxx-devel Pakete

Entwicklungspakete. Diese werden nur benötigt wenn Sie selbst Programme erstellen wollen, die auf dem SvxLink-Projekt basieren

Hinweis: Für ALSA-basierte Systeme (wie Fedora >=2) wird zur Soundein- und Ausgabe die Alsa OSS-Emulation genutzt. Allerdings gibt es einen Fehler in der Emulationsschicht der bewirkt, dass SvxLink und Qtel nicht funktionieren. Um diesen Fehler zu umgehen wurde die Umgebungsvariable `ASYNC_AUDIO_NOTRIGGER` eingeführt.

Diese ist standardmäßig auf 1 gesetzt um diesen Fehler zu umgehen, da die meisten modernen Linuxdistributen standardmäßig ALSA nutzen. Falls Sie mit der Audioausgabe Probleme bekommen sollten, setzen Sie diese Variable auf 0. Für den SvxLink-Server können Sie das im Startskript `/etc/sysconfig/svxlink` machen.

Für Qtel können Sie diese Variable manuell in der Kommandozeile, in einem Startskript oder im Loginskript der Shell setzen. Die nachfolgende Anleitung setzt als Shell die Bash voraus die bei den meisten Linuxdistributen als Default-Shell eingestellt ist:

```
export ASYNC_AUDIO_NOTRIGGER=0
```

```
qtel &
```

Oder auf der Kommandozeile:

```
ASYNC_AUDIO_NOTRIGGER=0 qtel &
```

Die Umgebungseinstellungen gehen jedes Mal verloren, wenn Sie sich aus dem System ausloggen. Aus diesem Grund sollten Sie die oben aufgeführte Exportzeile in die Datei `~/.bash_profile` eintragen, welche Sie in Ihrem Home-Verzeichnis vorfinden. Das Setzen dieser Variable ist nicht zwangsläufig notwendig und kann dazu führen, dass SvxLink oder Qtel nicht ordnungsgemäß funktioniert. Setzen Sie deshalb diese Variable nur wenn Sie Audio-Probleme haben!

Hinweis: Beenden Sie alle anderen Applikationen, die auf die Soundkarte zugreifen und zur selben Zeit laufen wie SvxLink/Qtel. Wenn ein anderes Programm das Soundinterface bereits geöffnet hat, dann wartet SvxLink so lange bis diese Applikation beendet wird. Insbesondere im Falle von Problemen mit SvxLink, z.B. Äufhängen und weiteren Soundproblemen, prüfen Sie ob eventuell ein Soundserver wie z.B. artsd oder andere laufen. Wie dem auch sei, diese Problem ist nicht weiter wichtig, da die meisten Distributionen den ALSA-Sound-Layer benutzen. Wenn Sie ausschließlich Qtel nutzen möchten, so lesen Sie zuerst den Abschnitt Audio-Pegeleinstellungen und im Anschluß die Qtel-Dokumentation.

Hardware

Um den SvxLink-Server zu betreiben ist eine bestimmte Hardware notwendig um den Computer mit dem Funkgerät zu verbinden. Im Moment nutze ich ein Interface von Aleks, S54S, welches den PC vollständig vom Funkgerät mittels Optokoppler und Transformatoren trennt. Für lange Zeit habe ich ein einfaches, nicht isoliertes Interface betrieben welches auch funktioniert hatte. Allerdings habe ich auf diese Weise eine Soundkarte zerstört.

WBORXX/Tim hat eine Schaltung entwickelt, welche er zusammen mit SvxLink nutzt. Diese Schaltung kann man sich hier anschauen.

Auch die für Echolink entwickelten Interfaces können für SvxLink genutzt werden. Schauen Sie einfach auf die Echolink-Webseite. Getestet habe ich keine von diesen, deshalb kann ich keine Garantie auf Funktion geben. Bitte geben Sie mir eine kurze Rückinformation, ob diese Interfaces zusammen mit SvxLink funktionieren oder nicht.

Audiopegel einstellen

Es gibt weder in Qtel noch im SvxLink-Server die Möglichkeit den Audiopegel einzustellen. Die Pegel müssen mit Hilfe von aumix, kmix, alsamixer oder einer anderen externen Applikation eingestellt werden. Suchen Sie die Regler, die für die Einstellung der Pegel benötigt werden. Der NF-Ausgangspegel wird durch die Regler Pcm und Vol definiert und der Eingangspegel durch den Regler Capture (manchmal auch mit IGain bezeichnet), nicht aber der Mic bzw. line-in Regler! Die letzten beiden werden genutzt, um den Pegel für das Mithören der beiden Eingänge einzustellen.

Setzen Sie beide auf den Pegel 0. Wählen Sie entweder den Mic- oder den line-in Eingang aus.

Die Regler Pcm Vol und Mic/Input sollten bei etwa 50 % stehen. Stellen nun Sie die Pegel entsprechend der folgenden Anleitung ein.

Um die Pegel für Qtel einzustellen, verbinden Sie sich mit dem *ECHOTEST*-Server. Dieser EchoLinkserver sendet alles zurück, was er empfangen hat. Kurz nachdem die Verbindung aufgebaut wurde, hören Sie eine Begrüßung. Stellen Sie jetzt den Pegel mit Hilfe der Regler Pcm und Vol auf eine akzeptable Lautstärke ein. Betätigen Sie die PTT-Taste, sprechen Sie etwas und hören Sie dann wie das Gesprochene wiedergegeben wird. Verstellen Sie die Pegel des Capture-Reglers so lange bis Sie zufrieden sind.

Um die Audiopegel für den SvxLinkserver einzustellen, setzen Sie zunächst die Konfigurationsvariable PEAK_METER auf 1 im Abschnitt Empfängerkonfiguration der svxlin.conf.

Starten Sie den svxlink und drücken Sie * # auf Ihrer Tastatur. Dieses Kommando bewirkt, das sich SvxLink selbst identifiziert. Das Nummernzeichen # ist nur notwendig wenn das *-Kommando direkt auf der Tastatur eingegeben wird. Wenn dies per Funk machen möchten, so senden Sie ein * und das anschließende Schließen der Rauschsperr bewirkt den selben Effekt. Machen Sie das mehrere Male und stellen Sie die Regler für Pcm und Vol auf die höchstmöglichen Werte ohne dabei zu übersteuern.

Um den Audioeingangspegel einzustellen öffnen Sie die Rauschsperr des Empfängers, so dass SvxLink das Rauschen des Empfängers zugeführt wird.

Erhöhen Sie den Audioeingangspiegel so lange, bis Meldungen über Übersteuerung auf der Konsole erscheinen. Reduzieren Sie jetzt wieder den Pegel bis diese Meldungen nicht mehr ausgegeben werden. Falls diese Meldungen gar nicht erscheinen ist der Audiopegel insgesamt zu niedrig. In diesem Fall sollten Sie versuchen das Problem auf der analogen Eingangsseite zu lösen oder die PREAMP-Konfigurationsvariable im Abschnitt Empfängerkonfiguration entsprechend zu verändern.

Aktivieren Sie nun das Parrotmodul (Papagei) durch drücken von 1 # auf ihrer Tastatur. Senden Sie jetzt mit einem anderen Funkgerät etwas und hören Sie, ob das wiedergegebene Signal eine gute Qualität besitzt. Senden Sie einige DTMF-Zeichen und schauen Sie, ob diese auch ordnungsgemäß dekodiert werden. Falls nicht, dann verändern Sie die Audiopegel erneut.

Testen Sie alle 16 Zeichen: 0-9, *, #, A, B, C, D.

Als Regel für alle Einstellungen gilt, setzen Sie niemals die Regler auf Werte über 90 %. Bei den meisten Soundcards führt es zu Verzerrungen wenn der Pegel zu hoch ist. Falls der Audiopegel nicht ausreichend sein sollte so verändern Sie den Wert der Konfigurationsvariable PREAMP im Konfigurationsfile.

Download (2011-01-14 17:38)

Die Auto-Installations-CD von Martin (DF1AMB) kann man von hier herunterladen:

Download Iso-Image debian-svxlink-auto-install.iso Größe: 244416 kByte

ACHTUNG: Diese Installation läuft vollautomatisch ab, sie formatiert die 1.Festplatte und installiert ungefragt diese Distribution mit SvxLink! Bitte sicherstellen, dass auf dem Rechner keine wichtigen (ungesicherten) Daten vorhanden sind. Diese sind dann unwiederbringlich verloren! Die Installation erfolgt auf eigenes Risiko, eine Haftung wird nicht übernommen!

Weiterhin liegen hier ein paar SvxLink-Binaries für verschiedene Distributionen zum Download bereit: Achtung, hierbei handelt es sich um eine neuere Trunk-Version mit neuen Features.

Leider hat das Programm checkinstall", welches ich zum Erstellen der *.deb-Pakete verwende einen Bug, so dass es zu Problemen beim Installieren der *.deb-Pakete kommt, wenn vorher noch kein svxlink auf dem System installiert war. Abhilfe: Bitte das Skript

<http://svxlink.de/download/svxlink-pre-deb.sh>

als root auf den betreffenden Systemen vor der Installation des *.deb-Paketes starten.

Ubuntu:

<https://launchpad.net/felix.lechner/+archive/hamradio>

svn-Version von Martin (DF1AMB), Rev. 2005:

32 bit-Version (svxlink_0.svn2005_i386.deb)

OpenSuse 11.3:

Für OpenSuSE 11.3 und 11.4 nutzen Sie bitte das hamradio-Repository, siehe Installation

Synthetische Mary-Soundfiles (deutsch, männlich und weiblich, verschiedene Sampleraten):
stimmen-mary.tar.bz2 (110.8MByte)

Entpacken Sie die Datei z.B. nach /tmp . diese Datei enthält verschiedene Unterverzeichnisse, z.B. 16k-female, 22.05k-male, usw...

Kopieren Sie das für Ihre svxlink-Version gültige Unterverzeichnis nach /usr/share/svxlink/sounds und benennen sie es nach de_DE um, so dass sich eine Struktur wie folgt ergibt:

```
/usr/share/svxlink/sounds/de_DE/Core  
/usr/share/svxlink/sounds/de_DE/Default  
/usr/share/svxlink/sounds/de_DE/DtmfRepeater  
/usr/share/svxlink/sounds/de_DE/EchoLink  
...  
usw.
```

Diese Stimmen sind OpenSource-Stimmen (Mary, siehe <http://mary.dfki.de/>), welche qualitativ nicht optimal sind. Es ist auch ein qualitativ hochwertiger Satz vorhanden, dieser setzt allerdings das Vorhandensein einer kostenpflichtigen Lizenz voraus. Falls daran Interesse bestehen sollte, bitte Kontakt mit uns aufnehmen.

Weitere Erklärungen zu den Begriffen Release,Trunk und Branch können Sie auf der Wikipedia-Seite nachlesen.

Allgemeines (2011-01-14 17:38)

Die folgenden Seiten sollen Interessierten den Einstieg in SvxLink vereinfachen. Für Fragen und Hinweise nutzen Sie bitte unser (im Aufbau befindliches) Forum.

SvxLink ist eine OpenSource-Software, die jeder frei nutzen darf und unter der GNU Public License veröffentlicht wird.

OpenSource bietet ungeheure Potentiale, erwartet aber im Gegenzug vom Nutzer auch ein gewisses Entgegenkommen und die Bereitschaft sich in bestimmte Problemstellungen "hineinzulesen".

Wer hier das 1-Klick-Fertig-Installations-Programm erwartet, sollte seine Zeit lieber anderweitig nutzen!

73's

Den Inhalt dieser Domain gibt es auch als PDF-Webbook zum Download: SvxLink.de Webbook Stand 11/2012 (v2)

Links (2011-01-14 17:40)

Stationen / Hams, die SvxLink betreiben:

- SM0SVX - The big cheese ;-) <http://www.ludd.luth.se/blomman/>
- DB0WA <http://www.db0wa.de>
- DB0DIN <http://www.db0din.de/>
- DB0PER <http://www.db0per.de/>
- DB0BLO <http://www.ov-d20.de/smb.htm>
- DB0HHW <http://db0hww.blogspot.com/>
- DB0TUD <http://www.qsl.net/dh1dm/db0tud/>
- DB0HRC <http://www.darc.de/distrikte/w/34/db0hrc-bad-duerrenberg/>
- DB0WSF <http://www.darc.de/distrikte/w/34/relaisstation-db0wsf/>
- DB0DBT <http://www.darc.de/distrikte/w/34/db0dbt-allstedtflugplatz/>
- DB0HFT <http://www.db0hft.de/>
- DL1BIR <http://www.mydarc.de/dl1bir/>
- DB0HDF <http://www.db0hdf.de/>
- DB0HSN <http://df0fn.hsnr.de/db0hsn/db0hsn.htm>
- DM0LEI <http://wiki.dh8tom.net/doku.php?id=dm0lei>
- DB0CSD <http://www.dm2kl.de/relais.html>
- DB0ZD <http://www.db0zd.de.vu/>
- DB0LHR <http://www.funkamateure-lahr.de/htm/db0lhr.htm>
- DB0MYK <http://prgm.org/db0myk>
- DB0FBG <http://www.qsl.net/dl0fg/db0fbg.html>
- DL0RI <http://www.avsk.net/165.html>
- DB0PB http://db0ug.ulrich-klute.de/index.php?option=com_co...
- DB0BUL http://www.dl8rds.de/index.php/The_Echolink_Node_DB0BUL
- DB0TIT <http://www.ov-a34.de/index.php/echolink>
- DB0KOE <http://www.db0koe.de/cms17/.../16-23cm-demnaechst-mit-svxlink>
- DO3DT <http://home.arcor.de/db0dxm/index.html>
- OK1PJV <http://www.cbradio.cz/echolink/ok1pju-l.php3>
- OE5XYP <http://www.adl509.at/.../6-meter-umsetzer/oe5xyp/>
- OE5XOL http://wiki.oevsv.at/index.php/70cm_Relais_OE5XOL
- PE1JLX <http://www.splint.nl/>

- SK3GW <http://sk3w.se/sk3gw-repeater>
- SLOZS <http://www.stockholm.fro.se/paketradio/slozs/index.html>
- S50EDX http://www.s50e.si/?page_id=71
- S55VTO <http://rpt.hamradio.si/?modul=galerije &site=0 &id=0-39->
- TA3CP <http://ta3cp.blogspot.de/search/label/svmlink>
- F1SMF <http://f1smf.free.fr/articles.php?lng=fr &pg=20>
- ONOLN http://www.on8ar.be/index.php?option=com_content &...
- WB0RXX <http://www.hacem.org/svmlink.html>
- 4Z7GAI <http://guysoft.wordpress.com/2011/08/08/svmlink/>
- KJ6ZD-L <http://www.svmlink.net>

Weiterführende Informationen:

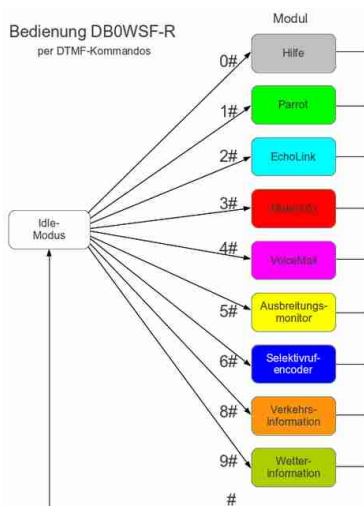
- SvxLink.de Webbook Stand 08/2013
- http://www.dl8rds.de/index.php/A_SVXLink_Server_based_on_the_BeagleBoard
- Die offizielle SvxLink Seite bei Sourceforge (Engl.)
- Vortrag von Prof.-Dr. Michael Hartje zum Thema "Digitale Amateurfunkrelais auf der HAMRADIO 2012"
- Youtube Video, Prof. Dr. Michael Hartje / DK5HH, HAMRADIO 2012
- Linux-Livesystem mit SvxLink - Vollständiges Linuxsystem mit graphischer Oberfläche und bereits enthaltenem SvxLink
- Echolink bei Wikipedia <http://de.wikipedia.org/wiki/Echolink>
- Echolink für Windows

Bedienungsanleitung (2011-01-14 17:52)

Die Bedienungsanleitung wurde von Martin (DF1AMB) erstellt und erhebt keinen Anspruch auf Vollständigkeit.

!!!!ACHTUNG!!!!

Die jeweiligen (Sub-)Kommandos und die Verfügbarkeit von Modulen sind natürlich abhängig von der jeweiligen Konfiguration auf dem Link bzw. Repeater und ggf. beim jeweiligen Sysop zu erfragen.



SvxLink-Bedienung (konfigurationsabhängig, hier DB0WSF-R)

Konfiguration (2011-01-14 18:30)

Dies ist eigentlich der wichtigste Abschnitt, die Konfiguration von SvxLink. Von vielen Rückmeldungen der Nutzer wissen wir, dass hier teilweise erheblicher Klärungsbedarf besteht. Vielleicht können die folgenden Seiten einige Fragen klären und Ihnen eine Hilfestellung bei der Einrichtung Ihres SvxLink-Nodes bieten. Achten Sie bitte darauf, dass es für jedes einzelne Modul eine eigene Konfigurationsdatei gibt, diese Dateien befinden sich bei einer Standardinstallation unter `/etc/svxlink/svxlink.d`

Es ist also normalerweise nicht damit getan die `svxlink.conf` einzurichten, sondern Sie müssen danach in Abhängigkeit der von Ihnen konfigurierten Module auch noch diese einzeln einrichten!

Falls Fragen nicht beantwortet werden, so nutzen sie bitte das Forum, hier können spezifische Probleme oft viel schneller geklärt werden.

Verzeichnisstruktur

`/etc/svxlink/etc/rc.d/` ...die System-Startskripte `svxlink` und `remotetrx` `/usr/bin/` ...hier sind die Binaries zu finden (`svxlink`, `qtel`, `remotetrx`, `siglevdetcal`) `/usr/include/svxlink` enthält die Async-Header-Dateien (z.B. `AsyncAudioSink.h`) `/usr/lib/` ...die SvxLink-core-Bibliotheken wie z.B. `libasyncaudio-0.17.0.99rc1.so` `/usr/lib/svxlink` ...die SvxLink-Modul-Bibliotheken wie z.B. `ModuleMetarInfo.so` `/usr/share/qtel` ...die Translationfiles (Übersetzungen in 8 Sprachen) für Qtel `/usr/share/svxlink` ...die `events.tcl`, das "Hauptskript" der Eventbehandlungsroutinen `/usr/share/svxlink/events.d` ...hier liegen die logik- und modulspezifischen tcl-Files, die sozusagen die Präsentationsebene bilden, also Sprachausgaben generieren, Repeatererkennung als Morsezeichen ausgeben, etc. `/usr/share/svxlink/modules.d` ...hier liegen weitere TCL's für SvxLink-Module `/usr/share/svxlink/sounds` ...die Soundfiles für die Sprachausgaben in weitere landes- bzw. sprachspezifische Unterverzeichnissen wie z.B. `de_DE/` oder `en_EN/`. Wird im Linuxsystem keine Ländereinstellung erkannt bzw. wurde es nicht explizit im Logik-Abschnitt definiert, erfolgt defaultmäßig die Suche in `en_US/` `/var/log` Hier liegt standardmäßig das SvxLink-Logfile `svxlink`, dass über Ereignisse und Fehler informiert. Wird SvxLink in der Konsole gestartet, erfolgen die Ausgabe nicht in diesem File sondern in der Konsole selbst. `/var/spool/svxlink` Spool-Verzeichnis für Mails, Sprachmailbox, Voice-Mitschnitte usw., wenn der QSO-Recorder aktiviert oder der PropagationMonitor konfiguriert wurde.

SvxLink-Features (2011-01-14 19:37)

Auf diese Seite finden sie eine Übersicht über die Funktionen des SvxLink-Servers sowie Qtel. Sie dient der schnellen Übersicht, wofür die Programme genutzt werden können.

Der SvxLink-Server

Das sind die wichtigsten Funktionen des SvxLink-Servers:

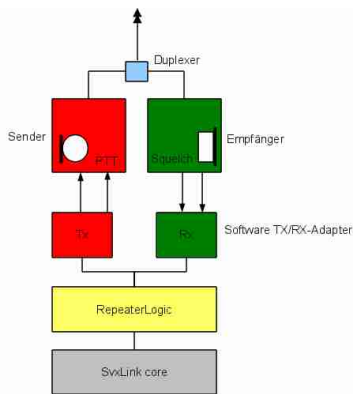
- # Kann als Relais-Controller fungieren oder als einfacher Simplex-Link
- # Modulbasiertes System, um Sprachservices in den Programmkern nachladen zu können
- # steuerbar per DTMF-Töne
- # DTMF - Anti-Flutter-Modus, um bei Mobilbetrieb und/oder schwachen Signalstärken einen DTMF-Ton nicht mehrfach zu erkennen
- # Sound-Dateien für Ansagen und Informationen
- # Periodische Identifizierung frei programmierbar
- # Ein TCL-basiertes System, das bestimmte Ereignisse speziell behandeln kann
- # Stereo-Kanäle können wie einzelne Mono-Kanäle behandelt werden
- # Roger-Piep
- # Zusammenschaltung von verschiedenen Logiken (Repeater, Links) per Kommando
- # Multi-Tone-Encoder nach ZVEI-xx, CCITT, EIA, Natel, Motorola, ...
- # Multi-Tone-Decoder als Macro nutzbar
- # Manuelle Identifizierung beim DTMF-Ton *
- # einfaches Makrosystem für Kurzwahlfunktionen
- # APRS-Status- und Baken-Funktionen
- # RemoteTRX erlaubt Vernetzung von verschiedenen RX und TX
- # Unterstützung von EchoLink-Proxyservern

Details (2011-01-14 19:43)

Hier sind Informationen zu finden, wie Sie SvxLink installieren und konfigurieren können. Für spezifische Fragen und bei Problemen nutzen Sie bitte unser im Aufbau befindliches Forum.

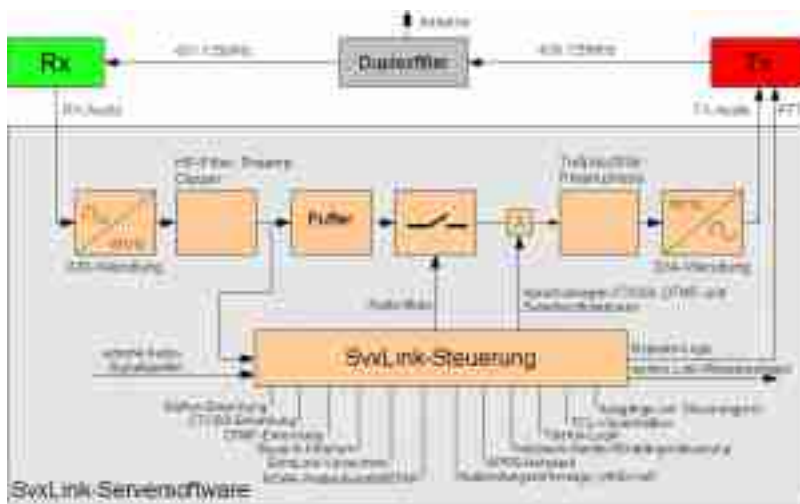
RepeaterLogic (2011-01-14 19:48)

Das folgende Bild zeigt die prinzipielle Funktionsweise einer RepeaterLogic.



Stark vereinfachtes Prinzip einer RepeaterLogic

Im folgenden Bild sind die Aufgaben etwas detaillierter dargestellt. Nicht alle Funktionen sind für eine Repeater-Logic notwendig. Es soll nur den möglichen Funktionsumfang von SvxLink verdeutlichen.



Prinzip der Funktionsweise/Möglichkeiten einer RepeaterLogic am Beispiel von DB0WSF

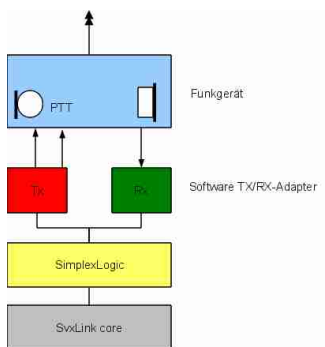
Funktionen der RepeaterLogic

- # verschiedene Aktivierungsmöglichkeiten wie 1750Hz-Ton, CTCSS, DTMF oder per Rauschsperr
- # Periodischer "idle-tone" wenn das Relais aufgetastet ist aber kein Signal empfangen wird
- # Ansagen werden mit der normalen Sprachübertragung gemischt
- # Die Lautstärke der Ansagen wird bei anderen Sprachübertragungen vermindert
- # Möglichkeit, mehrere Kernprogramme zu definieren um mehrere Funkgeräte anzuschließen
- # Kernprogramme können miteinander verbunden werden
- # Unterstützung für mehrere Empfänger, die über eine eingebaute Software automatisch zu- oder abgeschaltet werden können (Auswahl des RX mit bester Signalqualität)
- # Netzwerk-Empfänger können per TCP/IP mit dem Kernprogramm verbunden werden
- # Netzwerk-Sender können per TCP/IP mit dem Kernprogramm verbunden werden
- # Mehrere Squelsh-Detektoren: VOX, CTCSS, Signalerkennung über die serielle Schnittstelle
- # Höhenanhebung/-absenkung auf dem Übertragungsweg
- # DTMF-Unterdrückung
- # Rauschenunterdrückung vor dem Einsetzen der Rauschsperr
- # Senden von CTCSS-Subaudiotönen, ständig oder abhängig vom Status der Rauschsperr
- # einfaches Hilfemodul in englischer/deutscher Sprache
- # Parrot-Modul, das das empfangene wieder aussendet ("Papagei")
- # Echolink-Modul das die Teilnahme am weltweiten Echolink-Netz ermöglicht
- # Leerlauf-Zeitbegrenzung
- # Rufzeichensuche bzw. Verbindung durch Rufzeicheneingabe

- # Auflistung verbundener Stationen durch Kommando
- # Funktion zur Ansage der eigenen Node-ID
- # Verbindungsaufnahme zu Relais, Konferenzen oder Links per Zufall möglich
- # Wiederverbinden mit der letzten Station
- # SWL-Modus (nur Zuhören)
- # Öffnen des Repeaters durch Squelch, Rufton, CTCSS, Mehrton(ZVEI,EIA,...) und/oder DTMF

SimplexLogic (2011-01-14 19:52)

Eine SvxDLink-SimplexLogic entspricht einem Link (DL1ABC-L) bei Echolink. Kennzeichnend ist der reine Simplexbetrieb (wie der Name schon beschreibt).



Prinzipielle Darstellung der SimplexLogic

Ein einzelnes Funkgerät wird an die Soundkarte (mit Hilfe eines Adapters) angeschlossen. Die PTT-Ansteuerung liefert die TX-Logic mit. Schauen Sie unter *Details->Konfiguration->SimplexLogic* wie Sie solch einen einfachen SvxDLink-Node konfigurieren können.

Module (2011-01-14 19:53)

Das Konzept von SvxDLink unterscheidet sich im Vergleich zur herkömmlichen Echolink-Software nicht unerheblich. SvxDLink beinhaltet verschiedene Module, die beliebig aktiviert bzw. deaktiviert werden können. Nur ein Modul davon ist das Modul EchoLink mit der bekannten Funktionalität.

Auf den folgenden Seiten wird erklärt, wie Sie ein Modul zur Konfiguration hinzufügen können. Der hier verwendete Syntax:

```
[RepeaterLogic]
```

```
...
```

```
MODULES=...,ModuleDtmfRepeater,...
```

...

soll darlegen, dass an den Stellen mit "...unter Umständen schon zusätzliche Parameter konfiguriert worden sind, wie z.Bsp:

```
MODULES=...,ModuleDtmfRepeater,...
```

ist gleichbedeutend mit

```
MODULES=ModuleHelp,ModuleDtmfRepeater,ModuleMetarInfo,ModuleEchoLink
```

DTMF-Repeater (2011-01-14 19:57)

Wenn dieses Modul aktiviert ist, sendet es alle empfangenen DTMF-Zeichen wieder aus. Dabei werden bis auf das #-Symbol alle DTMF-Zeichen bzw. Kommandos vom SvxLink-Kernsystem ignoriert. Das #-Symbol deaktiviert das DtmfRelais-Modul

Dieses Modul wird üblicherweise bei einem SvxLink-Relais genutzt um andere Dienste fernzusteuern. Die DTMF-Stummschaltung sollte dabei aktiviert sein, damit die DTMF-Töne nicht über das Relais ausgesendet werden.

Aktivierung

in der /etc/svxlink/svxlink.conf durch Hinzufügen des Eintrages

```
[RepeaterLogic]
```

...

```
MODULES=...,ModuleDtmfRepeater,...
```

...

Beispielkonfiguration in der /etc/svxlink/svxlink.d/ModuleDtmfRepeater.conf

```
[ModuleDtmfRepeater]
```

```
NAME=DtmfRepeater
```

```
PLUGIN_NAME=DtmfRepeater
```

```
ID=7
```

```
TIMEOUT=5
```

```
REPEAT_DELAY=500
```

Beschreibung

NAME

Name des Moduls

ID

Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig

vergeben werden.

TIMEOUT

Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

REPEAT _DELAY

Pausenzeit in Millisekunden, nach der die empfangenen Informationen wieder ausgesendet werden.

Parrot (2011-01-14 19:57)

Der Sprachpapagei

Das Parrot-Modul gibt alles wieder was es selbst hört. Es kann als Simplex-Repeater genutzt werden oder um einfach zu prüfen, wie sich die eigene Aussendung über Funk anhört. Weiterhin sagt es die DTMF-Zeichen an, die man selbst gerade gesendet hat. Wie immer sind alle Eingaben durch das Nummernsymbol zu bestätigen. Dieses Modul wird durch das Senden eines einzelnen Nummern-Symbols (#) wieder verlassen.

Aktivierung

in der /etc/svxlink/svxlink.conf durch

```
[RepeaterLogic]
```

```
...
```

```
MODULES=...,ModuleParrot,...
```

```
...
```

Beispielkonfiguration in der /etc/svxlink/svxlink.d/ModuleParrot.conf

```
[ModuleParrot]
```

```
NAME=Parrot
```

```
ID=1
```

```
TIMEOUT=60
```

```
FIFO_LEN=60
```

```
REPEAT _DELAY=1000
```

Beschreibung

NAME

Name des Moduls

ID

Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig vergeben werden.

TIMEOUT

Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

REPEAT _DELAY

Zeit in Millisekunden nach der der gespeicherte Audiostream wieder ausgegeben wird.

Echolink (2011-01-14 19:58)

Das Modul Echolink ermöglicht die Verbindung mit anderen Echolink-Stationen über das Internet.

Das Echolink-Modul wird genutzt um eine Verbindung mit anderen Echolinkstationen herzustellen. Um eine Verbindung herzustellen geben Sie einfach die Node-Nummer der Station gefolgt von einer Raute (#) ein. Um die Verbindung wieder zu trennen senden Sie eine einzelne Raute (#).

Um das Echolink-Modul wieder zu verlassen drücken Sie erneut #.

Es gibt eine Reihe von Unterkommandos, die innerhalb des Echolink-Modules genutzt werden können.

- 1 # - Zeigt alle verbundenen Stationen
- 2 # - Ausgabe der lokalen Echolink Nodenummer
- 31 # - Verbindung mit einem zufällig gewählten Link oder Repeater
- 32 # - Verbindung mit einer zufälligen Konferenz
- 4 # - mit der Station wiederverbinden, die zuletzt getrennt wurde
- 50 # - nur RX-Modus deaktivieren
- 51 # - nur RX-Modus aktivieren

Die Funktion per Rufzeichen verbinden ermöglicht es die Verbindung zu einer Station herzustellen, deren Nodenummer nicht bekannt ist. Dabei werden die Rufzeichen auf Ziffern nach folgender Methode abgebildet:

ABC=2, DEF=3, GHI=4, JKL=5, MNO=6, PQRS=7, TUV=8, WXYZ=9. Es handelt sich um die selbe Abbildung wie bei den meisten Telefonen.

Buchstaben werden auf die zugehörigen Zeichen abgebildet und diese entsprechen den zugehörigen Nummern.

Alle anderen Zeichen werden der Ziffer 1 zugeordnet. Ein Such-Kommando, beginnt mit einem Stern, danach folgt der Rufzeichen-Code. Wenn Sie also nach dem Rufzeichen SM3SVX-L suchen so senden Sie `"*76378915 #"`;

Da die Codes nicht eindeutig sind, wird Ihnen eine Liste gefundener Stationen angeboten, die dem gesuchten Muster entsprechen und aus denen Sie auswählen können. Das Stern-Zeichen entspricht dabei einer Art Joker, d.h. ein beliebiges Zeichen. Wenn Sie beispielsweise alle Stationen mit der Zeichenfolge SSM3Buch, dann senden Sie einfach `"*763* #"`.

Für weitere Informationen zum Echolinksystem schauen Sie bitte auf die EchoLink-Webseite <http://www.echolink.org>.

Aktivierung

in der `/etc/svxlink/svxlink.conf` durch Hinzufügen des Eintrages

```
MODULE=...,ModuleEchoLink,...
```

```
Konfigurationsdatei /etc/svxlink/svxlink.d/ModuleEchoLink.conf  
[ModuleEchoLink]
```



```

NAME=EchoLink
ID=2
TIMEOUT=60
ALLOW_IP=192.168.0.0/24
#DROP_INCOMING=^(.*) $
#REJECT_INCOMING=^(.*) $
#ACCEPT_INCOMING=^(.*) $
#REJECT_OUTGOING=^(.*) $
#ACCEPT_OUTGOING=^(.*) $
#CHECK_NR_CONNECTS=3,300,120
SERVERS=europe.echolink.org servers.echolink.org
CALLSIGN=DB0ABC-R
PASSWORD=*****
SYSOPNAME=Your name
LOCATION=[Svx] City
#PROXY_SERVER=
#PROXY_PORT=
#PROXY_PASSWORD=
#BIND_IP=192.168.0.10
#REJECT_CONF=1
USE_GSM_ONLY=1
MAX_QSOS=6
MAX_CONNECTIONS=7
#AUTOCON_ECHOLINK_ID=
#AUTOCON_TIME=
#LINK_IDLE_TIMEOUT=60
LOCATION_INFO=LocationInfo
DESCRIPTION="You have connected to a SvxLink node,\n"
"ä voice services system for Linux with EchoLink\n"
"ßsupport. For more info check out:\n"
"http://sourceforge.net/apps/trac/svmlink/\n"
"\n"
Locator: AB12cd\n"
"QRG : 438.700 MHz (-7.6MHz)\n"
CTCSS : NONE\n"
Äntenna: X50\n"
Öutput : 15W EIRP\n"

```

Beschreibung

NAME

Name des Moduls

ID

Die eindeutige ID unter der das Modul aktiviert wird

TIMEOUT

Zeit in Sekunden, nach der das Modul deaktiviert wird

ALLOW_IP

Netzwerkadresse aus der ein lokaler Connect erlaubt wird, z.B. ALLOW_IP=192.168.0.0/24

DROP_INCOMING

Regex-Syntax, EchoLink-Stationen deren Connect-Anfragen ignoriert werden (keine Information an die Gegenstation)

REJECT _INCOMING

Regex-Syntax, EchoLink-Stationen, deren Connectanfragen abgewiesen werden (Information an die Gegenstation)

ACCEPT _INCOMING

Regex-Syntax, EchoLink-Stationen, deren Connectanfragen akzeptiert werden.

ACCEPT _OUTGOING

Regex-Syntax, wenn definiert, dann werden nur zu den Stationen aufgebaut, die der Regex-Definition entsprechen

REJECT _OUTGOING

Regex-Syntax, wenn definiert wird der (ausgehende) Verbindungsaufbau unterbunden.

REJECT _CONF

Wenn =1, dann werden alle eingehenden Verbindung deren Namen CONF am Ende enthalten abgewiesen. Dabei wird an die anfragende Station eine Information gesendet, dass die Verbindung nicht aufgebaut wird. Es werden alle Stationen im Konferenz-Modus geblockt. Verbindet sich eine Station mit einer Konferenz nach dem die Verbindung zur eigenen Station bereits aufgebaut wurde, so wird die Verbindung zu dieser Station unterbrochen (disconnected).

CHECK _NR _CONNECTS

Wenn eine Station zu oft connected, so kann die Anzahl der Versuche begrenzt werden, ein erneuter Connectversuch wird dann abgewiesen. Beispiel CHECK _NR _CONNECTS=3,300,120

Baut eine Station 3x innerhalb von 300 Sekunden eine Verbindung zur eigenen Station auf, so wird diese Station für 120Minuten gesperrt

SERVERS

Leerzeichenseparierte Liste von Echolink-Servern. Wird die Verbindung zum Ersten getrennt, wird automatisch mit dem nächsten EchoLink-Server verbunden:

SERVERS=europe.echolink.org asia.echolink.org

CALLSIGN

Das Rufzeichen unter dem der eigene Node arbeitet (-L = Link, -R = Repeater), z.B. CALLSIGN=DB0ABC-R

PASSWORD

Das Passwort für die Anmeldung beim EchoLink-Verzeichnisdienst im Klartext(!)

SYSOPNAME

Der Name des Sysops

USE _GSM _ONLY

Auf Systemen mit schwacher CPU kann es sinnvoll sein, die automatische Codec-Umschaltung (von GSM nach SPEEX) zu verhindern, die standardmäßig zwischen SvxLink-Nodes ausgeführt wird. SPEEX benötigt eine höhere Rechenleistung, die auf schwachen Systemen unter Umständen zu unterbrochener Audioausgabe führen kann.

LOCATION

Die Name der Stadt/Dorfes wo sich der Link bzw. Repeater befindet, max. 27 Zeichen

MAX _QSOS

Die maximale Anzahl gleichzeitiger Verbindungen

MAX_CONNECTIONS

Wenn mehr Stationen als MAX_QSOS eine Verbindung aufbauen, so wird die Verbindung nur angenommen um der Gegenseite mitzuteilen, dass die Verbindungsabfrage auf Grund zu hoher QSO-Teilnehmerzahl abgelehnt wird. Bei mehr Anfragen als in MAX_CONNECTIONS definiert werden die Anfragen ignoriert.

PROXY_SERVER

Wenn gesetzt, erfolgt der Verbindungsaufbau zum konfigurierten EchoLink-Proxyserver. Aus- und eingehende Verbindungen werden ab sofort darüber geleitet.

PROXY_PASSWORD

Das Passwort für den EchoLink-Proxyserver

PROXY_PORT

Der Netzwerkport über den der Proxyserver erreicht wird.

BIND_IP

Mit diese Konfigurationsvariable kann der EchoLink-UDP-Socket an eine spezielle IP-Adresse/-Interface gebunden werden. Diese Konfiguration kann sinnvoll sein, wenn der Computer mehrere Netzwerkinterfaces besitzt.

Beispiel: BIND_IP=192.168.0.10

AUTOCON_ECHOLINK_ID

Konfigurieren Sie hier eine EchoLink-Nodenummer des EchoLink-Nodes zu dem eine Verbindung automatisch aufgebaut werden soll.

AUTOCON_TIME

Zeit in Sekunden zwischen zwei Verbindungsversuchen zum gewünschten EchoLink-Node

LINK_IDLE_TIMEOUT

Zeit, nach der die Verbindung zur Gegenstation abgebaut wird, wenn keine Aktivität feststellbar ist.

DESCRIPTION

Eine längere Beschreibung, die der Gegenseite Informationen zur eigenen Station übermitteln, z.B. CTCSS, Antenne, TRX, ...

METAR (2011-01-14 19:58)

Das METAR-Modul ermöglicht das Abrufen von aktuellen Wetterinformationen, i.A. von Wetterstationen, die sich an oder in der Nähe von internationalen Verkehrsflughäfen befinden entsprechend der Konfiguration des SvxLink-Administrators.

Kommando Funktion

0 # Hilfe allgemein

01 # Aufzählung konfigurierter Wetterstationen

1 # 1. konfigurierte Wetterstation

2 # 2. konfigurierte Wetterstation

3 # 3. konfigurierte Wetterstation

Xx # Xx. Wetterstation

Für weitere Informationen zu METARs siehe Metar bei Wikipedia.

Aktivierung in der /etc/svxlink/svxlink.conf durch
[RepeaterLogic]

...
MODULE=...,ModuleMetarInfo,...
...

Konfigurationsdatei /etc/svxlink/svxlink.d/ModuleMetarInfo.conf

[ModuleMetarInfo]

NAME=MetarInfo

ID=3

TIMEOUT=60

#STARTDEFAULT=EDDP

LONGMESSAGES=1

REMARKS=1

DEBUG=1

insert ICAO airport shortcuts here. You can

request the METAR by sending dtmf commands as follows

1 # 2 # 3 # 4 # 5 # 6 # 7 # 8 # 9 # 10 # 11 # 12 # 13 # 14 # 15 # 16 # 17 # 18 # 19 # 20

AIRPORTS=EDDP,EDDS,EDDM,EDDK,EDDV,LEMD,ESSB,KJAC,KTOL,ZBAA,KJFK,SKBO

,SKSM,EDJA,ZM-

UB,EGLL,LEPA,EDDH,LIRF,LIRN

Beschreibung

NAME

Der Name dieses Modules, z.B. *NAME=MetarInfo*

ID

Die eindeutige Identifikationsnummer, die für jedes SvxLink-Modul eindeutig vergeben werden muss. Die Aktivierung des Modules erfolgt bei *ID=3* mit dem Kommando *3 #*

TIMEOUT

Das Modul beendet sich nach der konfigurierten Zeit (in Sekunden) selbstständig, wenn keine Aktivität erfolgte.

STARTDEFAULT

Beim Aktivieren des METAR-Modules werden die Wetterdaten der konfigurierten Station sofort abgerufen und ausgegeben.

LONGMESSAGES

Bei bestimmten METAR-Informationen wird nicht die Kurzform ausgegeben sondern die vollständige Beschreibung des Tags, z.B. statt *NOSIG*> no significant change oder *CAVOK*> ceiling and visibility OK"

Für weitere Informationen schauen Sie bei Wikipedia unter METAR.

REMARKS

Wenn *-1*", dann werden auch die METAR-Abschnitte im RMK (Remark)-Bereich ausgegeben. Hier sind u.a. Landbahninformationen, Wettertrends usw. zu finden. Leider gibt es hier keinen durchgängigen Standard, der von allen Stationen eingehalten wird. Deshalb hier keine Garantie, dass jedes Tag auch ordnungsgemäß ausgegeben wird.

DEBUG

Wenn "-1", dann erfolgt ein erweitertes Loggen im svxlink-Log, so können Fehler beim Abruf der Wetterinformationen analysiert werden.

AIRPORTS

Kommaseparierte Liste von Wetterstationen, die per Kommando abgerufen werden können. Der Abruf erfolgt in der Reihenfolge der Konfiguration. z.B.

`AIRPORTS=EDDP,EDDM,EDDS,EDDH`

Bei dieser Konfiguration kann nach Aktivierung des METAR-Moduls und dem Kommando "1 #" die Wetterinformation von Leipzig abgerufen werden, bei "2 #" die von München, bei "3 #" von Stuttgart usw...

TclVoiceMail (2011-01-14 19:59)

Dieses Modul beinhaltet ein einfaches Sprachmailboxsystem welches von den lokalen Teilnehmern genutzt werden kann. Es ist im Moment noch nicht möglich, Sprachnachrichten zu den Nutzern anderer Echolink-Knoten zu senden. Um dieses Sprachmailboxsystem nutzen zu können, müssen Sie eine eindeutige Nutzernummer und ein Passwort haben. Nehmen Sie bitte hierzu Kontakt mit Ihrem Sysop auf.

Wenn das Modul aktiviert wurde (Standardeinstellung 3 #) dann werden sie aufgefordert Ihre Nutzernummer und ein Passwort einzugeben. Die Nutzernummer ist immer 3 Ziffern lang, das Passwort zwischen einem und sieben Ziffern.

Geben Sie Ihre Nutzernummer und Ihr Passwort hintereinander ein, z.B. 123456 # wenn Ihre Nutzernummer 123 und Ihr Passwort 456 ist.

Danach befinden Sie sich in einem Kontextmenü, das Sie durch das System führen wird. Eine einzelne Raute (#) beendet die aktuelle Operation.

Um eine Sprachnachricht aufzunehmen nutzen Sie das Kommando 2 # .

Das System fragt Sie jetzt nach dem Empfänger der Nachricht. Sie können beides miteinander verbinden, indem Sie das Kommando und die Nutzernummer hintereinander eingeben. Zum Beispiel: Wenn eine Nachricht für den Nutzer 123 aufgezeichnet werden soll, geben Sie bitte die Folge 2123 # ein.

Der empfohlene Weg um Nutzernummern zu vergeben ist, die Telefon-Methode zu nutzen. Viele Telefone besitzen Buchstaben auf dem Tastenfeld, z.B. 2=ABC, 3=DEF, 4=GHI, 5=JKL, 6=MNO, 7=PQRS, 8=TUV, 9=WXYZ.

Um beispielsweise das Rufzeichen SM0SVX zu kodieren, nutzen Sie bitte die letzten drei Buchstaben und bilden Sie diese auf die Zahlen ab. Dann erhalten Sie die Nutzernummer 789. Im Falle einer Doppelvergabe fügen Sie einfach eine 1 hinzu. Zum Beispiel AFC und ADB lassen sich beide als 232 abbilden. Eine mögliche Abbildung wäre dann AFC=232 und ADB=233. Wenn diese Methode genutzt wird um Nutzernummern zu kodieren, dann kann ein Teilnehmer auf die Nutzernummer eines anderen Teilnehmers schließen.

Es ist für den Sysop möglich für jeden Nutzer eine eMail-Adresse zuzuweisen.

In diesem Fall erhält der Nutzer eine eMail, falls eine neue Sprachnachricht für ihn hinterlegt ist.

Um dieses Modul zu deaktivieren, drücken Sie die Raute-Taste (#) wenn sie sich im Hauptmenü befinden.

Aktivierung

in der /etc/svxlink/svxlink.conf durch Hinzufügen des Eintrages

```
[RepeaterLogic]
```

```
...
```

```
MODULES=...,ModuleTclVoiceMail,...
```

```
...
```

Beispielkonfiguration in der /etc/svxlink/svxlink.d/ModuleTclVoiceMail.conf

```
[ModuleTclVoiceMail]
```

```
NAME=TclVoiceMail
```

```
PLUGIN _NAME=Tcl
```

```
ID=4
```

```
TIMEOUT=60
```

Beschreibung

NAME

Name des Moduls

ID

Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig vergeben werden.

TIMEOUT

Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

Dateien und Verzeichnisse

/etc/svxlink/svxlink.conf

Das systemweite svxlink-Konfigurationsfile

./svxlink/svxlink.conf

Das userspezifisches svxlink-Konfigurationsfile

./svxlink/svxlink.d/ModuleTclVoiceMail.conf

Das userspezifische Konfigurationsfile für das Modul TclVoiceMail, abhängig von der CFG _DIR Konfigurationsvariable.

/etc/svxlink/TclVoiceMail.conf

Das systemweite Konfigurationsfile für das Modul TclVoiceMail.

/var/spool/svxlink/voice _mail

Spool-Verzeichnis für Sprachdateien (der User svxlink muß hier Schreibrechte haben!)

Hilfe (2011-01-14 20:05)

Das Hilfemodul ermöglicht es Hilfe zum System als Ganzes zu bekommen. Im Moment steht die Hilfe allerdings nur in englischer Sprache zur Verfügung.

Senden Sie die ID-Nummer des Systems zu dem Sie eine Hilfe erhalten wollen und bestätigen Sie wie immer mit der Nummerntaste (#)

Aktivierung

in der /etc/svxlink/svxlink.conf durch Hinzufügen des Eintrages

```
[RepeaterLogic]
```

```
...
```

```
MODULES=...,ModuleHelp,...
```

```
...
```

Beispielkonfiguration in der /etc/svxlink/svxlink.d/ModuleHelp.conf

```
[ModuleHelp]
```

```
NAME=Help
```

```
ID=0
```

```
TIMEOUT=60
```

Beschreibung

NAME

Name des Moduls

ID

Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig vergeben werden.

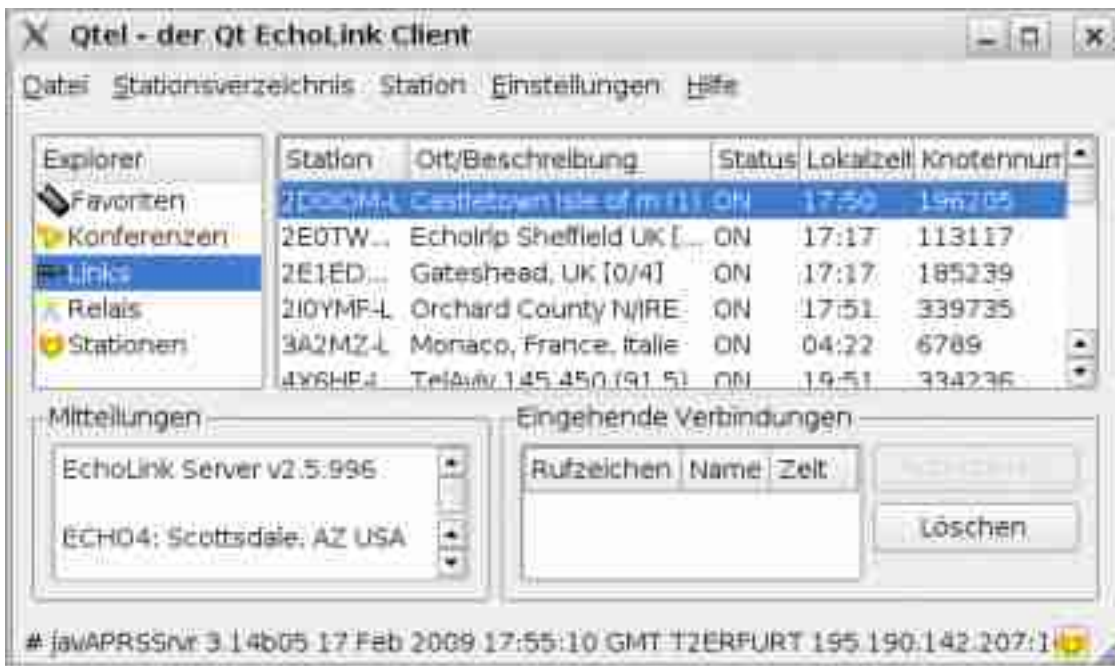
TIMEOUT

Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

QTel-Client (2011-01-14 20:30)

Qtel ist die Abkürzung für "QT Echolink-Client". Es handelt sich hierbei um einen reinen Echolink-Clienten, der keinen SSystem-Modus besitzt. Wenn Sie einen Link aufbauen möchten dann werfen Sie einen Blick auf den SvxLink-server. Qtel ist ein in der Bedienung sehr einfaches Programm und die meisten Dinge sind selbsterklärend. Inzwischen gibt es Sprachunterstützung für Englisch, Schwedisch, Deutsch, Italienisch, Holländisch und Spanisch. Übersetzungen für Türkisch und Ungarisch sind in Vorbereitung. Seit 5/2013 wird auch die Verwendung von EchoLink-Proxyservern unterstützt.

Es gibt drei Fenster: das Hauptfenster, der Konfigurationsdialog und der Kommunikationsdialog.



Das Qtel-Hauptfenster

Das Hauptfenster besteht aus 4 Bereichen. Im oberen linken Bereich kann man auswählen, welche Stations-Kategorie gerade angezeigt werden soll. Alle Stationen einer Kategorie werden dann im rechten Fenster aufgelistet.

Es gibt 4 Kategorien: Konferenzen, Links, Relais und (private) Stationen. Die 5. Kategorie ist ein Bereich in dem alle Stationen selbsttätig eingetragen werden können, die möglicherweise interessant sind, analog zu den Begriffen Favoriten oder Bookmarks bei Webbrowsern. Klicken Sie einfach mit der rechten Maustaste auf eine für Sie interessante Station und fügen Sie diese per SZU den Favoriten hinzu...Ihrer Favoriten-Liste hinzu.

Im linken unteren Bereich befindet sich das Fenster für Mitteilungen, die vom Echolinkserver übermittelt werden. Im rechten unteren Bereich befindet sich eine Liste eingehender Verbindungen. Um eine eingehende Verbindung zu akzeptieren klicken Sie diese Verbindung einfach an und drücken Sie dann auf "Akzeptieren".

Der Kommunikationsdialog ermöglicht die Herstellung einer Verbindung zu einer anderen Echolink-Station. Um diesen Dialog aufzurufen, klicken Sie doppelt auf eine Station oder drücken Sie einfach , wenn diese Station farblich hinterlegt dargestellt ist. Um eine Verbindung herzustellen klicken Sie auf die Schaltfläche "Verbinden...". Wenn eine Audioübertragung von dieser Station erfolgt, so erscheint die RX-Anzeige hellgrün. Um zu Senden drücken Sie die Schaltfläche "PTT". Dabei leuchtet die TX-Anzeige rot. Um dauerhaft auf Sendung zu gehen drücken Sie zuerst auf der Tastatur die CtrlTaste und bei gedrückter Taste die Schaltfläche "PTT". Um die Verbindung zu trennen drücken Sie die Schaltfläche "Trennen...".

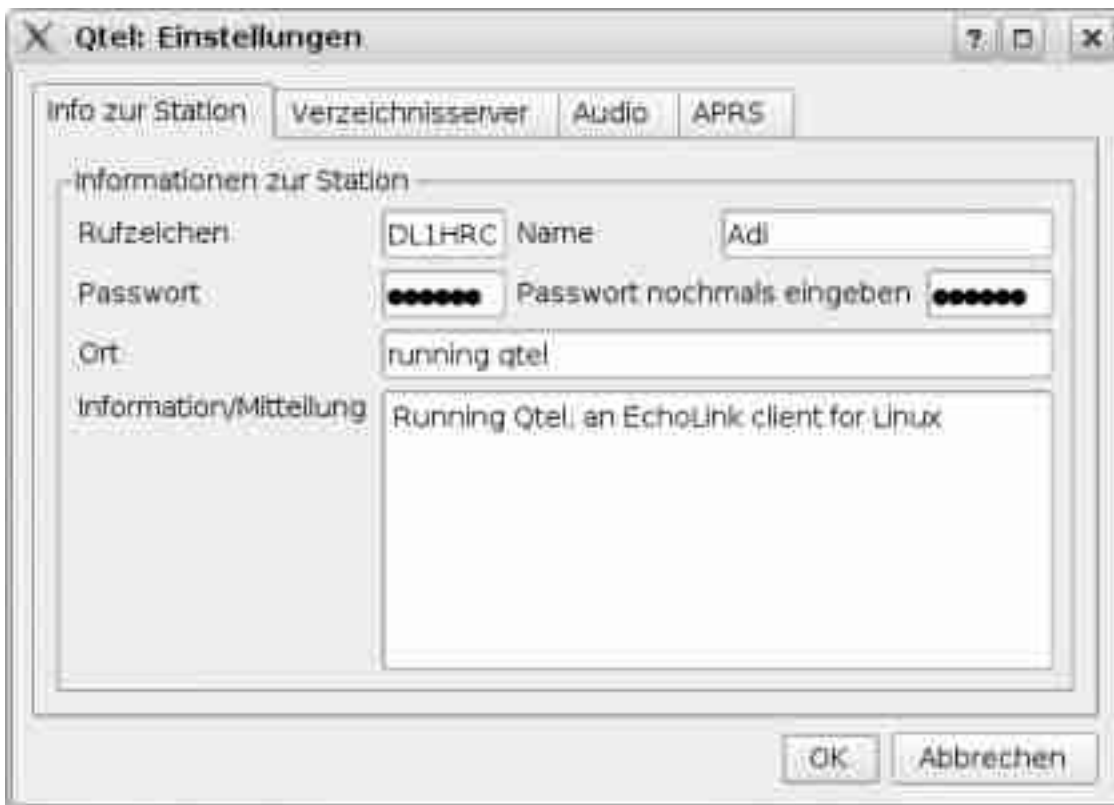


Der Fenster für die QTel-Kommunikation

Im großen weißen Bereich in der Mitte des Kommunikationsdialoges werden Mitteilungen der Gegenstation angezeigt. Auch die Chat-Mitteilungen der Gegenstation werden hier angezeigt. In dem schmalen Bereich unter dieser weißen Fläche können eigene Chat-Mitteilungen an die Gegenstation eingegeben werden.

Der Einstellungsdialog ("Qtel - Einstellungen") wird beim ersten Programmstart automatisch aufgerufen damit die Eingabe der eigenen Daten vereinfacht bzw. ermöglicht wird. Er kann jederzeit über das Menü "Einstellungen->Qtel Einstellungen" wieder aufgerufen werden. Die meisten Feldbezeichnungen sind selbsterklärend. Ort (bzw. Location) ist die im Echolink-Verzeichnis angezeigte Zeichenkette. Die im Bereich Information/Mitteilung eingegebenen Daten werden an die jeweilige Gegenstelle beim Verbindungsaufbau übermittelt.

ACHTUNG: Wenn man den Eintrag bei Ort bzw. Location vergisst erfolgt keine Anmeldung beim Echolink-Verzeichnisserver mit der Fehlermeldung: "The directory server closed the connection before all data was received"



Das Konfigurationsfenster

In der Registerkarte "Verzeichnissever" gibt es einige Einstellungsmöglichkeiten für die Verbindung zum Echolink-Verzeichnisdienst. Wenn Sie die Maus über diesen Bereich bewegen ohne diesen anzuklicken erhalten Sie eine kurze Hilfestellung.

Das ist eigentlich alles was über das Programm Qtel zu sagen ist. Es ist wirklich einfach.

Tips und Tricks (2011-01-14 20:43)

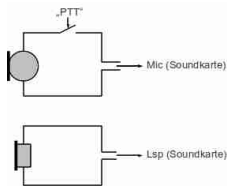
Hier finden Sie ein paar Hinweise, die vielleicht für Ihre eigenen Konfigurationen nützlich sein könnten.

Unter Tcl-Routinen sind Erweiterungen zusammengefasst, die nur durch Änderungen in der Tcl-Umgebung realisierbar sind.

MicSpkr-Logic (2011-01-14 20:47)

Die MicSpkr-Logic ist eigentlich nur eine "verbogene Repeater- oder SimplexLogic, die den Anschluß einer "Hör-Sprech-Garnitur an eine Relaisstelle (auch Simplex-Link) ermöglicht. Damit ergibt sich die Möglichkeit direkt an

der Kommunikation per Funk teilzunehmen. Sie wird dabei gleichberechtigt zu den übrigen Logiken konfiguriert und kann entsprechend genutzt werden.



Anschlußprinzip für Mikrofon und Hörer an einen Server, auf dem SvxLink läuft. In die Mikrofonleitung ist ein PTT-Taster eingefügt, der beim Betätigen die NF auf den Mikrofoneingang der Soundkarte legt und damit die VOX aktiviert

Merkmale

es wird ein Soundkartenkanal benötigt (Lautsprecherausgang und Mikrofoneingang)

Konfiguration als Repeater- oder SimplexLogic

Squelch sollte VOX sein, also sprachgesteuert, so spart man eine serielle Schnittstelle für die PTT

TYPE=Repeater ermöglicht auch einen quasi-Vollduplexbetrieb

Beispielkonfiguration, Ausschnitt aus der /etc/svxlink/svxlink.conf

```
[GLOABAL]
```

```
LOGICS=...,MicSpkrLogic,...
```

```
[MicSpkrLogic]
```

```
TYPE=Simplex
```

```
RX=LocalMic
```

```
TX=LocalSpkr
```

```
DEFAULT_LANG=de_DE
```

```
CALLSIGN=DL1HRC
```

```
EVENT_HANDLER=/usr/share/svxlink/events.tcl
```

```
[LocalMic]
```

```
TYPE=Local
```

```
AUDIO_DEV=oss:/dev/dsp
```

```
AUDIO_CHANNEL=0
```

```
SQL_DET=VOX
```

```
SQL_START_DELAY=0
```

```
SQL_DELAY=0
```

```
SQL_HANGTIME=500
```

```
SQL_TIMEOUT=600
```

```
VOX_FILTER_DEPTH=20
```

```
VOX_THRESH=1000
```

```
[LocalSpkr]
```

```
TYPE=Local
```

```
AUDIO_DEV=oss:/dev/dsp
```

```
AUDIO_CHANNEL=0
```

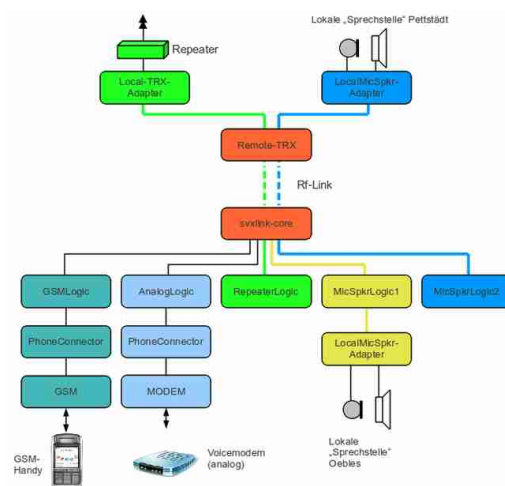
```
PTT_PORT=NONE
```

```
PTT_PIN=!DTRRTS
```

Für Erklärungen zu den Konfigurationsvariablen siehe Repeater- oder Simplexlogic.

Als kleines Beispiel noch eine Übersicht, wie es bei DBOWSF eingerichtet und betrieben wird. Es soll nur eine klei-

ne Anregung darstellen, selber mal damit zu experimentieren. Es macht wirklich Spaß.



SvxLink bei DBOWSF

Propagation-Monitor (2011-01-14 21:07)

Der Ausbreitungsmonitor informiert über besondere Ausbreitungsbedingungen im VHF, UHF und SHF-Bereich. Die Aktivierung erfolgt dabei automatisch immer genau dann wenn besondere Ausbreitungsbedingungen gemeldet wurden (DX-Robot, VHF-DX, ...)

Dieses Modul kann zwar aktiviert werden, besitzt aber keine weitere Funktionalität.

Aktivierung

in der `/etc/svxlink/svxlink.conf` durch Hinzufügen des Eintrages

```
[RepeaterLogic]
```

```
...
```

```
MODULES=...,ModulePropagationMonitor,...
```

```
...
```

Beispielkonfiguration in der `/etc/svxlink/svxlink.d/ModulePropagationMonitor.conf`

```
[ModulePropagationMonitor]
```

```
NAME=PropagationMonitor
```

```
PLUGIN_NAME=Tcl
```

```
ID=5
```

```
TIMEOUT=10
```

```
SPOOL_DIR=/var/spool/svxlink/propagation_monitor
```

Beschreibung

NAME

Der Name dieses Modules, z.B. `NAME=PropagationMonitor`

ID

Die eindeutige Identifikationsnummer, die für jedes SvxLink-Modul eindeutig vergeben werden muss. Die Aktivierung des Modules erfolgt bei `ID=5` mit dem Kommando `5 #`

TIMEOUT

Das Modul beendet sich nach der konfigurierten Zeit (in Sekunden) selbstständig, wenn keine Aktivität erfolgte.

SPOOL_DIR

Verzeichnis in dem die bearbeitete Textdatei, die aus der eMail erstellt wurde, abgelegt wird.

Einrichtung des Mailservers zum Empfang der DX-Meldungen vom DX-Robot:

- Nutzer svxlink auf dem SvxLink-Server erstellen
- Datei `.procmailrc` im Verzeichnis `/home/svxlink` erstellen mit folgenden Einträgen

```
# This is the configuration file for procmail that is used to
# extract received propagation alerts into separate files.
# This is used by the PropagationMonitor SvxLink module.
```

```
LOGFILE=/home/svxlink/procmail.log
VERBOSE=ON
```

```
# Store received propagation alerts here
MAILDIR=/var/spool/svxlink/propagation_monitor
```

```
# Put e-mails from the DXrobot (gooddx.net) in the folder below
:0:
^From.*DXrobot
/var/spool/svxlink/propagation_monitor/dxrobot
```

```
# Put e-mails from VHFDX (vhfdx.net) in the folder below, due to
# the change of the domain name you can comment out
# the next lines
#:0:
#* ^From.*DX.Sherlock
#/var/spool/svxlink/propagation_monitor/vhfdx
```

```
# the new domain name is dxmaps.com
:0:
^From.*warnings@dxmaps.com
/var/spool/svxlink/propagation_monitor/vhfdx
```

```
:0:
^From.*@gooddx.*
/var/spool/svxlink/propagation_monitor/dxrobot
```

```
# Send all other, non matching e-mails, to user root.
# Replace "!root" with "/dev/null" to throw non-matching
# e-mails away instead or comment both lines below out
# to just let non-matching e-mails pass through.
:0:
/dev/null
```

Selektivrufencoder (2011-01-15 20:49)

Mit dem Selektivrufencoder ist es möglich per DTMF aus 17 verschiedenen Selektivrufverfahren auszuwählen und einen Selektivruf auszulösen (z.B. DTMF nach ZVEI-1, DTMF nach EIA, usw.).

Aktivierung

in der `/etc/svxlink/svxlink.conf` durch Hinzufügen des Eintrages

```
[RepeaterLogic]
...
MODULES=...,ModuleSelCallEnc,...
...
```

Beispielkonfiguration in der `/etc/svxlink/svxlink.d/ModuleSelCallEnc.conf`

```
[ModuleSelCallEnc]
NAME=SelCallEnc
PLUGIN_NAME=Tcl
ID=6
TIMEOUT=10
```

Beschreibung

NAME
Name des Moduls

ID
Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig vergeben werden.

TIMEOUT
Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

Dateien und Verzeichnisse

`/etc/svxlink/svxlink.conf`
Das systemweite svxlink-Konfigurationsfile

`./svxlink/svxlink.conf`
Das userspezifisches svxlink-Konfigurationsfile

`./svxlink/svxlink.d/ModuleSelCallEnc.conf`
Das userspezifische Konfigurationsfile für das Modul Selektivrufencoder, abhängig von der CFG `_DIR` Konfigurationsvariable.

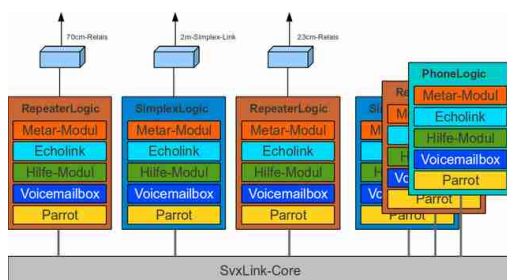
`/etc/svxlink/svxlink.d/ModuleSelCallEnc.conf`
Das systemweite Konfigurationsfile für das Modul Selektivrufencoder, abhängig von der CFG `_DIR` Konfigurationsvariable.

Konzept (2011-01-16 15:53)

SvxLink wurde von Tobias (SMOSVX) bereits um 2002 ins Leben gerufen und wird seitdem ständig weiterentwickelt.

Es besteht aus einem Haupt- bzw. Kernprogramm und sogenannten Logiken, die quasi beliebig kombiniert werden können. Dabei kann jeder Logik verschiedene Module zugeordnet werden. Echolink ist dabei nur eines von weiteren Modulen, somit unterscheidet sich SvxLink vom klassischen Echolink ganz erheblich.

Man ist dabei nicht auf den Betrieb eines einzelnen Relais/Simplex-Link beschränkt, sondern kann theoretisch beliebig viele Relais/Links unabhängig voneinander konfigurieren und betreiben. Das folgende Bild soll das Prinzip etwas verdeutlichen.



Das SvxLink-Konzept

Wenn es die Soundkarte unterstützt, dann lassen sich die beiden Stereokanäle getrennt nutzen, d.h. der linke Kanal steuert das 70cm-Relais, der rechte Kanal das 23cm-Relais und beide selbstverständlich unabhängig voneinander. Grenzen sind hier nur (irgendwann einmal) durch die Rechnerperformance gesetzt.

Prinzipiell gibt es bei dieser Applikation zwei unterschiedliche Ebenen, einmal das Kernsystem, welches die grundlegenden Funktionen realisiert und eine Art Präsentationsebene, die die Funktionalität nach außen hin bereitstellt.

Die Präsentationsebene wird durch ein in SvxLink implementiertes Ereignisbehandlungssystem realisiert. Bei jedem nur möglichen Ereignis wie z.B. Öffnen oder Schließen der Rauschsperr, eingehende Echolinkverbindung, ... wird eine TCL-Funktion aufgerufen, in die der SvxLink-Sysop eigene Routinen einbauen kann. Diese Schnittstelle ist grundsätzlich vergleichbar mit der VisualBasic-Erweiterung bei EcholinkPlus, bietet aber ungleich mehr Funktionalität.

Inhalt (2011-01-16 16:07)

Vorwort

Einführung

Das Modul Hilfe

Das Modul Parrot (Sprachpapagei)

Das Modul Echolink

o Connect by Callsign

Das Modul Mailbox

o Neue Nachrichten lesen

o Neue Nachrichten versenden

o Nachsehen ob neue Nachrichten warten

Das Modul Metar Information

o Die Suchfunktionen

Das Modul Selektiv-Ruf (Encoder)

o Selektivruf Encoder Makro

Selektiv-Ruf (Decoder)

Das Modul Ausbreitungsmonitor

Sprachumschaltung

o Momentan verfügbare Sprachen

Einführung (2011-01-16 16:09)

SvxLink ist ein Sprach-Service-System welches unter Linux läuft. Es ist für die Anwendung im Amateurfunk gedacht und verfügt über spezielle Funktionen. Zu diesen zählt unter anderem die Steuerung von Relaisstationen (Hardware in the loop) und der Betrieb eines Echolink Gateways.

Die Software verfügt über verschiedene Module welche zur Laufzeit geladen werden müssen um deren vollen Funktionsumfang nutzen zu können.

Deshalb soll im folgenden soll die Bedienung der Software durch den Funkamateure erläutert werden und auf die momentan vorhandenen Module der Software eingegangen werden.

Grundlegend ist zu sagen, dass jedes Kommando welches an den Link oder das Relais gesendet wird mit einer Raute (#) abgeschlossen werden muss. Weiterhin ist es möglich den Status der SvxLink Station durch das Senden eines Sterns (*) abzufragen. Beim Senden des Status werden folgende Informationen gesendet:

Name der Station, Uhrzeit, evtl. zu sendender CTCSS Ton, aktives Modul und die Anforderung 0 # zu senden um Hilfe zu erhalten.

Diese hört sich z.B. wie folgt an:

”Delta Bravo Null Whisky Sierra Foxtrott Relais es ist jetzt Null Uhr sieben und fünfzig. Bitte einen CTCSS Ton von

Einhundert Komma Null Hertz mitsenden. Aktives Modul: Echolink. Um Hilfe zu erhalten bitte Null Raute senden.”

Hilfe-Modul (2011-01-16 16:11)

SvxLink verfügt über eine eingebaute Hilfe - Funktion welche der Funkamateure zur Laufzeit aufrufen kann. Dies geschieht gewöhnlich durch das Senden des DTMF Kommandos 0 # zur SvxLink Station. Dadurch bekommt das Modul Hilfe den Fokus (es wird aktiviert). Nach dem das Modul den Fokus hat werden alle, auf diesem Link / Relais vorhandenen Module mit deren Modul ID gesprochen. Durch das Senden der Modul-ID wird die Hilfe des jeweiligen Moduls gesprochen. Nach dem die Hilfe gesprochen wurde befindet man sich wieder im Hauptmenü des Hilfe Moduls. Durch das Senden eines leeren Kommandos (einzelne Raute [#]) wird das Modul beendet. In der Standard Konfiguration sollten folgende Module verfügbar sein.

Modul ID
Modulanme

0
Hilfe

1
Sprachpapagei

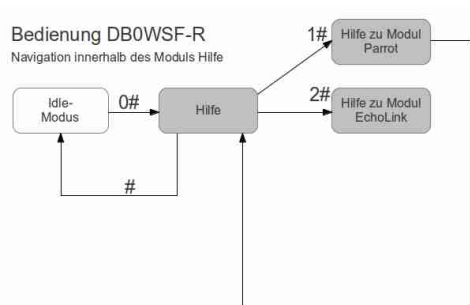
2
Echo Link

3
Metar Information

4
Mailbox

5
Ausbreitungs Monitor

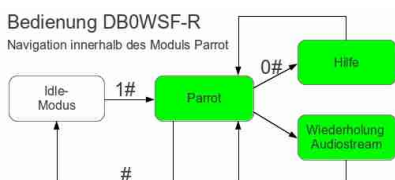
6
Selektiv Ruf (Encoder)



Bedienung Hilfe-Modul

Parrot-Modul (Sprachpapagei) (2011-01-16 16:13)

Dieses Modul speichert die empfangenen Signale und sendet sie erneut aus. Das Modul wird in der Standardkonfiguration mit dem Kommando 1 # aktiviert und kann zur Überprüfung der eigenen Aussendung genutzt werden. Sollte die DTMF Erkennung nicht zu 100 % Funktionieren kann mit Hilfe dieses Moduls festgestellt werden welche DTMF Zeichen vom SvxLink nicht erkannt werden. Das wird gemacht indem das Modul aktiviert wird und Anschließend eine DTMF Folge eingegeben wird. Nachdem diese DTMF Folge mit einer Raute abgeschlossen wurde werden die zuvor eingegebenen Zeichen wieder vorgelesen.



Bedienung Parrot-Modul (konfigurationsabhängig, hier DB0WSF-R)

Echolink-Modul (2011-01-16 16:15)

Mit diesem Modul ist es möglich Verbindungen zu anderen Echolink-Stationen herzustellen. Die Aktivierung erfolgt gewöhnlich durch das Senden des DTMF Kommandos 2 #. Nach dem das Modul den Fokus hat stehen folgende Kommandos zur Verfügung.

Kommando
Funktion

0
Abspielen der Hilfe

1
Abfrage bestehender Verbindungen

2
Anzeige der Lokalen Node ID

31
Verbindungsaufbau zu einem zufälligen Link oder Relais

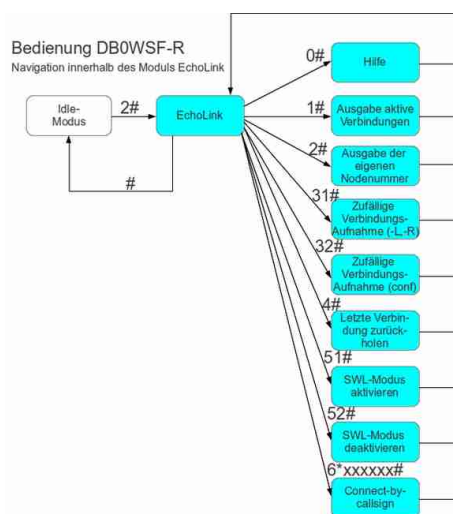
32
Verbindungsaufbau zu einem zufälligen Konferenzserver

4
Letzte Verbindung zurückholen

50
SWL Modus Deaktivieren

51
SWL Modus Aktivieren

6*
Connect by Callsign



Bedienung EchoLink-Modul (konfigurationsabhängig, hier DBOWSF-R)

Connect-by-callsign (2011-01-16 16:19)

Die Connect by Callsign Funktion ist dazu gedacht eine Verbindung zu Stationen herzustellen von denen die Node ID unbekannt ist. Dabei ist wie folgt vorzugehen. Die Funktion ist mit 6* einzuleiten danach ist die Buchstaben-Zahlen-Kombination des Rufzeichens einzugeben. Jedes Zeichen ist nur einmal zu senden, ähnlich der intelligenten SMS-Texteingabe bei Mobiltelefonen. Die Kombination kann aus untenstehender Tabelle abgelesen werden.

Taste 1 2 3

Bedeutung 1 A B C 2 D E F 3

Taste 4 5 6

Bedeutung G H I 4 J K L 5 M N O 6

Taste 7 8 9

Bedeutung P Q R S 7 T U V 8 W X Y Z 9

Taste * 0 #

Bedeutung KEINE 0 KEINE

Wird nach der Eingabe des Rufzeichens oder eines Teils des Rufzeichens ein * gesendet wird dies als Wildcard ausgewertet. Diese funktioniert jedoch nur wenn der * das letzte Zeichen im Kommando ist. Hier einige Beispiele:

6*331262 gibt unter anderem DF1AMB zurück. Dabei werden DF1AMB-L und DF1AMB-R jedoch vollständig ignoriert. Soll jedoch auch nach den anderen beiden Varianten von DF1AMB gesucht werden sollte der Suchbefehl 6*331262* lauten.

Hier noch weitere Beispiele:
KommandoStation

6*760789* SM0SVX*

6*36095* DM0ZL*

6*320973* DB0WSF*

Wichtig hierbei ist jedoch, dass es vorkommen kann das die Suche fehlschlägt. Dies ist auf zwei mögliche Ursachen zurückzuführen. Eine Ursache ist ein Fehler im eingegebenen Rufzeichen. Die andere Ursache ist, dass

die gesuchte Station nicht verfügbar ist. Dies ist vergleichbar mit einem abgeschalteten Mobiltelefon.
next up previous contents

Mailbox-Modul (2011-01-16 16:22)

Im Leerlauf kann dieses Modul durch das Senden des Kommandos 4 # aktiviert werden. Nach dem Aktivieren des Moduls wird man aufgefordert seine Nutzernummer gefolgt vom Passwort einzugeben. Das Login Kommando sieht z.B wie folgt aus:

1 2 3 1 2 3 4 5 6 7

Die ersten drei Ziffern entsprechen dabei immer der Nutzernummer und die letzten maximal 7 Ziffern entsprechen dem Passwort. Nach dem erfolgreichen Login kann man nun wählen ob man seine neuen Nachrichten hören möchte oder ob man eine neue Nachricht versenden möchte.

Mailbox: Nachrichten auslesen (2011-01-16 16:24)

Zum Lesen neuer Nachrichten ist nach erfolgreichem Login das Kommando 1 # zu senden. Nachdem die Nachricht abgespielt wurde hat man die Möglichkeit diese Nachricht zu löschen oder dem Absender eine Antwort zu schicken und die Nachricht danach zu löschen oder Die Nachricht noch einmal abzuspielen.

KommandoAktion

1 Nachricht löschen

2 Nachricht beantworten und anschließend löschen

3 Nachricht erneut Abspielen

Mailbox: Nachricht versenden (2011-01-16 16:25)

Mit Hilfe des Kommandos 2 # kann man, nach erfolgreichem Login in die Mailbox, eine Nachricht verschicken. Dazu muß man die Nutzernummer des Empfängers kennen. Man wird zu Beginn des Versendens aufgefordert die Nutzernummer des Empfängers einzugeben.

Sollte man jedoch eine Nachricht beantworten wollen ist es nicht notwendig die Nutzernummer des Empfängers einzugeben.

Nach der Eingabe des Kommandos wird man aufgefordert einen kurzen Betreff anzugeben. Dieser darf in keinem Fall länger als 10 Sekunden sein.

Wurde der Betreff gesprochen, wird man aufgefordert die Nachricht zu sprechen. Auch für die Nachricht besteht eine Zeitbegrenzung. Diese beträgt im Normalfall 2 Minuten. Jedoch kann diese Einstellung auch vom Betreiber des SvxLinks-Nodes je nach Bedarf geändert werden.

Mailbox: Nachschauen, ob Nachrichten bereitliegen (2011-01-16 16:26)

Durch eine Zusatzfunktion ist es möglich auch ohne Login in die Mailbox die Anzahl der wartenden Nachrichten für einen Nutzer abzufragen. Dabei ist es ebenfalls egal ob ein anderes Modul aktiv ist oder ob SvxLink sich im Leerlauf befindet. Der zu verwendende Befehl hierfür sieht wie folgt aus:

4 123 #

Der erste Teil ist eine „Vorwahl“ um SvxLink mitzuteilen, dass der folgende Befehl von dem Modul nummer 4, also in diesem Fall die Mailbox, verarbeitet werden soll. Der zweite fett geschriebene Teil ist die Nutzernummer. Die Antwort von SvxLink sieht dann wie folgt aus:

Delta Foxtrott Eins Alpha Mike Bravo es sind Sechs neue Nachrichten vorhanden.

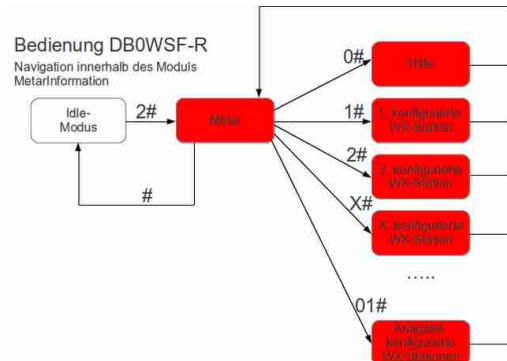
METAR-Modul (2011-01-16 16:27)

Mit Metar Information ist es möglich den aktuellen Wetterbericht von Wetterstationen abzurufen die sich im Allgemeinen an oder in der Nähe von internationalen Verkehrsflughäfen befinden. Dazu muss das Modul durch das Senden des Kommandos 3 # aktiviert werden. Nach dem Aktivieren kann der Wetterbericht von vordefinierten Stationen durch das Senden der Stations ID abgerufen werden. Es ist ebenfalls möglich sich die Liste der vordefinierten Stationen ansagen zu lassen. Dies geschieht durch das Senden des Kommandos 01 #. Es ist ebenfalls möglich die Metarinfos von nicht vordefinierten Flughäfen abzurufen. Hierfür muss die ICAO Kennung des Flughafens bekannt sein. Die Verwendung der Suchfunktion wird später beschrieben.

Die Ausgabe der Wettermeldungen erfolgt im METAR-Stil bzw. ähnlich der ATIS an Verkehrsflughäfen.

Hier ein Beispiel:

Leipzig-Halle Flughafen, Met-Report Zeit 1 0 5 0, Wind 2 4 0 Grad mit 1 2 Knoten, Sichtweite mehr als 1 0 km, Temperatur 2 3 Grad, Taupunkt 1 0 Grad, Luftdruck 1 0 1 3 Hektopascal, Trend keine signifikanten Änderungen”



Bedienung METAR-Modul (konfigurationsabhängig, hier DBOWSF-R)

HINWEIS: Die Daten werden erst unmittelbar nach der Eingabe des Kommandos vom NOAA-Server abgerufen. In Abhängigkeit der Netz- und Serverbelastung kann es zu Verzögerungen oder Timeouts kommen, so dass die Anzeige verzögert oder gar nicht erfolgt. In diesem Fall einfach später nochmals versuchen.

METAR: Suchfunktion 1 (2011-01-16 16:28)

Bei dieser Suchfunktion wird die ICAO Kennung im SMS-Stil eingegeben und die einzelnen Zeichen werden mit * voneinander getrennt. Die Ziffer welche dem jeweiligen Zeichen zugeordnet ist muss dabei so oft gesendet werden bis das Zeichen erreicht ist (SMS-Eingabe bei sehr alten Mobiltelefonen ohne automatische Texterkennung).

Taste 1 2 3

Bedeutung KEINE A B C D E F

Taste 4 5 6

Bedeutung G H I J K L M N O

Taste 7 8 9

Bedeutung P Q R S T U V W X Y Z

Taste * 0 #

Bedeutung Trenner KEINE KEINE

Die Methode soll hier an zwei kurzen Beispielen erklärt werden. Der Flughafen Memmingen hat die ICAO Kennung E D J A. Aus der oben stehenden Tabelle kann nun die Kombination entnommen werden. Diese lautet 33 *

3 * 6 * 2. Für den Flughafen Zürich (L S Z H) lautet der Suchbefehl 555 * 7777 * 9999 * 44.

METAR: Suchfunktion 2 (2011-01-16 16:35)

Diese Suchfunktion funktioniert ähnlich wie die erste jedoch ist hier kein Trennzeichen notwendig um die Zeichen voneinander zu trennen. Bei dieser Suchmethode wird zuerst die Taste angegeben auf welcher das Zeichen der zu finden ist, danach wird die Stelle des Zeichens angegeben. Die folgende Tabelle soll das veranschaulichen.

Taste 1 2 3

Position 1 2 3 1 2 3 1 2 3

Zeichen A B C D E F

Taste 4 5 6

Position 1 2 3 1 2 3 1 2 3

Zeichen G H I J K L M N O

Taste 7 8 9

Position 1 2 3 4 1 2 3 1 2 3 4

Zeichen P Q R S T U V W X Y Z

Taste * 0 #

Position 1 2 3 1 2 3 1 2 3

Zeichen

Hier noch ein Beispiel zur Verwendung dieser Suchmethode. Als Beispiel dient wieder der Flughafen Memmingen (E D J A). Das einzugebende Kommando lautet jetzt also 32 31 51 21.

Selektivruf-Modul (2011-01-16 16:39)

Mit diesem Modul ist es möglich, 18 verschiedene Arten von Selektivrufen zu senden. Dies geschieht nach dem Aktivieren des Moduls durch die Eingabe der Selektivrufart und der zu rufenden Nummer.

Somit ist das Kommando zweigeteilt, ähnlich wie das Login Kommando der Mailbox. Hier repräsentieren die ersten zwei Zeichen den Selektivruftyp und die restlichen maximal 18 Zeichen werden als Selektivruf in der gewählten Variante ausgestrahlt. In der folgenden Tabelle sind die Selektivruftypen und deren Selektivruf ID zu finden.

Selektivruf ID Selektivruf Typ Tonlänge

01 ZVEI1 70ms

02 ZVEI2 70ms

03 ZVEI3 70ms

04 PZVEI 70ms

05 DZVEI 70ms

06 EEA 40ms

07 CCIR1 100ms

08 CCIR2 70ms

09 VDEW 100ms

10 CCITT 100ms

11 NATEL 70ms

12 EIA 33ms

13 EURO 100ms

14 MODAT 40ms

15 PDZVEI 70ms

16 PCCIR 100ms

17 AUTO-A 70ms

18 QC2 1 / 3s

In der Standard Konfiguration hat dieses Modul die Nummer 6.

Selektivrufe können auch abgesetzt werden wenn ein anderes Modul aktiv ist. Hierfür ist es notwendig SvxLink mitzuteilen, dass das folgende Kommando nicht für das aktive Modul bestimmt ist sondern für ein anderes Modul. Dies geschieht mit Hilfe einer "Vorwahl" und sieht wie folgt aus:

6 01 03423 #

Die "Vorwahl" ist in diesem Beispiel fett dargestellt, das zu verwendende Verfahren Kursiv und die zu rufende Nummer ist unterstrichen.

Es ist auch möglich mehrere Stationen der gleichen Nutzergruppe zu rufen. Dies geschieht mit Hilfe des Sterns. Um beispielsweise alle Nutzer der Gruppe 0342* zu rufen ist folgender Befehl zu verwenden:

(*6) 01 0342*

Die in Klammern gestellte Vorwahl ist jedoch nur dann notwendig, wenn ein anderes Modul aktiv ist.

Selektivruf-Dekoder (2011-01-16 16:46)

Dies ist kein Modul im eigentlichen Sinne sondern nur eine Möglichkeit, SvxLink parallel zu bekannten DTMF-Funktionen per Mehrtonrufverfahren zu steuern.

Das verwendete Verfahren ist abhängig von der Konfiguration des SvxLinks. Somit kann man auch ohne DTMF-Encoder bestimmte Aktionen auslösen, dies ist interessant für Funkamateure mit kommerziellen Funkgeräten ohne eingebauten DTMF-Encoder. Da dieses Mehrtonrufverfahren auch zum selektiven Anruf einzelner Amateurfunkstationen verwendet werden kann, kann natürlich nur ein bestimmter Selektivrufbereich genutzt werden. Die empfangenen Befehle werden dabei in Makroaufrufe von SvxLink umgewandelt.

Die Konfiguration erfolgt in der /etc/svxlink/svxlink.conf. Es müssen die Variablen SEL5 _MACRO _RANGE, SEL5 _DET, SEL5 _DEC _TYPE sowie SEL5 _TYPE konfiguriert werden.

Hier nun ein Beispiel:

Selektivrufnr (z.B. ZVEI-1)SvxLink-Macro (DTMF)Aktion

```
03451 D03451 # MetarInfo von Leipzig
03452 D03452 # MetarInfo von Stuttgart
03453 D03453 # MetarInfo von München
03454 D03454 # MetarInfo von Köln/Bonn
```

Propagationmonitor-Modul (2011-01-16 16:47)

Dieses Modul kann zwar aktiviert werden aber es führt keine Funktionen aus. Es läuft im Normalfall im Hintergrund und empfängt DX Meldungen aus dem Internet per Email. Sobald eine solche Meldung empfangen wurde, wird diese von SvxLink angesagt.

Beispielansagen für den Ausbreitungsmonitor:

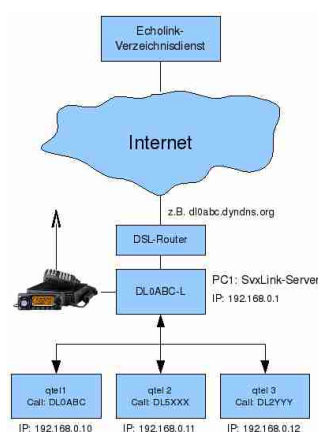
SSporadic-E Öffnung auf 6m, die maximal nutzbare Frequenz ist 68MHz über JN60”
Transäquatorialöffnung auf 6 m”

Mehrere Qtel-Clients mit SvxLink (lokal) (2011-01-16 17:08)

Mehrere Qtel-Clients an einem DSL-Anschluß betreiben

Mit SvxLink ist es relativ einfach möglich, mehrere PCs mit einem Echolink-Client (Qtel) an einem DSL-Anschluß zu betreiben um z.B. von verschiedenen Räumen (Shack, Küche, Wohnzimmer,...) aus gleichzeitig QRV zu sein. Ein kleiner Wermutstropfen soll gleich am Anfang nicht verschwiegen werden. Es geht nur eine gleichzeitige Echolink-Verbindung mit Stationen von/nach außen. Das Führen von mehreren unabhängigen Gesprächen ist nicht möglich was allerdings prinzipbedingt ist.

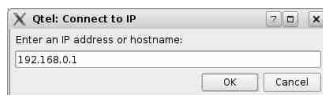
Die folgende Grafik zeigt eine mögliche Konfiguration.



SvxLink: Mehrere QTel-Clients an einem DSL-Anschluß

Auf PC1 ist ein SvxLink-Server installiert und am Echolink-Netzwerk unter z.B. DL1ABC-L angemeldet. Auf den anderen PCs läuft nur der Client Qtel. Die einzelnen Qtel-Clients können nun so konfiguriert werden, als würden sie sich eigenständig im Echolink-Netzwerk anmelden was sie auch tun. Jede der Stationen (DLOABX-L, DLOABC, DL5XXX und DL2YYY) ist nun für alle Echolink-Teilnehmer weltweit zu sehen.

Die Clients verbinden sich nun über das private Netzwerk mit dem eigenen SvxLink-Server DLOABC-L.



Wird nun einer der genannten Stationen von einem Echolink-Teilnehmer außerhalb des eigenen Netzwerkes connected, so nimmt grundsätzlich DLOABC-L die Verbindung entgegen. Da aber auch die anderen drei Stationen (DLOABC, DL5XXX sowie DL2YYY) mit DLOABC-L connected sind, werden alle eingehenden Audiodaten an alle weitergeleitet. Wenn ein Teilnehmer des eigenen Home-Netzwerkes spricht, z.B. DL2YYY, so hören alle anderen mit (DLOABC-L, DLOABC, DL5XXX und die aus dem Internet verbundene Station). Im oben gezeigten Fall wird das Gespräch also auch per Funk ausgesendet und Funkteilnehmer können sich in das laufende Gespräch hineinmelden.

Auch ohne Connect von außen ist eine Verbindung zwischen den Clients möglich, aber Achtung wenn man ein

eigenes Relais oder einen eigenen Link betreibt. Das Gespräch wird automatisch ausgesendet!

Auch Qtel-Stationen außerhalb des eigenen Netzwerkes können sich direkt mit dem eigenen SvxLink-Server verbinden ohne dabei die Verbindung über den Echolink-Verzeichnisdienst herstellen zu müssen:



Hierbei muß der eigene SvxLink-Server allerdings bei einem DDNS-Anbieter wie z.B. dyndns.org registriert sein, da es sonst sehr schwierig wird bei ständig wechselnden IP-Adressen von außen den SvxLink-Server zu erreichen. Die externe Station muß aber zwingend im Echolink-Netzwerk angemeldet und authentifiziert sein! SvxLink prüft beim Connect ab, ob es sich um eine im Echolink-Netzwerk registrierte und authentifizierte Station handelt.

Wer das einmal ausprobieren möchte, dem sei Puppylink empfohlen, ein Linux-Livesystem, welches von CD gebootet wird und Qtel bereits beinhaltet.

Dieser Anordnung sind nur durch die Bandbreite des DSL-Anschlusses bzw. des eigenen Netzwerkes Grenzen gesetzt.

Damit SvxLink die Anfragen der eigenen Clients akzeptiert, muß dies in der Datei /etc/svxlink.d/ModuleEchoLink.conf explizit freigeschaltet werden:

```
[ModuleEchoLink]
...
ALLOW_IP=192.168.0.0/24
...
```

wobei natürlich der IP-Adressraum des eigenen privaten Netzwerksegmentes eingetragen werden muß.

Probleme und Hilfe (2011-01-16 17:15)

Es gibt einen Weg um mit der SvxLink-Gemeinde Kontakt aufzunehmen: durch eine Mailingliste. Die Adresse dieser Mailingliste ist svxlink-devel@lists.sourceforge.net.

Was auch immer ihr Anliegen ist, bitte nutzen Sie hierfür die Mailingliste. Nutzen Sie NICHT Echolink und senden Sie auch keine direkte Mail an den Verfasser. Diese Mailingliste ist nicht stark frequentiert, so dass Sie keine Bedenken haben müssen sich dort zu registrieren.

Es gibt eine Reihe von Gründen weshalb nur die Mailingliste zur Unterstützung bei Fragen und Problemen genutzt werden sollten:

Es gibt mehr Leute, die helfen könnten. Ich selbst (Tobias, SMOSVX, d.Red.) nutze nur Fedora Linux aber es gibt hier Leute die haben SvxLink unter verschiedenen Distributionen laufen

Dokumentation. Alle eMails, die an die Liste gesendet werden, werden im Mailinglisten-Archiv gespeichert. Eine gute Empfehlung zur Suche nach Lösungen ist, zunächst im Mailinglistenarchiv nachzuschauen.

Andere Leute die dieser Liste beitreten, können sich an der Diskussion beteiligen

Es gibt noch eine andere Mailingliste svxlink-announce@lists.sourceforge.net welche noch weniger frequentiert ist. Bei jeder neuen Version wird eine einzelne eMail versendet. Das ist wirklich nicht viel. Wenn Sie bereits der svxlink-devel-Liste beigetreten sind, dann gibt es keine Notwendigkeit auch der svxlink-announce-liste beizutreten.

Um den Listen beizutreten klicken Sie bitte [hier](#). Beachten Sie bitte, dass in den Mailinglisten Englisch gesprochen wird!

Fehler melden (2011-01-16 17:16)

Fehler sollten dem SourceForge Fehler-Tracking-System gemeldet werden. Es kann manchmal schwierig sein, einen gute Fehlermeldung zu schreiben aber ich versichere Ihnen das es noch schwieriger ist eine schlechte Fehlerbeschreibung zu verstehen. Wenn Sie also einen Fehler finden, nehmen Sie sich bitte die Zeit die folgenden Schritte abzuarbeiten:

1) Schauen Sie im Fehler-Tracker nach um zu sehen, ob dieser Fehler schon einmal gemeldet wurde

2) Lesen Sie sich die Dokumentation noch einmal durch um auch ganz sicher zu sein, dass Sie nicht eventuell etwas mißverstanden haben. Das schließt die Installationsanleitung, die SvxLink Server-Dokumentation und die Qtel-Dokumentation, abhängig davon in welcher Applikation Sie den Fehler gefunden haben, mit ein.

3) Versuchen Sie wenn möglich den Fehler zu reproduzieren. Die Art und Weise wie ein Fehler zu reproduzieren ist macht es dem Entwickler einfacher den Fehler zu finden und zu beseitigen.

Wenn es nicht möglich ist den Fehler zu reproduzieren, versuchen Sie sich daran zu erinnern, welche genauen Handlungen dem Fehler vorausgingen. Fügen Sie der Meldung eine genaue Schritt-für-Schritt-Beschreibung hinzu damit man den Fehler eventuell reproduzieren kann.

4) Schreiben Sie einen Fehlerbericht im Fehler-Tracker. Der Fehlerbericht sollte folgendes beinhalten:

Eine detaillierte Beschreibung des Fehlers. Schreiben sie nicht einfach "Das funktioniert nicht". Erläutern sie auf unter welchen Umständen es nicht funktioniert. Es kann sehr viele Möglichkeiten geben, weshalb eine bestimmte Funktion nicht funktioniert.

Wenn möglich, schreiben Sie wie der Fehler reproduziert werden kann.

Wenn Sie diese einfachen Regeln befolgen, können Fehler viel schneller gefunden und behoben werden.

Hilfe (2011-01-16 17:33)

Hier finden Sie weitere Informationen falls Sie bei der Installation von Svxlk auf Probleme stossen oder auch sonst nicht weiterkommen.

Bitte vergessen Sie dabei nicht, Svxlk ist eine Amateurfunksoftware, die unter der GNU-Lizenz veröffentlicht wird: offen, frei und nichtkommerziell. Alle Entwickler öpfere für dieses Projekt ihre Freizeit. Sie haben hier keine Firma, von der Sie grenzen- und kostenlosen Support einfordern können.

PropagationMonitor (2011-01-16 18:06)

Eine kurze Anleitung in englischer Sprache ist hier zu finden:
http://sourceforge.net/mailarchive/message.php?msg_id=23367461

Die erforderlichen Wav's sind je nach Sprachpaket bereits vorhanden und befinden sich im Verzeichnis /usr/share/svxlk/sounds/ \$LOCALE/PropagationMonitor (\$LOCALE=de _DE für eine deutsche Sprachumgebung). Zur Ausgabe wird das Subject der eMail analysiert und aufbereitet, Bsp:

Date: Sun, 26 Oct 2014 17:52:30 +0000
Subject: TEP opening on 6m

Die Ansage dieser Information erfolgt durch Aneinanderreihung der Ausgaben:
Transäquatorialöffnung"
äuf"
"6"
Meter"

1) Installation von postfix und procmail

Wenn diese beiden Programme noch nicht auf Ihrem System installiert sein sollten, dann tun Sie das bitte. Erstellen Sie den Nutzer *svxlk* sowie das Home-Verzeichnis */home/svxlk*

Erstellen Sie eine Datei *.procmailrc* und speichern Sie diese bitte im Home-Verzeichnis des Users *svxlk* ab (*/home/svxlk*). Hier ein Beispiel:

```
#####  
#####  
LOGFILE=/home/svxlk/procmail.log  
VERBOSE=ON # debug, should be switched to OFF later  
  
# Store received propagation alerts here  
MAILDIR=/var/spool/svxlk/propagation_monitor  
  
# Put e-mails from the DXrobot (gooddx.net) in the folder below
```

```
:0:
^From.*DXrobot
dxrobot
```

Put e-mails from VHFDX (vhfdx.net) in the folder below

```
:0:
^From.*DX.Sherlock
vhfdx
```

```
# Send all other, non matching e-mails, to /dev/null.
# Replace "!root"with "/dev/nullto throw non-matching
# e-mails away instead or comment both lines below out
# to just let non-matching e-mails pass through.
```

```
:0:
/dev/null
```

```
#
# end of .procmailrc
#####
```

2) Erstellen Sie die Mail-Spool-Verzeichnisse

```
/var/spool/svlink/propagation_monitor
/var/spool/svlink/propagation_monitor/vhfdx
/var/spool/svlink/propagation_monitor/dxrobot
```

Der Eigentümer muß svlink sein.
chown -R svlink:users /var/spool/svlink/propagation_monitor

3) Auf manchen Linux-Systemen muss postfix für die Annahme von eMails außerhalb des eigenen Netzwerks eingerichtet werden.
Editieren Sie hierfür das Konfigurationsfile *main.cf* AM ENDE der Datei(!):

```
/etc/postfix/main.cf

inet_interfaces = all
```

4) Prüfen Sie ob postfix beim Sytemstart mit gestartet wird, sehen Sie hierzu in der Datei */etc/postfix/main.cf* nach

```
mailbox_command = /usr/bin/procmail
```

5) Auf Systemen mit geringen Ressourcen sollte das "size limit"herabgesetzt werden um das System vor "vollaufendenFehlermeldungen zu schützen
queue_minfree = 0
mailbox_size_limit = 512000
message_size_limit = 128000

Auf unserem EPIA-Board mit 8GByte SD-Karte und 512MByte RAM + Ramdisk verweigert postfix ohne o.g. Änderungen die Annahme von eingehenden eMails. Wenn Sie genügend Ressourcen haben, lassen Sie alles wie es ist.

6) Wenn Sie SvxLink an einem DSL-Anschluß mit täglich wechselnder IP-Adresse betreiben, ist die Anmeldung bei z.B. dyndns.org erforderlich. Zum Beispiel: *DEIN_RUFZEICHEN.homelinux.org*

Fügen Sie den Eintrag *DEIN_RUFZEICHEN.homelinux.org* in der Datei *main.cf* hinzu:

```
mydomain = DEIN_RUFZEICHEN.homelinux.org
```

Wenn Sie ein privates Netzwerk betreiben (ohne Hostnamen/Domains *.homelinux.org), dann muß postfix noch mitgeteilt werden, dass er für den Transport eingehender eMails an *svxlink@DEIN_RUFZEICHEN.homelinux.org* zuständig ist.

7) Richten Sie Ihr System so ein, dass postfix in den Runleveln 3 (und 5) gestartet wird

8) Richten Sie auf Ihrem DSL-Router ein Portforwarding für Port 25/tcp ein. Das Ziel muss auf die IP-Adresse des Rechners verweisen, auf dem Sie svxlink betreiben. Sie können die ordnungsgemäße Funktion testen, indem Sie telnet *YOUR_CALLSIGN.homelinux.org 25* von einem Rechner außerhalb Ihres eigenen Netzwerkes aus aufrufen.

9) Registrieren Sie sich in der vhfdx.net-Mailingliste mit Ihrer neuen eMail-Adresse *svxlink@DEIN_RUFZEICHEN.homelinux.org*

Nach einigen Minuten sollte die erste eMail eingehen, prüfen Sie ob in der Datei */home/svxlink/procmail.log* ein entsprechender Eintrag erscheint.

Prüfen Sie weiterhin, dass diese eMail im richtigen Verzeichnis abgespeichert wurde:
/var/spool/svxlink/propagation_monitor/vhfdx

10) Installieren Sie alle benötigten Sounddateien für den Propagationmonitor.
/usr/share/svxlink/sounds/de_DE/PropagationMonitor
oder erstellen Sie eigene Dateien (z.B. mit Mary)

11) Fügen Sie in der Datei */etc/svxlink/svxlink.conf* den Eintrag zur Aktivierung des Propagation hinzu

```
[RepeaterLogic]  
MODULES=...,ModulePropagationMonitor,...
```

```
oder in einer SimplexLogic  
[SimplexLogic]  
MODULES=...,ModulePropagationMonitor,...
```

und starten Sie SvxLink neu.

12) Der Propagationmonitor prüft jede Minute, ob eine neue eMail eingegangen ist. Diese eMail wird analysiert, aufbereitet, ausgegeben und danach in einen Archiv-Ordner verschoben. Prüfen Sie von Zeit zu Zeit, ob das Löschen alter eMails notwendig ist (Platzmangel)

LocationInfo (APRS) (2011-01-16 18:36)

SvxLink unterstützt sowohl das Senden von APRS-Baken zum APRS-Netzwerk als auch die Echolink-eigenen Statusmeldungen. Im APRS-Netzwerk wird dabei der Connect/Disconnect einer Station als APRS-Statusmessage angezeigt. Im APRS-Netzwerk wird dabei der SvxLink-Node als Objekt von...in der Form *EL-DL1ABC* bzw. *ER-DBOABC* angezeigt und nicht als *EL-12345*. Es gibt verschiedene Gründe, weshalb dies in der vorliegenden Form implementiert wurde.

Die Konfiguration erfolgt in der Datei */etc/svxlink/svxlink.conf* durch das Eintragen der Variable `LOCATION_INFO`:

```
[GLOBAL]
...
LOCATION_INFO=LocationInfo
...

[LocationInfo]
APRS_SERVER_LIST=rotate.aprs.net:14580
STATUS_SERVER_LIST=aprs.echolink.org:5199
LON_POSITION=11.12.13E
LAT_POSITION=51.52.53N
FREQUENCY=438.725
CALLSIGN=ER-DBOABC
TX_POWER=5
ANTENNA_GAIN=6
ANTENNA_HEIGHT=12m
ANTENNA_DIR=-1
PATH=WIDE1-1
BEACON_INTERVAL=10
TONE=0
COMMENT=SvxLink by SM0SVX (svxlink.sourceforge.net)
```

Beschreibung

`LOCATION_INFO`

Zeigt auf den Abschnitt innerhalb der */etc/svxlink/svxlink.conf*, die die Definitionen für die LocationInfo (APRS) enthält.

`APRS_SERVER_LIST`

Server des APRS-Netzwerkes, der die Positionsmeldungen/Statusinformationen entgegennimmt (via TCP). z.B. *APRS_SERVER_LIST=rotate.aprs.net:14580*

`STATUS_SERVER_LIST`

Echolink-Server, der die aktuellen Positions- und Statusmeldungen entgegennimmt (via UDP), z.B. *STATUS_SERVER_LIST=aprs.echolink.org:5199*

`LON_POSITION`

Geographischer Längengrad der eigenen Position, z.B. *LON_POSITION=11.12.13E*

`LAT_POSITION`

Geographischer Breitengrad der eigenen Position, z.B. *LAT_POSITION=51.52.53N*

`FREQUENCY`

Sendefrequenz des SvxLink-Nodes, z.B. *FREQUENCY=438.725*

CALLSIGN

Das Rufzeichen, unter dem SvxLink läuft, wobei ein ËR-für Relaisstationen und ËL-für Simplex-Links vorangestellt werden muß. z.B. *CALLSIGN=ER-DBOABC*

TX_POWER

Die Sendeleistung in Watt, z.B. *TX_POWER=5*

ANTENNA_GAIN

Der Antennengewinn in dB, z.B. *ANTENNA_GAIN=6*

ANTENNA_HEIGHT

Die Antennenhöhe über Grund, NICHT MSL(!), in Meter (*ANTENNA_HEIGHT=12m*) oder Fuss (*ANTENNA_HEIGHT=22*).

ANTENNA_DIR

Die Hauptstrahlungsrichtung der Sendeantenne in Grad, beim Rundstrahler 1 eintragen, z.B. *ANTENNA_DIR=-1*

PATH

Der APRS-Pfad, nur wichtig, wenn die Bake von einem benachbarten APRS-Digi wieder ausgesendet wird, z.B. *PATH=WIDE1-1*

BEACON_INTERVAL

Das Sende-Intervall in Minuten, z.B. *BEACON_INTERVAL=10*. Kleinere Werte als 10 werden automatisch auf 10 Minuten gesetzt.

tone

Gibt an, ob für den Funkverkehr über das Relais ein Rufton oder CTCSS-Ton benötigt wird, z.B.

tone=0 -> kein Rufton/CTCSS notwendig

tone=1750 -> Rufton 1750 Hz zum Öffnen des Relais

tone=250 -> CTCSS-Ton von 250Hz notwendig

COMMENT

Kommentar, der bei jeder Bake mit gesendet wird, z.B.

COMMENT=[Svx] Relais Musterstadt

RepeaterLogic (2011-01-17 11:11)

Um einen Repeater einzurichten, müssen Sie in der Datei */etc/svxlink/svxlink.conf* den Abschnitt *[RepeaterLogic]* bearbeiten und diesen Abschnitt im *[GLOBAL]*-Bereich aktivieren. Der Name in der Konfigurationsvariable *LOGICS=RepeaterLogic* muß dabei exakt mit der Abschnittsbezeichnung *[RepeaterLogic]* übereinstimmen, also auch in Groß- und Kleinschreibung.

Datei */etc/svxlink/svxlink.conf*

[GLOBAL]

LOGICS=...,RepeaterLogic,...

```

[RepeaterLogic]
TYPE=Repeater
RX=Rx
TX=Tx
MODULES=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleMetarInfo
CALLSIGN=DL1ABC-L
SHORT_IDENT_INTERVAL=60
LONG_IDENT_INTERVAL=120
IDENT_ONLY_AFTER_TX=4
#EXEC_CMD_ON_SQL_CLOSE=500
EVENT_HANDLER=/usr/share/svxlink/events.tcl
RGR_SOUND_DELAY=250
#REPORT CTCSS=136.5
#TX CTCSS=ALWAYS
MACROS=Macros
LINKS=LinkToR4
FX_GAIN_NORMAL=0
FX_GAIN_LOW=-12
ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink
MUTE_RX_ON_TX=1
#NO_REPEAT=1
SQL_FLAP_SUP_MIN_TIME=500
SQL_FLAP_SUP_MAX_COUNT=7
OPEN_ON_1750=0
OPEN_ON_SQL=500
CLOSE_ON_SEL5=12345
OPEN_SQL_FLANK=CLOSE
#OPEN_ON CTCSS=250.6:20
QSO_RECORDER=7:QsoRecorder

```

Beschreibung

TYPE

Definiert den Logictyp, also Simplex-, Repeater- (oder Phonelogik), z.B. TYPE=Simplex

RX

Verweis auf den Abschnitt innerhalb der /etc/svxlink/svxlink.conf, in dem der Empfänger (Rx) konfiguriert wird, z.B. RX=Rx

TX

Verweis auf den Abschnitt innerhalb der /etc/svxlink/svxlink.conf, in dem der Sender (Tx) konfiguriert wird, z.B. TX=Tx

MODULES

Kommaseparierte Liste von Modulen, die innerhalb dieser Logik aktiviert und zur Verfügung stehen sollen, z.B. MODULE=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleMetarInfo

CALLSIGN

Rufzeichen, unter dem dieser SimplexLink arbeiten soll, z.B. CALLSIGN=DL1ABC-L

SHORT_IDENT_INTERVAL

Interval in Minuten, nachdem eine kurze Identifizierung erfolgt. Default: Ausgabe des Rufzeichens. Konfiguration in der SimplexLogic.tcl.

z.B. SHORT_IDENT_INTERVAL=60

LONG_IDENT_INTERVAL

Interval in Minuten, nachdem eine lange Identifizierung erfolgt. Default: Ausgabe des Rufzeichens und der Uhrzeit. Konfiguration in der SimplexLogic.tcl. z.B. LONG_IDENT_INTERVAL=120

IDENT_ONLY_AFTER_TX

Dieser Parameter steuert wann die Identifikation ausgeführt wird. Standardmäßig wird die Identifikation jedes Mal ausgeführt, wenn die im Parameter SHORT_IDENT_INTERVAL konfigurierte Zeit (in Minuten) abgelaufen ist. Wird der Parameter aktiviert, so erfolgt die Identifizierung nur, wenn gerade gesendet wird. Gedacht ist dieses Feature für den Fall, dass der Node ein Repeater über HF ansteuert und die Identifizierung dieses Links nicht notwendig ist bzw. stören würde.

z.B. IDENT_ONLY_AFTER_TX=1

EXEC_CMD_ON_SQL_CLOSE

Zeit in Millisekunden nach dem Schließen der Rauschsperrung, innerhalb dieser ein Kommando auch ohne abschließende Raute (#) als Kommando erkannt wird. Dieser Parameter ist gedacht für Funkgeräte, bei denen das Senden von DTMF-Zeichen relativ schwierig ist (VX-2). Das Einschalten dieses Parameters kann ungewollte Effekte hervorrufen, z.B. dass mitten im QSO ein DTMF-Zeichen erkannt und automatisch ein Modul gestartet wird.

EVENT_HANDLER

Die Position des für diese Logik zuständige Eventhandler-File, z.B. EVENT_HANDLER=/usr/share/svxlink/events.tcl

RGR_SOUND_DELAY

Verzögerungszeit in Millisekunden nach dem Schließen der Squelch um einen Roger-Piep auszulösen, z.B. RGR_SOUND_DELAY=250

REPORT CTCSS=136.5

Wenn konfiguriert, erfolgt die Ansage der konfigurierten CTCSS-Frequenz sobald man mit * die manuelle Identifizierung auslöst.

z.B. REPORT CTCSS=136.5. Wenn der RX keinen CTCSS-Ton benötigt sollte dieser Parameter auskommentiert werden.

TX CTCSS

Konfiguriert in einer RepeaterLogic, unter welchen Umständen ein CTCSS-Ton mit ausgesendet wird: ALWAYS oder SQL_OPEN. Im ersten Fall wird der CTCSS-Ton zugeschaltet, sobald die Station auf Sendung geht. Im zweiten Fall erst wenn eine Audioausgabe erfolgt, welche per HF empfangen wurde. Dieser Parameter wird nur bei Relaisstationen berücksichtigt.

Mögliche Parameter: ALWAYS,SQL_OPEN,LOGIC,MODULE,ANNOUNCEMENT

MACROS

Namen des Abschnittes innerhalb der /etc/svxlink/svxlink.conf, die die Macrodefinitionen enthält

z.B. MACROS=Macros Es wird dann in der svxlink.conf ein Abschnitt erwartet der mit [Macros] beginnt

LINKS

Namen des Abschnittes innerhalb der /etc/svxlink/svxlink.conf, der die Link-Definitionen enthält, also die Kopplung mit anderen Logiken auf dem selben SvxLink-Node ermöglicht, z.B. LINKS=Link_to_R2

FX_GAIN_NORMAL

Lautstärke der Ansagen und Sounds wenn auf dem Relais kein weiterer Funkverkehr durchgeführt wird. In diesem Falle werden alle Ansagen um den konfigurierten Betrag (in dB) abgesenkt (oder auch verstärkt).

z.B. FX_GAIN_NORMAL=0

FX_GAIN_LOW

Lautstärke der Ansagen und Sounds wenn auf dem Relais Funkverkehr durchgeführt wird. In diesem Falle werden alle Ansagen um den konfigurierten Betrag (in dB) abgesenkt.

z.B. FX_GAIN_LOW=-12

ACTIVATE_MODULE_ON_LONG_CMD

Automatische Aktivierung eines konfigurierten Modules, wenn ein langes Kommando mit der konfigurierten Anzahl von Digits empfangen wird. D.h., das Modul muß vorher nicht erst explizit aktiviert werden.

z.B. ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink -> startet das Modul EchoLink, wenn ein mindestens 4stelliges Kommando empfangen wurde

MUTE_RX_ON_TX

Wenn =1, dann wird das RX-Audio-Signal während der Sendung stumm geschaltet. Wenn nicht erforderlich sollte dieser Parameter auskommentiert werden.

NO_REPEAT

Wenn =1, dann gibt SvxLink NICHT das empfangene Audiosignal an den TX weiter. Nutzen sie diese Einstellung, wenn Sie einen Repeater betreiben, bei dem die NF-Zweige von TX und RX direkt gekoppelt sind. Der Nachteil ist in diesem Fall, dass SvxLink die empfangene RX-NF nicht weiterverarbeiten kann (Clipper, Deemphasis, DTMF-Töne ausblenden, etc.). Wird die NF innerhalb von SvxLink verarbeitet, so entsteht systembedingt eine geringe Verzögerung, die für den praktischen Funkbetrieb aber keine große Rolle spielt.

IDLE_TIMEOUT

Anzahl der Sekunden bis der Sender nach dem Auftasten wieder abfällt (Abfallzeit).

z.B. IDLE_TIMEOUT=10

OPEN_ON_1750

Zeit in Millisekunden, die ein Rufton von 1750Hz empfangen werden muß bis der Repeater öffnet. Ein Wert von 0 deaktiviert das Öffnen auf einen Rufton.

OPEN_ON CTCSS

Wenn dieser Parameter konfiguriert wurde, öffnet der Repeater auf den entsprechenden CTCSS-Ton (PL). Der Eingabesyntax ist ton_frequenz:min_länge. Die Tonfrequenz wird in Hertz angegeben und die Tonlänge in Millisekunden. z.B. für einen CTCSS-Ton von 136.5Hz und einer gewünschten Tonlänge von 1 Sekunde muß OPEN_ON_CTCSS=136:1000 eingetragen werden.

OPEN_ON_DTMF

Bei diesem Parameter kann festgelegt werden, ob ein einzelnes DTMF-Digit den Repeater öffnen kann. z.B. OPEN_ON_DTMF=*

OPEN_ON_SQL

Bei diesem Parameter wird die Zeit (in Millisekunden) einer geöffneten Squelch als Kriterium für das Öffnen des Repeaters herangezogen.

z.B. OPEN_ON_SQL=500 -> Die Rauschsperrung muß mindestens 0.5 Sekunden geöffnet bleiben, damit der Repeater öffnet

OPEN_ON_SEL5

Öffnet den Repeater wenn der entsprechende Mehrtonruf ordnungsgemäß empfangen wurde, z.B. OPEN_ON_SEL5=02311

CLOSE_ON_SEL5

Schließt den Repeater wenn der entsprechende Mehrtonruf ordnungsgemäß empfangen wurde, z.B. CLOSE_ON_SEL5=02123

OPEN_SQL_FLANK

Hier kann festgelegt werden, ob der Repeater auf eine steigende oder fallende Flanke öffnet. Voraussetzung ist, dass entweder OPEN_ON_SQL oder OPEN_ON CTCSS als Öffnungs-Kriterium konfiguriert wurden.

Bei OPEN_SQL_FLANK=OPEN öffnet der Repeater und startet die Übertragung des Audiostreams augenblicklich. Es wird keine Identifizierung ausgegeben.

Bei OPEN_SQL_FLANK=CLOSE öffnet der Repeater erst, wenn die Squelch wieder schließt. In diesem Fall wird eine Identifizierung ausgesendet.

IDLE_SOUND_INTERVAL

Wenn der Repeater geöffnet ist, aber kein Funkverkehr darüber abgewickelt wird, dann kann mit der Konfiguration dieses Parameters die Ausgabe eines Sounds erreicht werden. Der Wert entspricht dabei der Zeit zwischen diesen "Erinnerungstönen".

z.B. IDLE_SOUND_INTERVAL=1000

SQL_FLAP_SUP_MIN_TIME

Dieser Parameter ermöglicht das Schließen des Repeaters im Fall von Interferenzen (flatternde Rauschsperr). Der Wert ist die Zeit in Millisekunden, die ein Signal mindestens anliegen muß um als reales Rx-Signal erkannt zu werden. Gute Werte bewegen sich im Bereich von 500ms - 1000ms

SQL_FLAP_SUP_MAX_COUNT

Dieser Wert gibt die maximale Anzahl von kurzen Squelchöffnungen (sog. Bursts) an, die empfangen werden dürfen. Wird dieser Wert überschritten, so wird der Sender des Repeaters abgeschaltet. Eingünstiger Wert ist 5 bis 10.

SEL5_MACRO_RANGE

Definiert einen Bereich für eingehende Selektivrufe, die als Macro erkannt werden sollen.

z.B. SEL5_MACRO_RANGE=01200, 01399 Wird bei dieser Konfiguration eine 5Ton-Folge "01225" empfangen, so geht SvxLink davon aus, dass im [Macro]-Bereich ein entsprechender Eintrag konfiguriert wurde und versucht diesen auszuführen. Andere 5Ton-Folgen, wie z.B. "23312" werden von SvxLink ignoriert.

DEFAULT_LANG

Definiert die Sprachumgebung und ist wichtig für das Finden der jeweiligen Sprachumgebung. Wird der Eintrag nicht angegeben, so wird die Sprachumgebung des Linuxsystemes genutzt.

z.B. DEFAULT_LANG=de_DE

QSO_RECORDER

Kommando und Verweis auf den Konfigurationsabschnitt in dem der QSO-Recorder konfiguriert wird. Die Nummer vor dem Doppelpunkt definiert das Aktivierungs-/Deaktivierungskommando. Die Bezeichnung danach die Bezeichnung des Konfigurationsabschnitts.

Bsp: OSQ_RECORDER=8:QsoRecorder

Mit 81 # wird der QSO-Recorder gestartet und mit 80 # gestoppt.

SimplexLogic (2011-01-17 11:12)

Um einen einfachen Link einzurichten, müssen Sie in der Datei /etc/svxlink/svxlink.conf den Abschnitt [SimplexLogic] bearbeiten und diesen Abschnitt im [GLOBAL]-Bereich aktivieren. Der Name in der Konfigurationsvariable LOGICS=SimplexLogic muß dabei exakt mit der Abschnittsbezeichnung [SimplexLogic] übereinstimmen, also auch in Groß- und Kleinschreibung.

Datei /etc/svxlink/svxlink.conf

```
[GLOBAL]
```

```
LOGICS=...,SimplexLogic,...
```

```
[SimplexLogic]
```

```
TYPE=Simplex
```

```
RX=Rx
```

```
TX=Tx
```

```
MODULES=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleMetarInfo
```

```
CALLSIGN=DL1ABC-L
```

```
SHORT_IDENT_INTERVAL=60
```

```
LONG_IDENT_INTERVAL=120
```

```
IDENT_ONLY_AFTER_TX=4
```

```
#EXEC_CMD_ON_SQL_CLOSE=500
```

```
EVENT_HANDLER=/usr/share/svxlink/events.tcl
```

```
RGR_SOUND_DELAY=250
```

```
#REPORT CTCSS=136.5
```

```
#TX CTCSS=ALWAYS
```

```
MACROS=Macros
```

```
LINKS=LinkToR4
```

```
FX_GAIN_NORMAL=0
```

```
FX_GAIN_LOW=-12
```

```
ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink
```

```
ACTIVATE_MODULE_ON_STARTUP=EchoLink
```

```
MUTE_RX_ON_TX=1
```

```
#OSQ_RECORDER=8:QsoRecorder
```

```
#STATE_PTY=/var/run/svxlink/state
```

Beschreibung

TYPE

Definiert den Logictyp, also Simplex-, Repeater- oder Phonelogik

z.B. TYPE=Simplex

RX

Verweis auf den Abschnitt innerhalb der /etc/svxlink/svxlink.conf, in dem der Empfänger (Rx) konfiguriert wird,

z.B. RX=Rx

TX

Verweis auf den Abschnitt innerhalb der /etc/svxlink/svxlink.conf, in dem der Sender (Tx) konfiguriert wird, z.B.

TX=Tx

MODULES

Kommaseparierte Liste von Modulen, die innerhalb dieser Logik aktiviert und zur Verfügung stehen sollen, z.B. MODULE=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleMetarInfo

CALLSIGN

Rufzeichen, unter dem dieser SimplexLink arbeiten soll, z.B. CALLSIGN=DL1ABC-L

SHORT_IDENT_INTERVAL

Interval in Minuten, nachdem eine kurze Identifizierung erfolgt. Default: Ausgabe des Rufzeichens. Konfiguration in der SimplexLogic.tcl.

z.B. SHORT_IDENT_INTERVAL=60

LONG_IDENT_INTERVAL

Interval in Minuten, nachdem eine lange Identifizierung erfolgt. Default: Ausgabe des Rufzeichens und der Uhrzeit. Konfiguration in der SimplexLogic.tcl. z.B. LONG_IDENT_INTERVAL=120

IDENT_ONLY_AFTER_TX

Dieser Parameter steuert wann die Identifikation ausgeführt wird. Standardmäßig wird die Identifikation jedes Mal ausgeführt, wenn die im Parameter SHORT_IDENT_INTERVAL konfigurierte Zeit (in Minuten) abgelaufen ist. Wird der Parameter aktiviert, so erfolgt die Identifizierung nur, wenn gerade gesendet wird. Gedacht ist dieses Feature für den Fall, dass der Node ein Repeater über HF ansteuert und die Identifizierung dieses Links nicht notwendig ist bzw. stören würde.

z.B. IDENT_ONLY_AFTER_TX=1

EXEC_CMD_ON_SQL_CLOSE

Zeit in Millisekunden nach dem Schließen der Rauschsperrung, innerhalb dieser ein Kommando auch ohne abschließende Raute (#) als Kommando erkannt wird. Dieser Parameter ist gedacht für Funkgeräte, bei denen das Senden von DTMF-Zeichen relativ schwierig ist (VX-2). Das Einschalten dieses Parameters kann ungewollte Effekte hervorrufen, z.B. dass mitten im QSO ein DTMF-Zeichen erkannt und automatisch ein Modul gestartet wird.

EVENT_HANDLER

Die Position des für diese Logik zuständige Eventhandler-File, z.B. EVENT_HANDLER=/usr/share/svmlink/events.tcl

RGR_SOUND_DELAY

Verzögerungszeit in Millisekunden nach dem Schließen der Squelch um einen Roger-Piep auszulösen, z.B. RGR_SOUND_DELAY=250

REPORT CTCSS=136.5

Wenn konfiguriert, erfolgt die Ansage der konfigurierten CTCSS-Frequenz sobald man mit * die manuelle Identifizierung auslöst.

z.B. REPORT CTCSS=136.5. Wenn der RX keinen CTCSS-Ton benötigt sollte dieser Parameter auskommentiert werden.

TX CTCSS

Konfiguriert in einer RepeaterLogic, unter welchen Umständen ein CTCSS-Ton mit ausgesendet wird: ALWAYS oder SQL_OPEN. Im ersten Fall wird der CTCSS-Ton zugeschaltet, sobald die Station auf Sendung geht. Im zweiten Fall erst wenn eine Audioausgabe erfolgt. Dieser Parameter wird nur bei Relaisstationen berücksichtigt.

z.B. TX CTCSS=ALWAYS

MACROS

Namen des Abschnittes innerhalb der /etc/svmlink/svmlink.conf, die die Macrodefinitionen enthält

z.B. MACROS=Macros Es wird dann in der svmlink.conf ein Abschnitt erwartet der mit [Macros] beginnt

LINKS

Namen des Abschnittes innerhalb der `/etc/svxlink/svxlink.conf`, der die Link-Definitionen enthält, also die Kopplung mit anderen Logiken auf dem selben SvxLink-Node ermöglicht, z.B. `LINKS=Link_to_R2`

`FX_GAIN_NORMAL`

Lautstärke der Ansagen und Sounds wenn auf dem Relais kein weiterer Funkverkehr durchgeführt wird. In diesem Falle werden alle Ansagen um den konfigurierten Betrag (in dB) abgesenkt (oder auch verstärkt).

z.B. `FX_GAIN_NORMAL=0`

`FX_GAIN_LOW`

Lautstärke der Ansagen und Sounds wenn auf dem Relais Funkverkehr durchgeführt wird. In diesem Falle werden alle Ansagen um den konfigurierten Betrag (in dB) abgesenkt.

z.B. `FX_GAIN_LOW=-12`

`ACTIVATE_MODULE_ON_LONG_CMD`

Automatische Aktivierung eines konfigurierten Modules, wenn ein langes Kommando mit der konfigurierten Anzahl von Digits empfangen wird. D.h., das Modul muß vorher nicht erst explizit aktiviert werden.

z.B. `ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink` -> startet das Modul EchoLink, wenn ein mindestens 4stelliges Kommando empfangen wurde

`MUTE_RX_ON_TX`

Wenn =1, dann wird das RX-Audio-Signal während der Sendung stumm geschaltet. Wenn nicht erforderlich sollte dieser Parameter auskommentiert werden.

`SEL5_MACRO_RANGE`

Definiert einen Bereich für eingehende Selektivrufe, die als Macro erkannt werden sollen.

z.B. `SEL5_MACRO_RANGE=01200,01399` Wird bei dieser Konfiguration eine 5Ton-Folge "01225" empfangen, so geht SvxLink davon aus, dass im [Macro]-Bereich ein entsprechender Eintrag konfiguriert wurde und versucht diesen auszuführen. Andere 5Ton-Folgen, wie z.B. "23312" werden von SvxLink ignoriert.

`DEFAULT_LANG`

Definiert die Sprachumgebung und ist wichtig für das Finden der jeweiligen Sprachumgebung. Wird der Eintrag nicht angegeben, so wird die Sprachumgebung des Linuxsystemes genutzt.

z.B. `DEFAULT_LANG=de_DE`

`VOICELOGGER_DIR`

Verzeichnis, in welches die Mitschnitte des Funkverkehrs auf dem Repeater gespeichert werden sollen. z.B. `VOICELOGGER_DIR=/var/spool/svxlink/voice`

`VOICELOGGER_CMD`

Kommando, mit welchem der Mitschnitt gestartet bzw. gestoppt werden kann, z.B. `VOICELOGGER_CMD=8` Im praktischen Betrieb startet das Kommando `81 #` den Mitschnitt und das Kommando `80 #` stoppt die Aufzeichnung wieder.

`QSO_RECORDER`

Kommando und Verweis auf den Konfigurationsabschnitt in dem der QSO-Recorder konfiguriert wird. Die Nummer vor dem Doppelpunkt definiert das Aktivierungs-/Deaktivierungskommando. Die Bezeichnung danach die Bezeichnung des Konfigurationsabschnitts.

Bsp: `OSQ_RECORDER=8:QsoRecorder`

Mit `81 #` wird der QSO-Recorder gestartet und mit `80 #` gestoppt.

`STATE_PTY`

Mit dieser Variable kann man ein UNIX PTY definieren über das Statusinformationen bzw. Ereignisse ausgegeben werden. Diese Informationen können von eigenen Programmen ausgewertet werden.

Das Ausgabeformat ist ein einfaches Textformat in Form von Leerzeichen als Trennung untereinander. SvXLink erzeugt einen Symlink zum aktuellen Slave-PTY. Aus diesem Grund muß SvXLink eine Schreibberechtigung auf das Verzeichnis besitzen in welchem dieser Symlink erzeugt werden soll. Ein Mitlesen der erzeugten Informationen ist mit

```
cat /path/to/pty
```

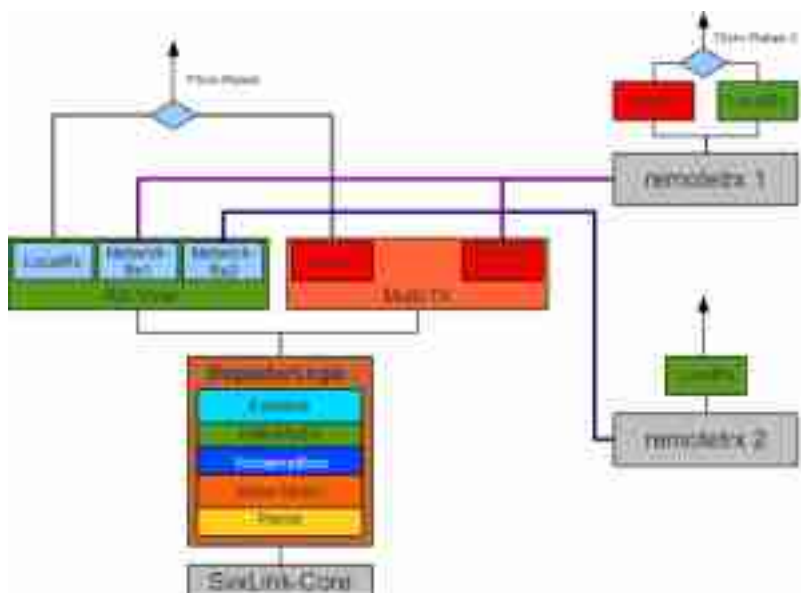
möglich, nachdem SvXLink gestartet wurde.

Beispiel:

```
STATE_PTY=/tmp/state _pty
```

remotetrx (2011-01-17 16:52)

Ein Remotetrx kann für den Betrieb von abgesetzten Sendern oder Empfängern genutzt werden. Die Einsatzfälle sind recht vielfältig. SvXLink bietet eine Voterfunktion, bei der die von verschiedenen Empfängern eingehenden Signale analysiert und das jeweils "beste" Empfangssignal zum Sender/den Sendern durchgeschaltet wird. Das folgende Bild soll eine Anwendungsmöglichkeit aufzeigen.

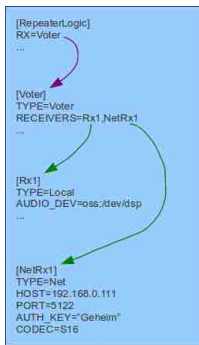


Prinzip des Remotetrx

Die Konfiguration auf der SvXLink-Base erfolgt in der /etc/svmlink/svmlink.conf. Auf der remotetrx-Seite erfolgt die Konfiguration in der Datei /etc/svmlink/remotetrx.conf.

1) SvXLink-Base

Die grundsätzliche Konfiguration des RX-Pfades für den Voter ist im folgenden Bild dargestellt



Konfigurationsprinzip eines Voters auf der SvxLink-Seite (/etc/svxlink/svxlink.conf)

/etc/svxlink/svxlink.conf

[RepeaterLogic]

...

RX=Voter

TX=Multi

...

[Voter]

TYPE=Voter

RECEIVERS=Rx1,NetRx1

VOTING _DELAY=200

[Rx1]

...

hier ist ein LocalRx konfiguriert, siehe Abschnitt RX-Konfiguration"

...

Abschnitt für den RemoteRx

[NetRx1]

TYPE=Net

HOST=192.168.0.99

TCP _PORT=5213

AUTH _KEY="βehr _geheim"

CODEC=S16

Beschreibung

TYPE

Gibt an ob es sich um eine lokale (=Local) oder remote-Komponente (=Net) handelt.

HOST

Die IP-Adresse oder Hostname auf dem der Remotetrx läuft.

TCP _PORT

Der Port, über den die TCP-Kommunikation erfolgt

AUTH _KEY

Das Passwort, mit dem sich remotetrx und SvxLink-Basis gegenseitig authentifizieren. Groß- und Kleinschreibung ist wichtig.

CODEC

Der zu verwendende Sprachcodec, mit dem die Audiodaten zwischen SvxLink und remotetrx übertragen werden. Es stehen folgende Codecs zur Verfügung:

Macros (2011-01-17 22:40)

Macros sind Aktionen auf dem Node, die beim Empfang eines bestimmten Kommandos ausgelöst werden. Allgemein werden bei SvxLink Macros durch Voranstellen des DTMF-Digit's "D" eingeleitet, z.B. *D1234 #* Konfiguriert werden Macros in einem eigenen Abschnitt, der durch einen Abschnittsnamen gekennzeichnet wird und auf den von einem Logik-Konfigurationsabschnitt (in der */etc/svxlink/svxlink.conf*) aus verwiesen werden muß.

Aktivierung der Macros, durch einen logikspezifischen Eintrag in der */etc/svxlink/svxlink.conf*

```
[RepeaterLogic]
...
MACROS=Macro _repeater
...
```

In der */etc/svxlink/svxlink.conf* muß jetzt ein Abschnitt *[Macro _repeater]* vorhanden sein, in dem die jeweiligen Macros definiert werden.

```
[Macro _repeater]
0=SelCallEnc:0112345 #
1=EchoLink:9999 #
2=MetarInfo:12 #
...
```

Im ersten Beispiel wird beim Empfang des Macros "D0 #" ein Selektivruf im Modus "01" (=ZVEI-1) und der Nummer "12345" ausgelöst.

Im zweiten Beispiel erfolgt beim Empfang des Macros "D1 #" der Aufbau einer Echolink-Verbindung mit dem EchoLink-Testserver.

Das dritte Beispiel startet beim Empfang von "D2 #" das Modul MetarInfo und die Ausgabe der Wettermeldungen der 12. konfigurierten Wetterstation.

Voraussetzung für alles ist natürlich, dass die jeweiligen Module entsprechend konfiguriert wurden.

Link(ing) (2011-01-17 23:03)

Es geht hierbei um die Möglichkeit, auf Anforderung zwei oder mehr Logiken miteinander zu verbinden. Interessant ist diese Anwendung, wenn z.B. an einem Standort sowohl ein 2m als auch 70cm-Repeater betrieben werden. Diese können bei Bedarf, z.B. während eines Rundspruchs miteinander "verlinkt" werden. In Kürze wird die Möglichkeiten bestehen, die Art und Weise freier definieren zu können, z.B. die automatische und permanente Verbindung beim Start von SvxLink usw.

Bitte dieses Linking nicht mit den Simplex-Links verwechseln.

Aktivierung durch einen Eintrag im jeweiligen Bereich einer Logik innerhalb der /etc/svxlink/svxlink.conf:

```
[RepeaterLogic _70cm]
```

```
...
```

```
LINKS=TestLink
```

```
...
```

```
[RepeaterLogic _2m]
```

```
...
```

```
LINKS=TestLink
```

```
...
```

```
[TestLink]
```

```
NAME=2Relais
```

```
CONNECT _LOGICS=RepeaterLogic _70cm:32:70cm,RepeaterLogic _2m:33:2m
```

```
DEFAULT _ACTIVE=1
```

```
AUTOACTIVATE _ON _SQL=1
```

```
TIMEOUT=300
```

Beschreibung

CONNECT _LOGICS

Hier werden die Logiken und Befehle konfiguriert, die miteinander verbunden werden sollen. Das Format ist Logicname1:Befehl1:Bezeichnung1,Logikname2:Befehl2:Bezeichnung1

Beispiel:

```
CONNECT _LOGICS=DBOABC:34:ABC,DBOXYZ:72:XYZ
```

Um beide Logiken miteinander zu verbinden muß von DBOABC aus das Kommando 341 # aufgerufen werden, von DBOXYZ ist es das Kommando 721 #.

Die kommandos 340 # sowie 720 # führen von der jeweiligen Logik aus zur Trennung der Verbindung.

DEFAULT _ACTIVE

Wenn dieser Parameter =1 ist, werden beide Logiken sofort nach dem Start von SvxLink verbunden.

TIMEOUT

Zeit in Sekunden, nach der die verbundenen Logiken wieder getrennt werden, falls kein Funkverkehr darüber geführt wurden.

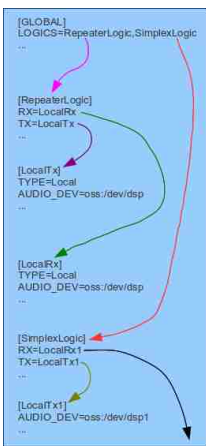
NAME

Bezeichnung für diese Konfiguration, wird beim Verbinden den TCL-Skripten übergeben und kann z.B. beim Verbinden/Trennen mit angesagt werden

svxlink.conf (2011-01-18 20:30)

Die Datei /etc/svxlink/svxlink.conf ist die Hauptkonfigurationsdatei von SvxLink. In ihr wird festgelegt welche Logik(en) betrieben werden soll(en) (Repeater und/oder Simplex), welchen Rauschsperrtyp man für den jeweiligen Rx nutzen will, ob es sich um eine lokale Station handelt oder ob man abgesetzte Stationen (sogenannte remotetrx's) mit anbinden möchte usw.

Je nach eigener Zielstellung und vorhandener Hardware, die angebunden werden soll, kann diese Datei eine nicht ganz unerhebliche Größe erreichen, so dass man u.U. ein wenig die Übersicht verlieren kann. Das nächste Bild zeigt den grundsätzlichen strukturellen Aufbau dieser Datei.



Struktur innerhalb der /etc/svxlink/svxlink.conf

Die Namen innerhalb der eckigen Klammern ([Name]) kennzeichnen dabei Abschnitte, die sozusagen Teil- bzw. Unterkonfigurationen enthalten. Mit Ausnahme von [GLOBAL] sind diese Namen frei wählbar. Was in diesem Abschnitt konfiguriert wird, muss mit dem Parameter TYPE=xxx definiert werden.

Achten Sie aber nach Möglichkeit auf die Verwendung von sprechenden Bezeichnungen, wir empfehlen für eine SSimplexLogicäuch einen entsprechenden Namen zu verwenden, sonst kommen Sie früher oder später durcheinander! Achten Sie weiterhin auf eine exakte Übereinstimmung der Bezeichnungen insbesondere auf die Groß- und Kleinschreibung.

Die Konfigurationsabschnitte der /etc/svxlink/svxlink.conf

```
[GLOBAL]
MODULE_PATH=/usr/lib/svxlink
LOGICS=RepeaterLogic,AnalogPhone,GSMLogic,MicSpkrLogic1,MicSpkrLogic
CFG_DIR=/etc/svxlink.d
TIMESTAMP_FORMAT="%c"
CARD_SAMPLE_RATE=48000
#CARD_CHANNELS=1
LOCATION_INFO=LocationInfo
#LINKS=TestLink
```

Der [GLOBAL]-Abschnitt enthält grundsätzliche Definitionen, wie die Art und Anzahl der zu aktivierenden Logiken.

MODULE_PATH

Der Pfad, in dem die Modul-Bibliotheken zu finden sind.

LOGICS

Die Namen der Logiken, welche man gerne nutzen möchte.

CFG_DIR

Das Verzeichnis, in welchem die Konfigurationsdateien für die Module zu finden sind (z.B. ModuleMetarInfo.conf).

CARD_SAMPLE_RATE

Die Samplingrate [Hz], mit der auf der Soundkarte gearbeitet wird (8000, 16000, 22050, 44100, 48000). Einfache Soundkarten unterstützen oft nur 8000 und 48000.

CARD_CHANNELS

Definiert die Anzahl der genutzten Soundkartenkanäle, z.B. 1=mono, 2=stereo

TIMESTAMP_FORMAT

Format des Zeitstempels, der im Log erscheint, für detaillierte Informationen siehe C++ Formatdeskriptoren.

LOCATION_INFO

Aktiviert den Bereich für die APRS-Positions- und Statusmeldungen, die SvxLink periodisch generiert.

LINKS

Kommaseparierte Liste von Abschnittsbezeichnungen, die Konfigurationsparameter zum Verbinden von verschiedenen Logiken enthalten.

Logic-Abschnitt

Der Logic-Abschnitt entspricht einer kompletten Ablaufsteuerung inkl. RX- und TX-Handling. Zunächst ein Beispiel für eine Simplex-Logic (EchoLink-Link"). Die Parameter werden hier http://svxlink.de/?page_id=385 beschrieben

```
[SimplexLogic]
```

```
TYPE=Simplex
```

```
RX=Rx1
```

```
TX=Tx1
```

```
MODULES=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleTclVoiceMail
```

```
CALLSIGN=MYCALL
```

```
SHORT_IDENT_INTERVAL=60
```

```
LONG_IDENT_INTERVAL=60
```

```
#IDENT_ONLY_AFTER_TX=4
```

```
#EXEC_CMD_ON_SQL_CLOSE=500
```

```
EVENT_HANDLER=/usr/share/svxlink/events.tcl
```

```
DEFAULT_LANG=de_DE
```

```
RGR_SOUND_DELAY=0
```

```
REPORT CTCSS=136.5
```

```
#TX CTCSS=ALWAYS
```

```
MACROS=Macros
```

```
FX_GAIN_NORMAL=0
```

```
FX_GAIN_LOW=-12
```

```
#ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink
#QSO_RECORDER=8:QsoRecorder
#ONLINE_CMD=998877
#MUTE_RX_ON_TX=1
#MUTE_TX_ON_RX=1
#STATE_PTY=/var/run/svxlink/state
```

und hier für eine Repeatersteuerung. Die einzelnen Parameter werden hier http://svxlink.de/?page_id=383 beschrieben.

```
[RepeaterLogic]
TYPE=Repeater
RX=Rx1
TX=Tx1
MODULES=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleTclVoiceMail
CALLSIGN=MYCALL
SHORT_IDENT_INTERVAL=10
LONG_IDENT_INTERVAL=60
#IDENT_ONLY_AFTER_TX=4
#EXEC_CMD_ON_SQL_CLOSE=500
EVENT_HANDLER=/usr/share/svxlink/events.tcl
DEFAULT_LANG=de_DE
RGR_SOUND_DELAY=0
REPORT CTCSS=136.5
#TX CTCSS=SQL_OPEN
MACROS=Macros
#SEL5_MACRO_RANGE=03400,03499
FX_GAIN_NORMAL=0
FX_GAIN_LOW=-12
#QSO_RECORDER=8:QsoRecorder
#NO_REPEAT=1
IDLE_TIMEOUT=30
OPEN_ON_1750=1000
#OPEN_ON CTCSS=136:2000
#OPEN_ON_DTMF=*
#OPEN_ON_SQL=5000
#OPEN_ON_SEL5=01234
#OPEN_SQL_FLANK=OPEN
#OPEN_ON_SQL_AFTER_RPT_CLOSE=10
IDLE_SOUND_INTERVAL=3000
#SQL_FLAP_SUP_MIN_TIME=1000
#SQL_FLAP_SUP_MAX_COUNT=10
#ACTIVATE_MODULE_ON_LONG_CMD=4:EchoLink
#IDENT_NAG_TIMEOUT=15
#IDENT_NAG_MIN_TIME=2000
#ONLINE_CMD=998877
#STATE_PTY=/var/run/svxlink/state
```

RX-Adapter

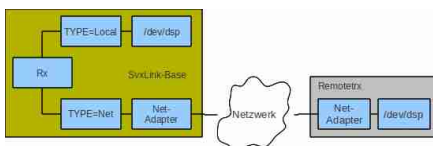
Der RX-Adapter realisiert den Empfangsteil der SvxLink-Applikation. Er ist für den Empfangssignalweg und damit verbunden Funktionen zuständig. Hierzu gehören:

- Aufbereitung und Verarbeitung des Empfangssignales (NF-seitig)
- Generierung eines Squelch-Kriteriums
- Dekodierung von Tonfrequenzen wie DTMF, Sel5, CTCSS und normalen Ruftönen

Beispiel:

```
[Rx1]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
SQL_DET=VOX
SQL_START_DELAY=10
SQL_DELAY=10
SQL_HANGTIME=3000
#SQL_TIMEOUT=600
VOX_FILTER_DEPTH=20
VOX_THRESH=1000
CTCSS_FQ=136.5
CTCSS_THRESH=10
SERIAL_PORT=/dev/ttyS0
SERIAL_PIN=CTS
#EVDEV_DEVNAME=/dev/input/by-id/usb-SYNIC_SYNIC_Wireless_Audio-event-if03
#EVDEV_OPEN=1,163,1
#EVDEV_CLOSE=1,163,0
#SIGLEV_DET=TONE
SIGLEV_SLOPE=1
SIGLEV_OFFSET=0
#TONE_SIGLEV_MAP=100,84,60,50,37,32,28,23,19,8
SIGLEV_OPEN_THRESH=30
SIGLEV_CLOSE_THRESH=10
DEEMPHASIS=0
#SQL_TAIL_ELIM=300
#PREAMP=6
PEAK_METER=0
DTMF_DEC_TYPE=INTERNAL
DTMF_MUTING=1
DTMF_HANGTIME=100
DTMF_SERIAL=/dev/ttyS0
#DTMF_MAX_FWD_TWIST=8
#DTMF_MAX_REV_TWIST=4
#1750_MUTING=1
#SEL5_DEC_TYPE=INTERNAL
#SEL5_TYPE=ZVEI1
```

Es stehen grundsätzlich zwei Varianten zur Verfügung: ein LocalRx, der die auf dem Server vorhandenen Audiointerfaces direkt anspricht (/dev/dsp bzw. hwplug:0) sowie ein NetRx, der den Empfangspfad eines über das Netzwerk angebandenen Servers (remotetrx) ansprechen kann. Das folgende Bild stellt diese Möglichkeiten prinzipiell dar.



Rx-Typen (NetRx und LocalRx)

TYPE

Definiert ob es sich um einen lokalen oder einen per Netzwerk angeschlossenen Rx handelt, z.B. TYPE=Local

AUDIO _DEV

Das zu verwendende Audiodevice AUDIO _DEV=oss:/dev/dsp oder AUDIO _DEV=alsa:plughw:0

AUDIO _CHANNEL

Die Nummer des Stereo-Kanals, der verwendet wird. Im Gegensatz zum herkömmlichen Echolink lassen sich beide Stereokanäle getrennt verwenden, wenn es die Soundkarte unterstützt!

z.B. AUDIO _CHANNEL=0 oder AUDIO _CHANNEL=1

SQL _DET

SQL _DET=CTCSS -> Rauschsperrtyp ist CTCSS", das heißt, nur wenn ein CTCSS-Ton empfangen wird, gilt es als Öffnen der Rauschsperr

SQL _DET=SIGLEV -> Signal-Detector, der den am Eingang anliegenden Rauschpegel auswertet. Die Rauschsperr des RX muß offen sein und es muß eine Signalpegelkalibrierung durchgeführt werden (/usr/bin/siglevdetcal)

SQL _DET=VOX -> Rauschsperrtyp ist VOX, also praktisch die NF des angeschlossenen Funkgerätes

SQL _DET=SERIAL -> Rauschsperrtyp ist Serial, d.h. es wird ein über die serielle Schnittstelle eingehendes (externes) Signal genutzt.

SQL _DET=EVDEV -> Wertet den Status des Interfaces /dev/input/eventX aus, z.B. Taste auf einer USB-Soundkarte

SQL _START _DELAY

Zeit in ms, die vergehen muß, damit das konfigurierte Rauschsperrkriterium das Ereignis als Öffnen der Rauschsperr erkennt. Hier sind Werte >0 einzutragen, falls eine VOX als Rauschsperr verwendet wird, z.B. SQL _START _DELAY=100.

SQL _DELAY

Zeit in ms, um das das Ereignis SSQL _OPEN"verzögert wird (Weitergabe an das SvxLink-System), Einsatzfälle sind z.B beim SigLevDetector oder beim RX-Voting. Wertebereich 20-100

SQL _HANGTIME

Zeit in ms, die die Rauschsperr noch offen bleibt, auch nachdem der Rauschsperrdetector ein Schließen registriert hat, z.B. SQL _HANGTIME=75

EVDEV _DEVNAME

Konfiguriert das Geräte /dev/input welches für den EVDEV-Squelchdetektor genutzt wird.

EVDEV _OPEN

Nutzen Sie das Programm evttest um den Typ, Code und Wert herauszufinden umd die Rauschsperr zu öffnen, Bsp: erhalten Sie die Paramter typ=1,code=163,value=1 so ist der Parameter EVDEV _OPEN wie folgt zu konfigurieren: EVDEV _OPEN=1,163,1

EVDEV _CLOSE

Nutzen Sie das Programm evttest um den Typ, Code und Wert herauszufinden umd die Rauschsperr zu schließen, Bsp: erhalten Sie die Paramter typ=1,code=163,value=0 so ist der Parameter EVDEV _CLOSE wie folgt zu konfigurieren: EVDEV _CLOSE=1,163,0

CTCSS _FQ

Wenn SQL _DET=CTCSS, dann wird die hier konfigurierte CTCSS-Frequenz als Rauschsperrkriterium verwendet., z.B. CTCSS _FQ=136.5

CTCSS _THRESH

nicht mehr genutzt

CTCSS_OPEN_THRESH

Wenn als Rauschsperrtyp CTCSS konfiguriert wurde (SQL_DET=CTCSS), dann definiert dieser Parameter den Pegel für das Öffnen der Rauschsperrung. Dieser Pegel stellt dabei eine Art Signal/Nutzsignal-Verhältnis [dB] dar. Wenn Sie CTCSS-Modus 2 oder 3 nutzen, dann ist es hilfreich den Parameter CTCSS_SNR_OFFSET zu konfigurieren. Der Standardwert für CTCSS_OPEN_THRESH ist 15dB.

CTCSS_CLOSE_THRESH

Wenn als Rauschsperrtyp CTCSS konfiguriert wurde (SQL_DET=CTCSS), dann definiert dieser Parameter den Pegel für das Schließen der Rauschsperrung. Dieser Pegel stellt dabei eine Art Signal/Nutzsignal-Verhältnis [dB] dar. Wenn Sie CTCSS-Modus 2 oder 3 nutzen, dann ist es hilfreich den Parameter CTCSS_SNR_OFFSET zu konfigurieren. Der Standardwert für CTCSS_CLOSE_THRESH ist 9dB.

CTCSS_SNR_OFFSET

Dieser Parameter wird nur berücksichtigt, wenn Sie den Parameter CTCSS_MODE auf 0, 2 oder 3 gesetzt haben. Er kalibriert den ermittelten SNR Wert so, dass er einem realen SNR nahekommt. Dieser Parameter muß bei jeder Änderung der Werte CTCSS_FREQ, CTCSS_MODE, CTCSS_BPF_LOW und CTCSS_BPF_HIGH neu ermittelt werden. Es besteht keine Notwendigkeit, diesen Parameter zu setzen. In diesem Fall müssen Sie aber mehr mit den Parametern CTCSS_OPEN_THRESH und CTCSS_CLOSE_THRESH experimentieren um korrekte Einstellungen für die Rauschsperrung zu finden.

CTCSS_MODE

Diese Variable definiert die CTCSS-Erkennungsmethode. Folgende Werte können stehen zur Verfügung:

- 0 (Default) - Diese Methode ist die Standardeinstellung. Momentan ist es die Estimated SNRMethode
- 1 (Neighbour bins) - Diese Methode nutzt drei schmale Frequenzbänder (8Hz) zur Erkennung. Ein Band liegt im Zentrum des zu erkennenden Tones, die anderen darüber bzw. darunter. Diese Frequenzbänder werden genutzt um den Rauschpegel zu berechnen. In SvxLink wurde diese Methode lange Zeit verwendet, allerdings ist sie sehr langsam (450ms) und sollte nicht mehr angewendet werden. Allerdings bleibt sie für den Fall verfügbar, dass bei bestimmten Hardware-Konfigurationen die anderen Methoden u.U. nicht verwendet werden können.
- 2 (Estimated SNR) - Dies ist eine neuere Implementierung mit einigen Verbesserungen, z.B. einer wesentlich schnelleren Detektierungszeit (200ms). Es ist das Standardverfahren, wenn nicht konfiguriert wurde. Dieses Verfahren nutzt eine höhere Bandbreite um das Grundrauschen zu bestimmen und ist damit stabiler. Standardmäßig wird das gesamte CTCSS-Band ausgewertet, kann aber mit den Parametern CTCSS_CPF_LOW und CTCSS_BPF_HIGH angepasst werden.
- 3 (Estimated SNR+Phase) - Dieser Detektor ist noch im Experimentierstadium. Er ist etwas schneller und wesentlich schmalbandiger als die anderen Verfahren und die Detektierungszeit liegt bei etwa 150ms. Damit ist er unempfindlicher gegenüber angrenzenden Tönen allerdings auch unempfindlicher beim zu detektierenden Ton.

SERIAL_PORT

Wenn SQL_DET=SERIAL, dann ist hier der serielle Port einzustellen, über die der Rauschsperrstatus vom externen Rx geliefert wird, z.B. SERIAL_PORT=/dev/ttyUSB2

SERIAL_PIN

Wenn SQL_DET=SERIAL, dann ist hier der Pin und Pegel zu definieren, an dem das externe Signal anliegt, z.B. SERIAL_PIN=CTS -> wenn an CTS ein high-Pegel anliegt gilt die Rauschsperrung als offen, ist ein gegenteiliges Verhalten gewünscht, dann "!" davor setzen (Negierung)

SERIAL_SET_PINS

Setzt die Pins einer seriellen Schnittstelle auf einen definierten Wert (H oder L). Das kann beispielsweise genutzt werden um eine Referenzspannung zum Pegel-Vergleich zu erzeugen. Hierbei können nur zwei Pins verwendet werden: RTS oder DTR.

Beispiel: SERIAL_SET_PINS=RTS!DTR setzt RTS auf H- und DTR auf L-Pegel

TONE_SIGLEV_MAP

Diese Variable weist Töne Signalpegeln zu wenn DIGLEV_DET=TONE. Es ist eine kommaseparierte Liste von Werten im Bereich 0-100. Der erste Wert korreliert mit der Frequenz 5500Hz, der Zweite mit 5600Hz ... der letzte Wert mit 6400Hz. Welche Werte konfiguriert werden hängt von der Implementierung ab. Die Standardeinstellung ist 10,20,30,40,...100.

SIGLEV_DET

Wählen Sie hier den Typ des Signaldetektors. Es gibt zwei Auswahlmöglichkeiten: NOISE oder TONE. Der SIGLEV_DET wird nur benötigt, wenn Sie mehrere Empfänger in einer Voter-Konfiguration betreiben oder als Rauschsperrentyp SIGLEV konfiguriert wurde.

Der Detektortyp NOISE nutzt ein Bandpassfilter im Bereich 5-5.5kHz (bei CARDSAMPLE_RATE >= 16000) oder einen Hochpass (bei CARDSAMPLE_RATE = 8000) um den Betrag des Rauschanteils zu bestimmen.

Der Detektortyp TONE ist kein wirklicher Signalpegeldetektor sondern mehr ein Mechanismus um einen Pegel von einem Remote-TRX zu erhalten, der per HF angekoppelt ist.

SIGLEV_OPEN_THRESH

Wenn SQL_DET=SIGLEV, dann gibt dieser Wert den Pegel [%] an bei dem die Rauschsperre öffnet

SIGLEV_CLOSE_THRESH

Wenn SQL_DET=SIGLEV, dann gibt dieser Wert den Pegel [%] an bei dem die Rauschsperre schließt

SIGLEV_SLOPE

Wenn SQL_DET=SIGLEV, dann hier die "Verstärkung" des SigLev-Detectors eintragen, siehe Rauschsperrenkalibrierung.

SIGLEV_OFFSET

Wenn SQL_DET=SIGLEV, dann hier den "Offset" des SigLev-Detectors eintragen, siehe Rauschsperrenkalibrierung.

DEEMPHASIS

Höhenabsenkung des RX-Signales um den hier konfigurierten Betrag in [dB]

SQL_TAIL_ELIM

Dieser Parameter kann genutzt werden, um das Rauschen nach Beendigung einer Übertragung bis zum Einsetzen der Rauschsperre zu minimieren. Die empfangenen Audiodaten werden um den im Parameter SQL_TAIL_ELIM definierten Betrag in ms gepuffert. Beim Schließen der Rauschsperre werden die verbleibenden Audiopakete (einsetzendes Rauschen) nicht weiter übertragen. Achtung: Der gesamt über den Sender übertragene RX-Audiostream wird um diesen Betrag verzögert.

PREAMP

Das eingehende RX-Signal wird um den hier konfigurierten Betrag verstärkt [dB].

PEAK_METER

Gibt eine Warnmeldung im /var/log/svxlink aus, falls der anliegende RX-Audiopegel zu hoch sein sollte. (=1 Warnungen aktiviert, =0 Warnungen deaktiviert)

DTMF_DEC_TYPE

Gibt den verwendeten DTMF-Decodertyp an, z.B. DTMF_DEC_TYPE=INTERNAL verwendet den in SvxLink

eingebauten DTMF-Softwaredecoder, DTMF_DEC_TYPE=S54 erwartet einen an die serielle Schnittstelle angeschlossenen DTMF-Hardwaredecoder.

DTMF_MUTING

Wenn =1, dann werden empfangene DTMF-Töne nicht über den Sender ausgegeben, nur wenn RX und TX audioseitig nicht direkt gekoppelt sind und NO_REPEAT nicht =1 ist.

DTMF_HANGTIME

Ist die Zeit in ms, die ein DTMF-Signal mindestens empfangen werden muß, damit es sicher als solches erkannt werden kann. Empfohlener Wertebereich 50-100ms

DTMF_SERIAL

Serielle Schnittstelle, an die ein S54-DTMF-Dekoder angeschlossen ist, z.B. DTMF_SERIAL=/dev/ttyS1

DTMF_MAX_FWD_TWIST

DTMF besteht aus zwei verschiedenen Tonfrequenzen (niedrige und höhere), die zeitgleich gesendet werden und normalerweise die gleiche Amplitude haben sollten. Das ist praktisch nie der Fall. DTMF_MAX_FWD_TWIST gibt an, um wieviel dB niedriger die Amplitude des höheren Tones gegenüber dem niedrigeren Ton sein darf. z.B. DTMF_MAX_FWD_TWIST=8

DTMF_MAX_REV_TWIST

Wie DTMF_MAX_FWD_TWIST, nur umgekehrt (wenn der hohe Ton eine um DTMF_MAX_REV_TWIST höhere Amplitude haben darf.)

1750_MUTING

Wenn =1, dann wird bei Empfang des 1750Hz-Ruftones der Sender stummgetastet. Die Funktion ist nur möglich, wenn RX und TX audioseitig nicht direkt gekoppelt sind und NO_REPEAT nicht =1 ist.

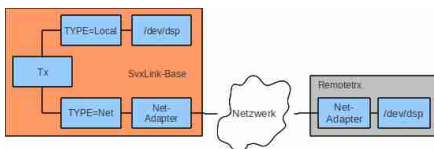
SEL5_DEC_TYPE=INTERNAL

Gibt an, ob ein interner oder externer Mehrtondekodeur genutzt werden soll, momentan darf hier nur INTERNAL konfiguriert werden

SEL5_TYPE

Definiert den zu dekodierenden Mehrton-Standard (ZVEI1,EEA,EIA,CCITT,CCIR,...), z.B. SEL5_TYPE=ZVEI1

TX-Adapter



Tx-Typen (NetTx und LocalTx)

Beispiel:

[Tx1]

TYPE=Local

AUDIO_DEV=alsa:plughw:0

AUDIO_CHANNEL=0

PTT_PORT=NONE

PTT_PIN=DTRRTS

#PTT_HANGTIME=1000

TIMEOUT=300

TX_DELAY=500

#CTCSS _FQ=136.5
#CTCSS _LEVEL=9
PREEMPHASIS=0
DTMF _TONE _LENGTH=100
DTMF _TONE _SPACING=50
DTMF _TONE _AMP=-18

TYPE

Gibt den Typ an, also lokaler Sender oder über Netzwerk angebunden, z.B. TYPE=Local

AUDIO _DEV

Das Audiogerät (Lautsprecherausgang) der Soundkarte, der als TX genutzt wird. Beim OpenSound-System ist vor der Gerätedefinition ein „oss“ voranzustellen, z.B. AUDIO _DEV=oss:/dev/dsp1 bei ALSA ein „alsa“, z.B. AUDIO _DEV=alsa:plughw:1

AUDIO _CHANNEL

Der Kanal des Ausgangs der genutzt wird. Im Gegensatz zum herkömmlichen Echolink, können bei SvxLink die Stereo-Kanäle getrennt verwendet werden, z.B. AUDIO _CHANNEL=1

PTT _PORT

Der Port, der für die RX/TX-Umschaltung genutzt wird, z.B. PTT _PORT=/dev/ttyS0

PTT _PIN

Der Pin des PTT-Ports, der den Pegel für die RX/TX-Umschaltung liefert, z.B.

PTT _PIN=DTRRTS -> bei TX sind DTR und RTS auf high

PTT _PIN=DTR -> bei TX ist nur DTR high

PTT _PIN=!DTR -> Invertierung, falls der angeschlossene Sender einen low-Pegel benötigt um auf Sendung zu gehen

PTT _PIN=NONE -> keine PTT erforderlich, z.B. bei der MicSpkrLogic

TIMEOUT

Maximale Sendezeitbegrenzung in Sekunden, z.B. TIMEOUT=3000. Der TX schaltet nach einer Sendezeit von 3000 Sekunden unabhängig von anderen Kriterien ab.

TX _DELAY

Auftastverzögerung des Senders in ms, z.B. TX _DELAY=0

CTCSS _FQ

Der auf dem TX-Träger aufmodulierte CTCSS-Ton, z.B. CTCSS _FQ=136.5

CTCSS _LEVEL

Der Audiopegel des CTCSS-Tones, z.B. CTCSS _LEVEL=9

PREEMPHASIS

Höhenanhebung des Audiosignales um x dB

Links

Im Link-Abschnitt wird konfiguriert, welche Logiken auf Anforderung zusammenschaltet werden. Im Moment wird das automatisierte Linken der Logiken noch nicht unterstützt, so dass dies per Hand durch ein Kommando passieren muß. Folgende Variablen können konfiguriert werden: CONNECT _LOGICS, NAME, COMMAND, TIMEOUT

CONNECT _LOGICS enthält eine kommaseparierte Liste der zu linkenden Logiken, z.B. CONNECT _LOGICS=RepeaterLogic,SimplexLogic,MicSpkrLogic.

NAME ist eine freie Definition, wird beim Linken ausgegeben, z.B. NAME=R2Link.

TIMEOUT wird momentan nicht genutzt.

COMMAND Das Kommando um die Logiken miteinander zu verbinden oder zu trennen, z.B. COMMAND=94 mit dem Kommando 941 # werden die konfigurierten Logiken verbunden und mit 940 # wieder getrennt

Macros

Auf diesen Macro-Abschnitt wird von den einzelnen Logiken heraus mit Hilfe der Variablen MACROS= aus verwiesen. Prinzipiell kann für jeder Logik ein eigener Macro-Bereich zugeordnet werden, so dass jede Logik verschiedene Macros beinhalten könnte. Weitere Informationen zur Konfiguration von Makros sind unter http://svxlink.de/?page_id=458 zu finden.

MultiTx

Ein MultiTx-Bereich wird durch die Variablendefinition TYPE=Multi eingerichtet. In diesem Bereich muss eine Zuweisung von mindestens einem Sender erfolgen: TRANSMITTERS=Tx1,Tx2,... Die hier definierten Sender werden im Sendefall praktisch gleichzeitig mit dem Audiostream versorgt bzw. angesteuert.

Voter

Ein RX-Voter wird durch die Variablendefinition TYPE=Voter eingerichtet. Innerhalb dieses Bereiches muss die Zuweisung mindestens eines Rx als Parameter für RECEIVERS erfolgen:

RECEIVERS=Rx1,NetRx1,... Zwischen den hier konfigurierten Empfängern erfolgt ein Voting, der Rx mit dem jeweils besten Signal wird "durchgeschaltet", wobei eine Verzögerung um die im VOTING _DELAY konfigurierte Zeit [ms] erfolgt.

LocationInfo

Auf diesen Abschnitt wird unter http://svxlink.de/?page_id=325 näher eingegangen.

Routereinstellungen (2011-01-23 13:52)

Für die Einstellungen Ihres DSL-Routers gelten grundsätzlich die selben Parameter wie beim Betrieb der herkömmlichen Echolink-Software.

Für den Port 5200/tcp muß der Aufbau einer ausgehenden Verbindung erlaubt sein, die Ports 5199/udp und 5198/udp benötigen einen Portforward auf dem Router zu dem Rechner, auf dem SvxDLink läuft.

Soll von außen der Zugriff per Qtel auf Ihren SvxDLink-Server möglich sein, so muss ebenfalls für den Port 5200/tcp ein Portforward eingestellt werden.

Achten Sie auch auf eventuelle Firewalls, die auf Ihrem Rechner laufen, bei OpenSuSE ist z.B. standardmäßig ein Firewall aktiviert, der je nach Einstellung und Konfiguration die Funktionsweise von SvxDLink einschränkt.

Tcl-Routinen (2011-01-27 16:06)

Die TCL-Skripte im Verzeichnis /usr/share/svxlink/events.d/ stellen sozusagen die Präsentationsebene von SvxLink dar. Hier wird festgelegt auf welche Art und Weise beim Eintreten von bestimmten Ereignissen die Ausgaben erfolgen, ob z.B. vor oder nach einer Ansage noch ein Ton gesendet wird, wenn ja in welcher Höhe und Länge...usw.

Ein paar Beispiele

Ereignis

Funktion im Tcl-Skript

Öffnen oder Schliessen der Rauschsperr

Logic.tcl -> squelch _open { }

Schließen der Rauschsperr und Ablauf des RGR-Timers

Logic.tcl -> send _rgr _sound { }

Empfang des DTMF-digits "*"

Logic.tcl -> manual _identification { }

Sender ein oder aus

Logic.tcl -> transmit { }

Beispiel für Änderungen am Roger-Piep

Der für den Roger-Piep zuständige Abschnitt in der /usr/share/svxlink/events.d/Logic.tcl ist die Prozedur send _rgr _sound { }:

```
#
# Executed when the squelch just have closed and the RGR_SOUND_DELAY timer has
# expired.
#
proc send _rgr _sound { } {
    variable sql _rx _id;
    playTone 440 80 50;
    playSilence 50;

    for {set i 0} { $i < $sql _rx _id } {incr i 1} {
        playTone 880 50 50;
        playSilence 50;
    }
    playSilence 100;
}
```

playTone 440 80 50; erzeugt einen Ton von 440 Hertz mit eine Amplitude von 80 und einer Länge von 50 Millisekunden. playSilence 50; liefert eine Pause von 50 ms. Der darauf folgende Abschnitt

80


```

for {set i 0} { $i < $sql_rx_id } {incr i 1} {
playTone 880 50 50;
playSilence 50;
}

```

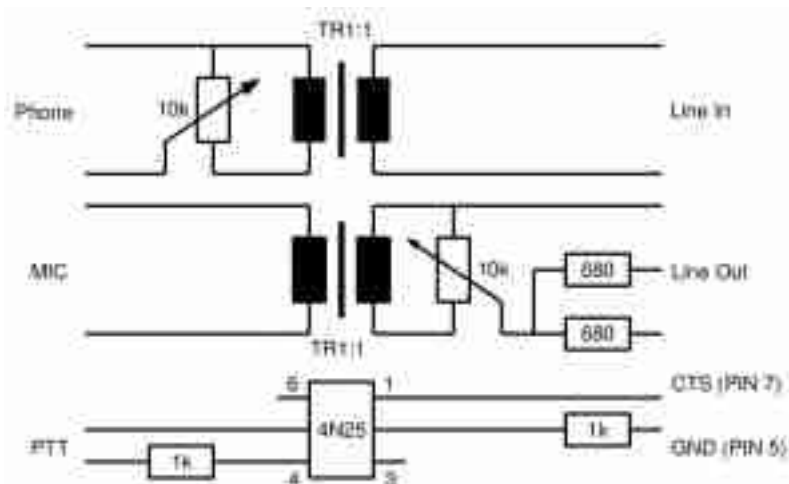
ist nur interessant, wenn Sie mehrere remote-trx betreiben. Anhand der Anzahl der nachfolgenden Töne läßt sich ermitteln, über welchen Rx die vorhergehende Sendung aufgenommen wurde.

In der Logic.tcl gibt es ca. 38 Routinen, welche bei unterschiedlichsten Ereignissen ausgeführt werden, so dass sich sicher für jeden nur erdenklichen Fall eine Konfiguration für ein bestimmtes Ereignis realisieren läßt. Achte Sie aber bei Anpassungen darauf, dass diese in userspezifischen tcl-Dateien passieren und NICHT in den mitgelieferten tcl-Routinen. Weitere Informationen finden Sie unter locale-Konzept": http://svxlink.de/?page_id=903

Interfaces (2011-01-28 08:44)

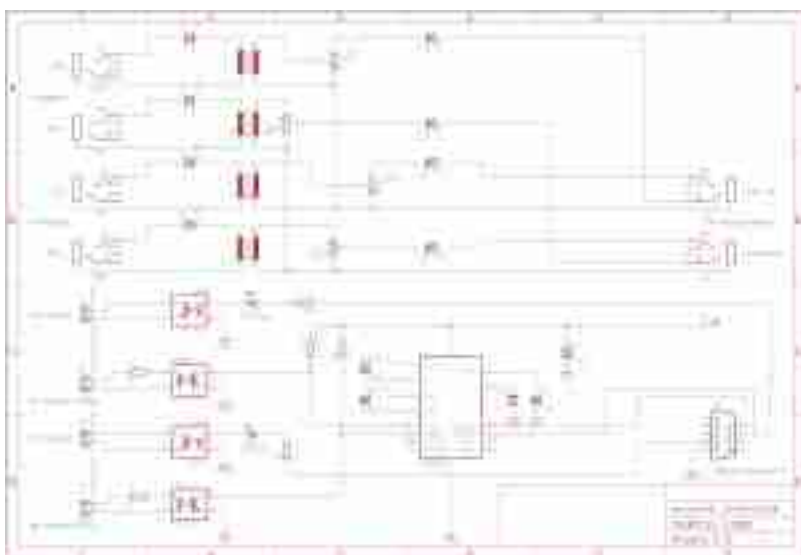
Für SvxLink können die selben Interfaces genutzt werden wie für das original Echolink-Programm für Windows. Wenn man aber selbst basteln möchte, so kann man auch die beiden Kanäle trennen und einen Adapter bauen, mit dem 2 Funkgeräte an eine Soundkarte angeschlossen werden können.

Nachfolgend ein kleiner Schaltplan von Martin (DF1AMB):



Hardwareadapter für SvxLink

SvxLink kann mit einer Stereo-Soundkarte und einem RS232-Port zwei Funkgeräte unabhängig voneinander ansteuern. Nachfolgend ein einfaches Interface welches auch die Möglichkeit bietet, ein externes Rauschsperrkriterium (wenn Ihr RX eine entsprechende Information liefert) an SvxLink zu übermitteln.



Einfaches SvxLink-Interface zur unabhängigen Ansteuerung von zwei Funkgeräten

Muss in jedem Fall ein Interface mit galvanischer Trennung verwendet werden?

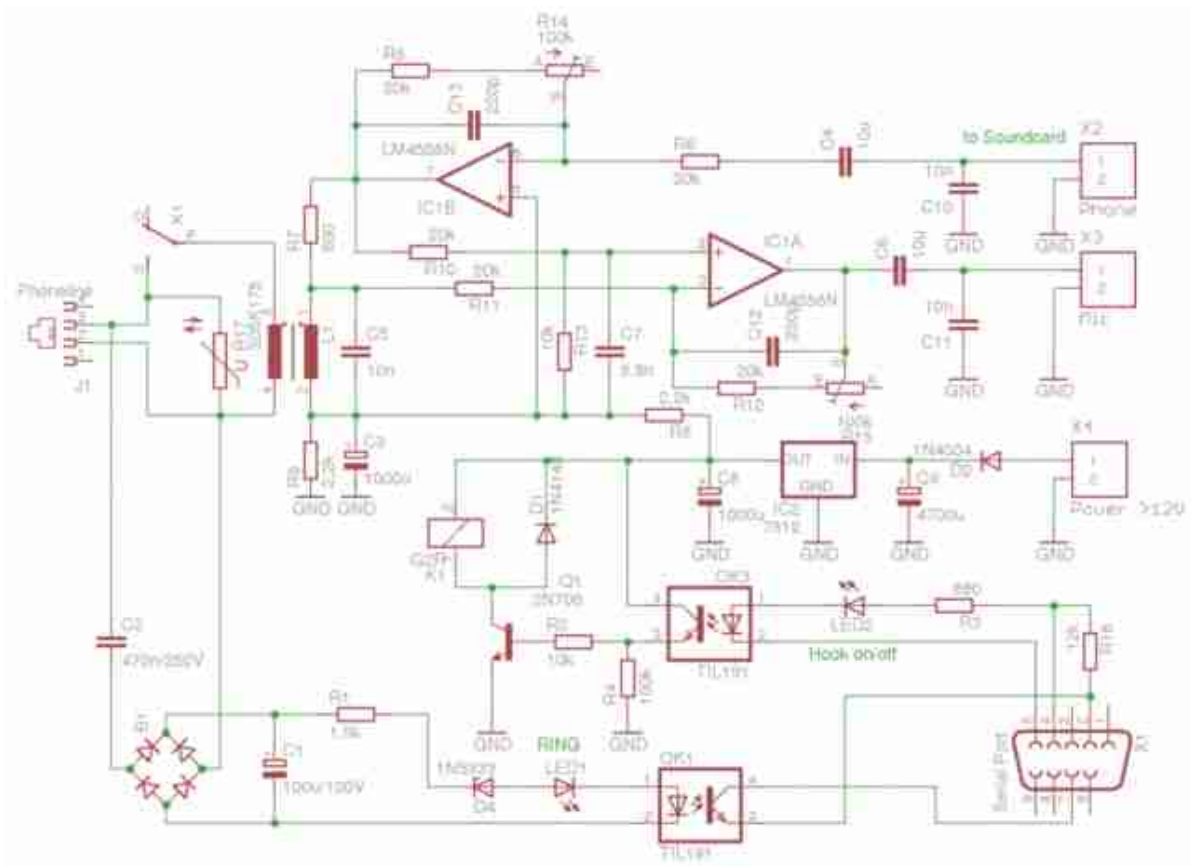
Nein, wenn Sie z.B. das Funkgerät an die selbe Spannungsversorgung anschließen mit der der PC betrieben wird, hat die galvanische Trennung ohnehin keine Wirkung. Wir haben über lange Zeit einen Link ohne Trennung betrieben, bei dem das Funkgerät an die 12V des PC's angeschlossen war und hatten keinerlei Probleme.

Ist für SvxLink ein Hardware-DTMF-Dekoder notwendig?

Nein, SvxLink besitzt folgende Softwaredekoder: DTMF, Rufton, CTCSS, SEL5 (ZVEI, EEA,EIA,CCIR,...).

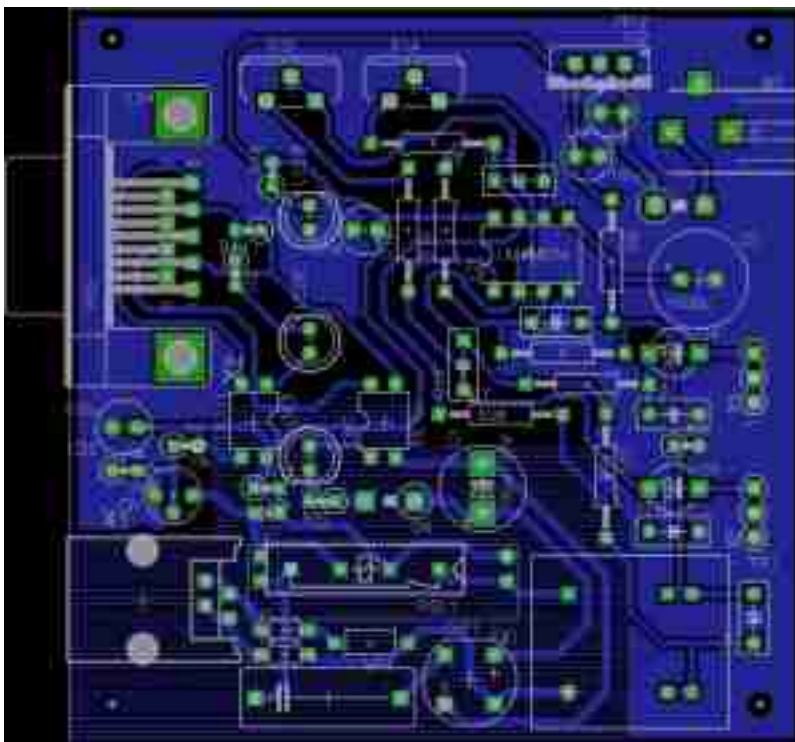
Cradle (2011-01-29 12:45)

Um den PC mit einer analogen Telefonleitung zu verbinden gibt es verschiedene Möglichkeiten. Eine davon ist eine sehr simple Telefonweiche (Cradle). Im ersten Bild ist der Schaltplan dargestellt.



Schaltplan einer einfachen Telefonweiche

Und hier das Layout für das Board.



Layout für die Cradle

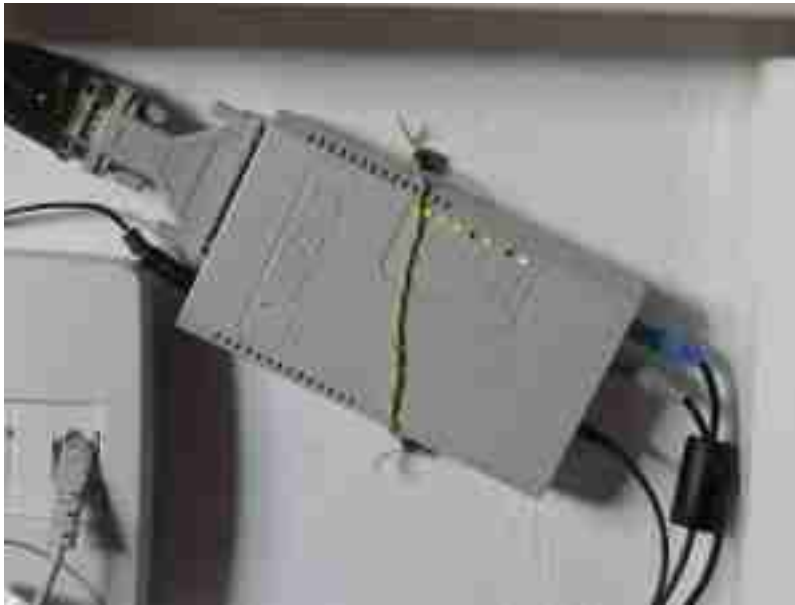
Bild des Testaufbaus.



Testaufbau der Cradle ... Tja, ich bin eben ein Ossi ;-)

Analoges Voicemodem (2011-01-29 14:43)

Die Ansteuerung des analogen Voicemodems wird über eine normale RS232 vorgenommen. Audioein- und Ausgang können direkt mit Mic- bzw. Lineout der Soundkarte verbunden werden. Es funktioniert ausdrücklich nur mit einem analogen Voicemodem, dieses muss die AT-Befehle: AT+VLS=..., AT+CLS=... bzw. AT #VLS=..., AT #CLS= unterstützen. Leider kocht auch hier jeder Hersteller sein eigenes Süppchen und das Verhalten der Modems bei ein und den selben Befehlen ist oft sehr unterschiedlich.



Creatix VF14400

Getestet haben wir mit einem uralten Creatix VF14400. Das eingebaute Elektret-Mikrofon wurde durch eine Buchse ersetzt, damit man es mit dem Lautsprecherausgang der Soundkarte verbinden kann. Weitere Information bzgl. Installation und Konfiguration folgen in Kürze.

GSM-Interface (2011-01-30 12:49)

Für Mobiltelefone der Siemens M/S-35-Serie ist eine kleine Zusatzschaltung erforderlich. Ein Schaltplan ist hier zu finden. Das größte Problem ist eigentlich einen passenden Stecker zu bekommen, bei dem alle notwendigen Kontakte verfügbar sind. Auf ebay gibt es leider nur Adapter, die entweder die Audioleitungen herausführen ODER eine Datenkommunikation mit/ohne Akkumulatorladefunktion bereitstellen....leider nicht alles.



GSM-Interface für Mobiltelefone der Siemens-Serie (C35, M35)

Steckerbelegungen von Mobiltelefonen finden Sie auf Nobbi.com

Etwas einfacher gestaltet sich der Anschluß eines Sony-Ericsson P910i, hier ist kein Interface mit MAX3232 erforderlich.



GSM-Telefon (hier Sony Ericsson P910i) zum Anschluß an SvxLink

Die billigen Soundkarten für teilweise 5€ eignen sich recht gut dafür. Allerdings besitzen diese oft keinen Stereo-Mikrofoneingang.

Momentan befindet sich alles im absoluten Bastelstadium. Getestet wurde mit einem Sony Ericsson P910i, es müssten allerdings auch andere Geräte wie z.B. K610, 750 funktionieren. Eine Unterstützung für Siemens C35 etc. ist in Arbeit.

Momentan muss das Gerät auf den AT-Befehl `ÄT &FE0*ECAM=1` mit OK reagieren. Sonst funktioniert es nicht. Für die Unterstützung weiterer Geräte benötigen wir eine AT-Befehlsreferenz zum interessierenden Gerät. Wer etwas entsprechendes besitzt und es zur Verfügung stellen möchte, bitte melden.

Gesucht werden momentan AT-Befehlssätze zu:

Blackberry 9000

bold 9000

1.2 Februar

Forum (2011-02-02 18:36)

Zum Forum geht es hier lang

1.3 Oktober

Internetstreaming vor Linux Kernel Version 2.6.37 (2011-10-17 20:42)

Um den Funkverkehr einer breiten Masse zugänglich zu machen bietet sich die Einrichtung eines Streaming-servers an.

Im Folgenden wird die Einrichtung beschrieben. Diese ist momentan aber noch nicht optimal, an einer Verbesserung wird gearbeitet. Das Problem ist, dass Ices2 offensichtlich nicht von einem Playback-Device streamen kann. Aus diesem Grund ist momentan bei dieser Version noch eine zusätzliche Soundkarte erforderlich. Es ist aber mein Ziel, darauf zukünftig verzichten zu können.

Voraussetzungen:

Icecast2 als Streamingserver

Ices2 als Streaming client

eine einfache USB-Soundkarte, bei der der Lautsprecherausgang und Mikrofoneingang miteinander verbunden werden.

Ich nutze hierfür eine billige USB-Soundkarte für 4,95 € bei einem bekannten virtuellen Auktionshaus.

Zunächst müssen Icecast2 und Ices2 installiert und nach Wunsch konfiguriert werden, im Internet gibt es hierfür mehr als genügend Konfigurationsbeispiele.

Der ices2 wird auf dem selben PC installiert, auf dem auch SvxLink läuft. Hier gibt es folgendes zu beachten:

Ices2 kann nicht auf das selbe Device zugreifen, welches von SvxLink als Sounddevice genutzt wird, es kommt hierbei zu einer resource busy Meldung.

Aus diesem Grund splittet man den Audiopfad mit Hilfe der /etc/asound.conf auf:

```
pcm.convert {
type plug
slave { pcm "hw:1,0" }
}
```

```
pcm.dmix1 {
type copy
slave { pcm convert }
}
```

```
pcm.dmix2 {
type copy
slave { pcm dmix1 }
}
```

Ices2 grabbt nun von z.B. dmix2 und SvxLink liefert die Daten an dmix1. Nachfolgend Ausschnitte für beide Applikationen:

/etc/ices2.xml:

```
...
< stream >
< metadata >
< name >DB0WSF< /name>
< genre >Hamradio Network< /genre >
< description >Livestream of DB0WSF-R / Pettstaet / JO51wg< /description >
< url >http://svxlink.de< /url >
< /metadata >
```

```

< input >
< module >alsa< /module >
< param name=rate>16000< /param>
< param name=channels>1< /param>
< param name="device">plug:dmix2< /param>
< param name="buffer-time">500< /param>
< param name=metadata>1< /param>
< param name=metadatafilename>/tmp/info.txt< /param>
< /input>
...

```

SvxLink wird als Multi-Tx konfiguriert (/etc/svxlink/svxlink.conf):

```

[RepeaterLogic]
TYPE=Repeater
RX=QuantarRx
TX=MultiTx
MODULES=ModuleEchoLink,ModuleMetarInfo,ModulePropagationMonitor,Modu
ModuleTrafficInfo
....

```

leSelCallEnc,-

```

[MultiTx]
TYPE=Multi
TRANSMITTERS=RepeaterTx,StreamingTx

```

```

[RepeaterTx]
TYPE=Local
AUDIO_DEV=alsa:hw0
AUDIO_CHANNEL=0
PTT_PORT=/dev/ttyS0
PTT_PIN=DTRRTS
TX_DELAY=0
PREEMPHASIS=0
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18

```

```

[StreamingTx]
TYPE=Local
AUDIO_DEV=alsa:dmix1
AUDIO_CHANNEL=0
PTT_PORT=NONE
PTT_PIN=DTRRTS
TX_DELAY=0
PREEMPHASIS=0
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18

```

Jetzt alle Applikationen starten und die Funktion prüfen.
Viel Erfolg wünscht Adi, DL1HRC

2. 2012

2.1 Januar

SvxLink für 16k-Sounds selber kompilieren (obsolete) (2012-01-09 18:39)

ACHTUNG: Aktuelle SvxLink-Trunks und -Releases sind bereits für 16k-Sounds konfiguriert! Sie müssen nichts mehr umkonfigurieren!

SvxLink wurde bis 2013 standardmäßig als fertiges Paket ausgeliefert, welches nur 8k-(wav|raw)-Dateien unterstützt. Diese klingen sagen wir mal „etwas digital“. Eine wesentlich bessere Qualität kann man von 16k-Sounds erwarten. Hier ist noch etwas Handarbeit gefragt, man muß die Quellen noch selber kompilieren.

1) Laden Sie sich die Quellen per svn herunter, entweder das offizielle Release:

```
svn co https://svxlink.svn.sourceforge.net/svnroot/svxlink svxlink-release
```

oder für die mutigen SvxLink-Nutzer die Trunk-Version

```
svn co https://svxlink.svn.sf.net/svnroot/svxlink/trunk svxlink-trunk
```

2) wechseln Sie in das entsprechende src-Verzeichnis und ändern Sie in der Datei makefile.cfg den Eintrag wie folgt ab:

```
CXXFLAGS += -DINTERNAL_SAMPLE_RATE=8000
```

in

```
CXXFLAGS += -DINTERNAL_SAMPLE_RATE=16000
```

3) führen Sie die Schritte make sowie make install aus.

Ab jetzt können Sie die 16k-Sounds nutzen.

Locale-Konzept (2012-01-18 13:09)

Da es immer wieder zu Fragen und Problemen führt, wird auf dieser Seite das SvxLink-locale-Konzept etwas genauer beschrieben. Der Hintergrund des locale-Konzeptes ist einerseits den Usern eine einfache Anpassung an die

in der jeweiligen Landessprache üblichen Ausdrucksweise zu ermöglichen. Andererseits sollen diese userspezifischen Änderungen nicht bei Systemupdates jedes Mal neu überschrieben werden sondern natürlich auch erhalten bleiben. Es ist deshalb ungünstig die mitinstallierten tcl-Dateien nach den eigenen Bedürfnissen zu verändern, da diese Änderungen nach jedem Update wieder verloren gehen würden. Auch ein Sichern der tcl-Dateien und späteres Rücksichern ist ungünstig, da die neuen tcl-Dateien eventuell neue oder veränderte Prozeduren enthalten könnten, die dann durch die alten Einstellungen überschrieben würden.

Beim Start von SvxLink wird normalerweise vom Eventhandler zunächst die Datei events.tcl geladen, welche die anderen modul- und logikspezifischen tcl-Dateien nachlädt. Das passiert in der folgenden Reihenfolge (Stand 1/2012 für deutsche Umgebung):

- 1) /usr/share/svxlink/events.d/
- 2) /usr/share/svxlink/events.d/local/
- 3) /usr/share/svxlink/sounds/de_DE/events.d/
- 4) /usr/share/svxlink/sounds/de_DE/events.d/local/
- 5) /usr/share/svxlink/modules.d/

Wenn man also eine tcl-Systemfunktion mit einer eigenen Funktion überschreiben möchte um z.B. die Ansage der Uhrzeit an die deutsche Sprechweise anzupassen, so erstellt man jeweils in den zugehörigen local-Verzeichnissen eine beliebige tcl-Datei, definiert darin zunächst den Namespace und innerhalb dieses die eigene Funktion. Da diese eigene Datei im local-Verzeichnis NACH den System-tcl's gelesen wird, werden die System-tcl-Funktionen durch die selbstdefinierten Funktionen überschrieben.

Hier mal Beispiel für die Änderung des roger beeps, dieser wird in der Datei Logic.tcl generiert. Im Beispiel soll die Tonhöhe von 440Hz auf 800Hz verändert werden:

- 1) Erstellen Sie eine Datei beep.tcl mit folgendem Inhalt:

```
namespace eval Logic {
#
# Executed when the squelch just have closed and the RGR_SOUND_DELAY timer has
# expired.
#
proc send_rgr_sound { } {
playTone 800 80 50;
playSilence 50;
}
}
```

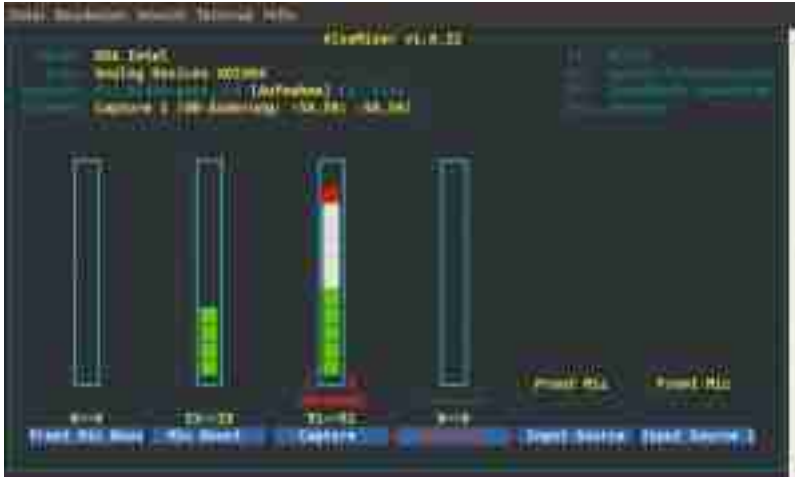
- 2) Speichern Sie diese Datei in dem Verzeichnis /usr/share/svxlink/sounds/de_DE/events.d/local/ ab. Starten Sie SvxLink neu, damit die neuen Einstellungen geladen werden können.

Audioeinstellungen (2012-01-23 14:38)

Zur Laufzeit müssen die Audiopegel der Soundkarte an die örtlichen Gegebenheiten angepaßt werden. Hier gibt es viele Varianten/Programme, stellvertretend soll die Variante mit dem alsamixer erläutert werden.

1) Loggen Sie sich auf dem SvxBLink-Rechner lokal oder per ssh ein und starten sie SvxBLink

2) Starten sie den alsamixer, ohne Option wird die erste Soundkarte angesprochen, sollte SvxBLink die zweite, dritte, n-te Soundkarte nutzen, so starten Sie den alsamixer wie folgt: alsamixer -cX wobei X die Nummer der Soundkarte angibt, die SvxBLink nutzt. Die Zählung beginnt bei 0 für die Erste, 1 für die Zweite, usw. .. N für die N+1.te Soundkarte.



Alsamixer Aufnahmekanäle

Die Bedienung erfolgt mit den Pfeiltasten, sowie den Funktionstasten [F1..F6]



Alsamixer Wiedergabekanäle

3) Verändern Sie die Einstellungen entsprechend, deaktivieren Sie alle nicht benötigten Kanäle (Stummschalten per Taste [M]). I.A. benötigt man nur die Kanäle Master, PCM, Capture und e.v. Mic Boost

4) Die Regler sollten nicht auf 100 % stehen

5) Verlassen Sie den Alsamixer und speichern Sie die Einstellungen mit alsactl store X" wobei X die Nummer der Soundkarte angibt (siehe oben)

2.2 Februar

Unterschiedliche Roger-Beep's (2012-02-09 21:33)

Für die Identifizierung einer EchoLink-Verbindung und HF-Verbindung kann man unterschiedliche Roger-Beeps ausgeben. Um das zu realisieren muss man die Prozeduren Logic::squelch _open und Logic::send _rgr _sound überschreiben. Hier die Kurzanleitung für eine deutsche Sprach-Umgebung:

1) eine Datei z.B. roger-beep.tcl erstellen mit folgendem Inhalt:

```
#
# zwei verschiedene Roger-beeps:
# hoher Ton (540Hz) - EchoLink
# tieferer Ton (440Hz) - HF-Verbindung
#

namespace eval Logic {

variable is _rf 0;

proc squelch _open {rx _id is _open } {
variable sql _rx _id;
variable is _rf;
set sql _rx _id $rx _id;
if {! $is _open } {
playSilence 100;
playTone 440 70 50; # Ton mit 440Hertz ausgeben
set is _rf 1;
}
}

proc send _rgr _sound { } {
variable is _rf;
if {! $is _rf } {
playTone 540 70 50; # Ton mit 540Hertz ausgeben
}
set is _rf 0;
}
}
# end of namespace
```

2) diese Datei in das Verzeichnis /usr/share/svxlink/events.d/local/ speichern

3) SvxLink neu starten

Trennung EchoLink-Verbindung verhindern (2012-02-14 12:46)

Es gibt leider eine ganze Reihe von Leuten, deren Horizont arg beschränkt ist, die dem QSO-Thema nicht folgen können, den OMs durch die Trennung der EchoLink-Verbindung permanent das QSO zerstören oder für Rundsprünge zusammenschaltete Relaisstationen trennen. Um zu verhindern, dass eine EchoLink-Verbindung ständig durch diese Störer getrennt wird, kann man einen der folgenden Wege gehen.

1) Disconnect wird durch "geheimes Kommando" verhindert

Erzeugen Sie eine Datei z.B. no-disconnect.tcl mit dem folgenden Inhalt:

```
# das Überschreiben der Prozedur mit folgendem Inhalt verhindert das
# einfache Trennen einer EchoLink-Verbindung durch "#"
# Ersetzen Sie die Werte für
# "XX" mit dem Kommando für die Aktivierung der Disconnect-Sperre
# "YY" mit dem Kommando für die Deaktivierung der Disconnect-Sperre
# für den Zeitraum der Aktivierung nimmt svxlink kein "leeres"
DTMF-Kommando
# entgegen (z.B. Rundspruch)

namespace eval Logic {

    # variable for Logic.tcl (muss im namespace Logic definiert werden,
    Nachtrag, 5.3.13, M. H.)
    variable ignore_dtmf 0;
    proc dtmf_cmd_received {cmd} {
        puts "DTMF command received: $cmd";
        variable ignore_dtmf;
        if {$cmd == "XX"} {
            set ignore_dtmf 1;
            return 1;
        }
        if {$ignore_dtmf} {
            if {$cmd == "YY"} {
                set ignore_dtmf 0;
                return 1;
            } elseif {$cmd == ""} {
                return 1;
            }
        }
        return 0;
    }
}
# end of namespace Logic
```

2) Disconnect ist erst nach xx Sekunden möglich und nur wenn von der EchoLink-Station kein Audiostream mehr zu empfangen ist, benötigt eine Erweiterung im Modul EchoLink/ModuleEchoLink::intsReceiving():
Erzeugen Sie eine Datei z.B. no-disconnect.tcl mit dem folgenden Inhalt:

```

namespace eval Logic {
  proc dtmf_cmd_received {cmd} {
    global last_el_stream;
    global is_el_rx;
    variable now;
    variable ignore_dtmf;
    variable diff;
    set cs $EchoLink::num_connected_stations;
    set now [clock seconds];

    # Trennung erst nach 35 Sekunden oder durch geheimes Kommando
    "78#"
    set diff [expr {$last_el_stream + 35 - $now}];
    if {$cs > 0} {
      if {(($diff < 0) && !$is_el_rx) || $cmd == "78"} {
        puts "disconnect OK";
        return 0;
      }
      return 1;
    } else {
      return 0;
    }
  }
}
# end of namespace Logic

namespace eval EchoLink {
  proc is_receiving {isrec} {
    global last_el_stream;
    global is_el_rx;
    set is_el_rx $isrec;
    if {$isrec == 0} {
      set last_el_stream [clock seconds];
    }
    puts "is_receiving $isrec $is_el_rx $last_el_stream";
  }
}
# end of namespace EchoLink

```

2) Speichern Sie diese Datei(en) in ein entsprechendes locale-Verzeichnis (siehe Locale-Konzept) und starten Sie SvxLink neu

2.3 März

Projekte (2012-03-10 20:37)

SvxLink auf Pandaboard (2012-03-10 20:40)

Liebe VoIP-Gemeinde,

ich möchte euch heute mein Projekt vorstellen und dazu animieren, CO2 einzusparen und Echolink auf einem Rechner zum Laufen zu bringen, der zwischen 3-5 Watt Leistung im Dauerbetrieb aufnimmt.

Voraussetzungen für das Projekt:

- Einige Grundkenntnisse der englischen Sprache, das gilt sowohl für Linux/Ubuntu, für svxlink, als auch für Informationen rund um das Pandaboard.
- Grundlagen mit dem Umgang von Ubuntu (Installation, Bedienung über das Terminal, Umgang mit dem Ubuntu-Wiki).

Als mehrjähriger Ubuntu-Anwender und enthusiastischer Echolinkbenutzer bekam ich Lust auf ein "Stromsparprojekt". Ich wollte meinen Linux svxlink-Server mit möglichst geringer Stromaufnahme laufen lassen. Dafür kommen Prozessoren und Mainboards in Frage die nicht mit einer x86-Architektur laufen, sondern sog. ARM - Prozessoren z.B. von TI (Texas Instruments). Nach mehreren Tagen Recherche im Internet kamen für mich dann zwei Boards in Frage, das BeagleBoard oder das Pandaboard (pandaboard.org). Ausschlaggebend für die Entscheidung eines dieser beiden Boards zu nehmen war, dass es eine möglichst große Community und damit einen möglichst grossen Support übers Internet gibt, mit dem es auch einen Anfänger gelingt nach Anleitung zu compilieren und zu installieren.

Entschieden habe ich mich letztendlich für das normale Pandaboard, also nicht die neue ES-Variante. Pandaboard, weil es WLAN und BT beinhaltet (im Gegensatz zum Beagleboard) und die non-ES Version, weil es in dieser Form schon seit geraumer am Markt ist und die Kinderkrankheiten weg sind/sein sollten. Das Pandaboard ES kommt jetzt erst frisch auf den Markt. Es hat zwar einen dual-core Prozessor, aber es ist zu erwarten, dass erst in den kommenden Rev.'s die Anlaufschwierigkeiten beseitigt sein könnten. Ob es mehr Strom aufnimmt weiß ich nicht.

Meine Einkaufsliste:

- Pandaboard (Lieferzeit 14 Tage, Lieferanten in DL und OE)
- Netzteil 5V/4A
- Adapterkabel RS232 auf USB
- Gehäuse (es gibt in USA jemanden, der ein Acrylgehäuse vertreibt welches schick aussieht)
- 8 GB SDHC-Karte (mind. 4 GB Class 6)
- optional: Adapter DVI auf HDMI, falls ein HDMI-Monitor betrieben werden soll.

Benötigt man nicht unbedingt zum Start, aber wer weiß was noch kommt:

- WLAN Zubehör: HF-Kabel vom WLAN-Chip zur Antenne, eine 2,4 GHz WLAN-Antenne

Vorbereitungen:

- Download der Ubuntu-Distribution für arm+omap (<https://wiki.ubuntu.com/ARM/OMAP>). Man hat die Wahl zwischen der Server- und der Desktopvariante. Ich habe die Desktopvariante ubuntu-11.10-preinstalled-desktop-armel+omap4.img.gz gewählt, da sie u.a. eine mir bekannte Benutzeroberfläche mitbringt. Achtung - der Prozessortyp ist dabei wichtig. Unterschieden werden u.a. 3er und 4er OMAP-Typen.

- Es gibt auch eine unter <https://wiki.ubuntu.com/ARM/OmapNetbook> beschriebene Anleitung für die Vorbereitung auf einem Linuxrechner. Durch die Verwandtschaft zum Beagleboard sind die beiden Prozeduren auch unter <http://code.google.com/p/beagleboard/wiki/BeagleBoardDiagnosticsNext> gut beschrieben. Ich bin den Weg über den Windowsrechner gegangen, in meinem Falle mit Win-XP und dem Programm Win32DiskImager.exe, zu beziehen unter <https://launchpad.net/win32-image-writer/+download>. So habe ich die Ubuntu-Distro auf die SD Karte gebracht



Fertiges Gerät

Board starten

Die vorbereitete SD-Karte nun ins Board legen, alle Kabel (evtl. USB-Hub, Bildschirm, Tastatur, Maus, WLAN-Antenne) verbinden und den Strom einschalten. Bitte die beiden unterschiedlichen Bildschirmports beachten, der eine ist für HDMI der andere für DVI. Etwas modernere Screens sollten – dann aber über einen Adapter – per HDMI zu betreiben sein. Es erfolgt nun die Installation von Ubuntu, dazu werden einige Eingaben wie z.B. die Zeitzone benötigt. Nach erfolgreicher Installation hat man zunächst einen großen Rechner im Sinne eines funktionsfähigen und mit vielen Programmen ausgestatteten modernen PC. Das ist die Voraussetzung um mit dem nächsten Schritt weiter zu machen.

Installation von svxlink

Wenn man dieser Anleitung <http://sourceforge.net/apps/trac/svxlink/wiki/InstallSrcUbuntu> exakt folgt kann eigentlich nichts schief gehen.

Per `<copy>+<paste>` einfach die Befehle aus den grau hinterlegten Fenstern in ein Terminal unter Ubuntu nacheinander ausführen. Wenn alles nach einigen Minuten glatt gelaufen ist, müssen nur noch die Parameterdateien, wie bei jeder Installation von svxlink, angepaßt werden.

Zeitfresser Soundkarte

Das was dann Zeit gekostet hat, war die Konfiguration

a) der Parameter für die externe Soundkarte in der svxlink.conf `AUDIO_DEV=alsa:plughw:0; 0` (null) bestimmt dabei die Soundkarte. Das Pandaboard hat 2, nämlich 0 = den analogen Ein- und Ausgang, 1=den HDMI-Ausgang. Steckt man eine externe USB-Soundkarte an bekommt sie die Nummer 2.

b) das Eigenleben unter Linux mit externer Hardware wie eben dieser USB-Soundkarte.

Hier gilt die Maxime try on error. Ein für mich wichtiges Tool ist die Anwendung pavucontrol mit der man parametrisieren kann als mit dem ALSA-Mixer oder den Audioneinstellungen.

Das Board läuft nun seit wenigen Tagen und ich muß jetzt noch einige Audiokinderkrankheiten beseitigen, wie z.B. ein plötzliches Verstummen nach mehrstündigem Lauf. Aber das kriegen wir auch noch hin ...

73, Frank <DL7ATA>
EL- # 41041

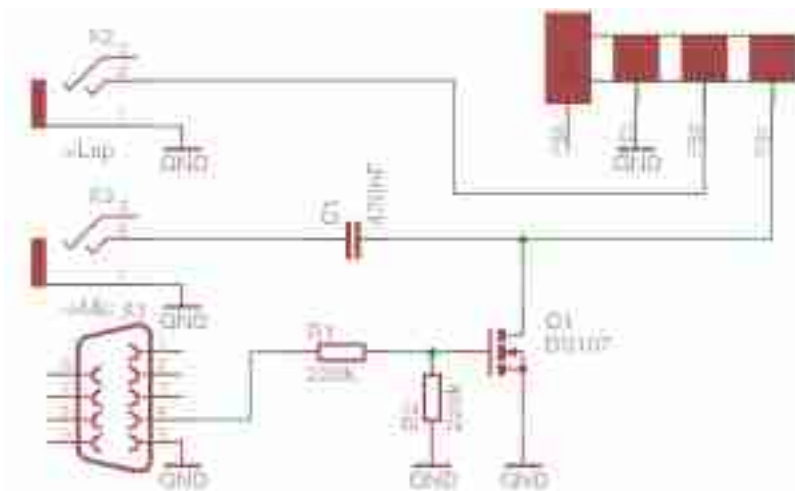
2.4 April

XTS3000 an SvxLink (2012-04-06 09:41)

Die XTS3000 von Motorola sind Handfunkgeräte, die sowohl analogen als auch digitalen (Apco25) Betrieb erlauben. Es liegt also nahe, einen einfachen Adapter zu bauen, der SvxLink mit der digitalen Apco25-Welt verbindet.

Einen Audio-Adapter für die XTS-Serie gibt es für wenig Geld bei einem einschlägigen Auktionshaus, mit etwas probieren habe ich auch die Anschlussbelegung herausgefunden. Man benötigt einen 3.5mm Klinkenstecker mit insgesamt 4(!) Kontakten wobei der Außenring (Masse) nicht genutzt wird. Belegung der Kontakte am 4poligen Klinkenstecker (Ohne Garantie!):

1. ext. Lautsprecher (über Kondensator) / PTT
2. Mikrofon
3. Masse
4. (Außenring) N.C.



XTS3000-SvxLink Adapter



XTS3000 Audioadapter

Leider kann SvxLink nicht per Apco25-Datenstrom in irgendeiner Weise gesteuert werden, dazu müsste man diesen dekodieren und auswerten (per RS232, z.B.). Es wird also lediglich eine Audioverbindung geschaltet. Die Konfiguration in SvxLink entspricht der einer MicSpkrLogic mit PTT. Als Soundkarte dient eine einfache USB-Soundkarte, die für knapp 5€ beim bereits erwähnten Auktionshaus erworben werden kann. Für diesen Anwendungsfall ist das vollkommen ausreichend. Nachfolgend eine Beispielkonfiguration: [ApcoLogic] TYPE=Simplex RX=Apco-Micro TX=Apco-Spekr MODULES=ModuleHelp DEFAULT _LANG=de _DE CALLSIGN=APCO EVENT_HANDLER=/usr/share/svxlink/events.tcl RGR_SOUND_DELAY=10 NO_REPEAT=1 SHORT_IDENT_INTERVAL=60 LONG_IDENT_INTERVAL=60 MUTE_RX_ON_TX=1 [Apco-Micro] TYPE=Local AUDIO_DEV=alsa:plughw:2 AUDIO_CHANNEL=0 SQL_DET=VOX SQL_START_DELAY=0 SQL_DELAY=0 SQL_HANGTIME=1700 SQL_TIMEOUT=600 VOX_FILTER_DEPTH=20 VOX_THRESH=1000 SERIAL_PORT=/dev/ttyUSB1 SERIAL_PIN=DCD:SET PEAK_METER=0 DTMF_DEC_TYPE=INTERNAL DTMF_MUTING=1 DTMF_HANGTIME=100 DTMF_SERIAL=NONE [Apco-Spekr] TYPE=Local AUDIO_DEV=alsa:plughw:2 AUDIO_CHANNEL=1 PTT_PORT=/dev/ttyUSB1 PTT_PIN=DTR TX_DELAY=0 TIMEOUT=1200



Anschluss eines XTS3000 an SvxLink (DBOWSF-R)

2.5 Juni

Parameter zur Laufzeit ändern (2012-06-06 13:41)

Es gibt eine relativ einfache Möglichkeit, TCL-Variablen zur Laufzeit von SvxLink zu ändern und diese innerhalb der TCL-Funktionen auszuwerten. Ein kleines Beispiel soll das Prinzip verdeutlichen.

Gefordert sei: Beim Connect einer Station via EchoLink soll an diese Station eine konfigurierbare Ansage erfolgen, die z.B. darüber informiert, dass z.B. DL1ABC nicht an der Station ist.

Erstellen Sie eine Datei (z.B. /tmp/user_params.tcl), die verschiedene Variablendefinitionen enthält:

```
set away 1;
set ansage abwesenheitsinformation;
```

Erstellen Sie eine Datei (/usr/share/svxlink/events.d/locale/el.tcl) nach dem locale-Prinzip.

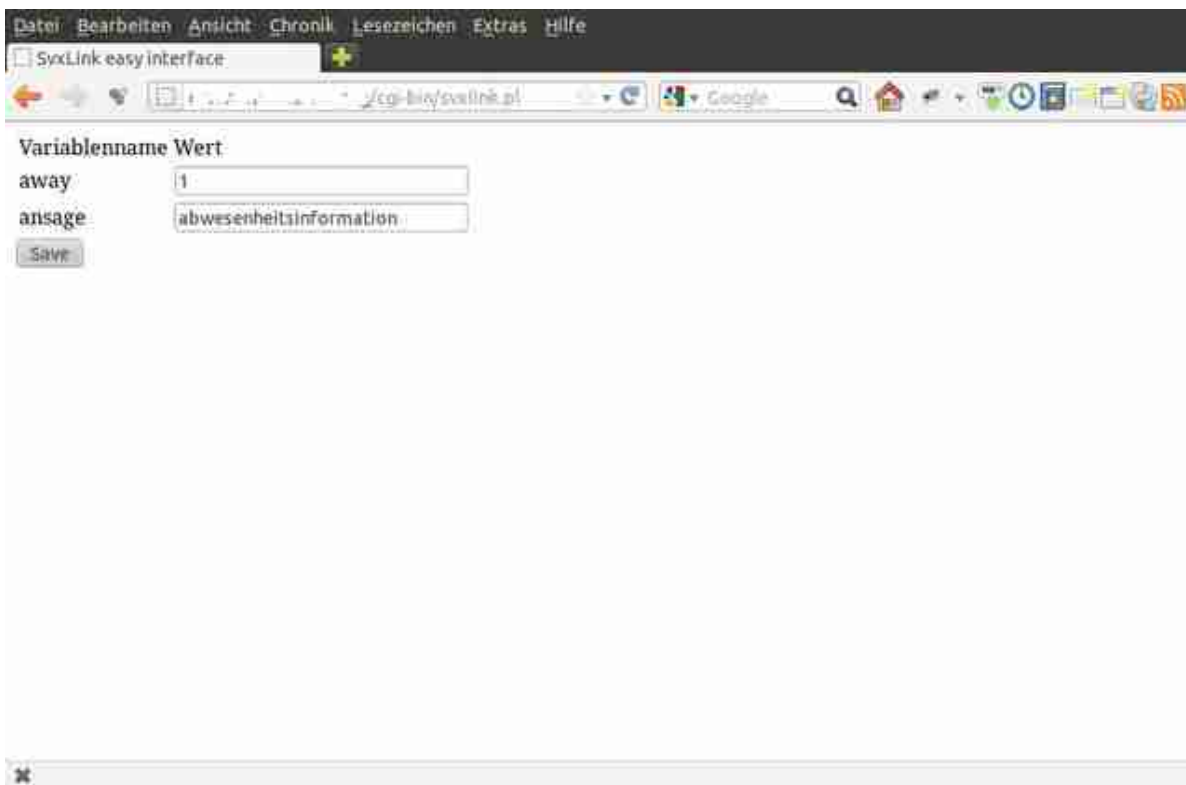
```
namespace eval EchoLink {
  proc remote_greeting {} {
    variable away;
    variable ansage;
```

```

source "/tmp/user_params.tcl";
playSilence 1500;
if {$away == 1} {
playMsg $ansage;
}
}
}
}

```

Um die Parameter zu ändern, kann man einfach das file /tmp/user _params.tcl bearbeiten oder mein kleines Webinterface installieren (apache2, perl, perl::CGI sind notwendig). Im Beispiel wird für die eingehende EchoLink-Verbindung die Audiodatei abwesenheitsinformation.wav abgespielt, wenn away=1 gesetzt ist.



Einfaches SvxLink Webinterface mit Perl::CGI

Auf diese Weise könnte man z.B. auch Wetterdaten aus einem File einlesen, welches periodisch mit den aktuellen Daten einer angeschlossenen Wetterstation gefüttert wird. Das File müsste dann in Form eines einfachen TCL-Files erstellt werden, d.h. in folgender Form (/tmp/wetterdaten.tcl):

```

set wind 100;
set temperatur 33;
set taupunkt 22;

```

...USW

Zur vollen Stunde könnte dieses File gesourced werden und neben der aktuellen Uhrzeit kann die aktuelle Temperatur mit ausgegeben werden, hier das Prinzip:

```

...
source "/tmp/wetterdaten.tcl";
playMsg "die_temperatur_betraegt";
playNumber $temperatur;
playMsg "grad_celsius";
...

```

Internetstreaming ab Linux Kernel Version 2.6.37 (2012-06-08 00:00)

Ab der Linux Kernel Version 2.6.37 kann auf eine 2. Soundkarte verzichtet werden, da ab dieser Kernel-Version das Modul `ßnd-aloop` was ein loopback-Device erzeugt bereits mitgeliefert wird.

Die folgende Beschreibung bezieht sich auf OpenSuSE12.1 sie sollte aber auch leicht auf andere Distributionen übertragbar sein.

Voraussetzungen:

einen Streaming-Server (Icecast2 oder Shoutcast)

einen Streaming-Client (Ices2 oder SC _Trans)

Linux Kernel ab Version 2.6.37 auf dem SvxLink-PC (z.B. OpenSuSE 12.1)

Zunächst müssen Icecast2 und Ices2 oder bei Verwendung von Shoutcast der Shoutcast-Server und SC _Trans installiert und nach Wunsch konfiguriert werden, im Internet gibt es hierfür mehr als genügend Installations- und Konfigurationsanleitungen.

Der Streaming-Server muss nicht zwingend auf dem selben PC installiert werden auf dem auch SvxLink läuft, hier bietet sich schon allein wegen der besseren Netzwerkanbindung ans Internet ein root-Server an, soweit dieser zur Verfügung steht.

Ices2 oder auch SC _Trans werden entweder auf dem PC auf dem SvxLink läuft oder wie bei mir, auf dem PC auf dem remotetrx läuft (um den SvxLink-PC von der CPU-Last zu befreien) installiert.

Dann müssen wir sicherstellen das die Module `ßnd-aloop` und `ßnd-pcm-oss`, dieses wird für die Verbindung zum Streaming-Client benötigt, beim starten des Systems auch geladen werden.

```
[/etc/sysconfig/kernel]:
```

```
MODULES_LOADED_ON_BOOT="ßnd-aloop snd-pcm-oss"
```

Nach dem Neustart des Systems kann nun mit dem Befehl `äplay -l` überprüft werden ob das Modul `ßnd-aloop` auch geladen wurde.

```
remote-trx1: # aplay -l
```

```
*** List of PLAYBACK Hardware Devices ****
```

```
card 0: rev40 [VIA 82C686A/B rev40], device 0: VIA 82C686A/B rev40 [VIA 82C686A/B rev40]
```

```
Subdevices: 0/1
```

```
Subdevice #0: subdevice #0
```

```
card 1: Loopback [Loopback], device 0: Loopback PCM [Loopback PCM]
```

```
Subdevices: 8/8
```

```

Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7
card 1: Loopback [Loopback], device 1: Loopback PCM [Loopback PCM]
Subdevices: 8/8
Subdevice #0: subdevice #0
Subdevice #1: subdevice #1
Subdevice #2: subdevice #2
Subdevice #3: subdevice #3
Subdevice #4: subdevice #4
Subdevice #5: subdevice #5
Subdevice #6: subdevice #6
Subdevice #7: subdevice #7

```

Wenn soweit alles funktioniert und auch das Modul `ßnd-pcm-oss` ordnungsgemäß geladen wurde dann kann nun `"/dev/dsp1` als Capturedevice für den Streaming-Client verwendet werden.

[`sc_trans.conf`]:

```

...
capture=1
capturedevice=/dev/dsp1
captureinput=Capture
capturesamplerate=16000
capturechannels=1
...

```

[`ices2.xml`]:

```

...
< stream >
< metadata >
< name >DB0DIN< /name>
< genre >Hamradio Network< /genre >
< description >DB0DIN Livestream< /description >
< url >http://db0din.de< /url >
< /metadata >
< input >
< module >oss< /module >
< param name=rate>16000< /param>
< param name=channels>1< /param>
< param name="device">/dev/dsp1< /param>
< param name="buffer-time">500< /param>
< param name=metadata>1< /param>
< param name=metadatafilename>/tmp/info.txt< /param>
< /input>
...

```

SvxLink oder der remotetrx wird noch als Multi-Tx konfiguriert

[/etc/svxlink/svxlink.conf oder /etc/svxlink/remotetrx.conf]:

```
[RepeaterLogic]
TYPE=Repeater
RX=QuantarRx
TX=MultiTx
MODULES=ModuleEchoLink,ModuleMetarInfo,ModulePropagationMonitor,ModuleSelCallEnc,-
ModuleTrafficInfo
....
```

```
[MultiTx]
TYPE=Multi
TRANSMITTERS=RepeaterTx,StreamingTx
```

```
[RepeaterTx]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
PTT_PORT=/dev/ttyS0
PTT_PIN=DTRRTS
TX_DELAY=0
PREEMPHASIS=0
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18
```

```
[StreamingTx]
TYPE=Local
AUDIO_DEV=alsa:plughw:1,1 #hier wird nun das 1. capture-device des snd-aloop ausgewählt
AUDIO_CHANNEL=0
PTT_PORT=NONE
PTT_PIN=DTRRTS
TX_DELAY=0
PREEMPHASIS=0
```

Jetzt alle Applikationen starten und die Funktion prüfen.
Viel Erfolg wünscht do6ep, Mathias

Linking RemoteTrx und SvxLink (2012-06-20 20:32)

Bei dieser Konfiguration werden von einem Remotetrx/SvxLink-Basis empfangenen Signale an die angeschlossenen Stationen übertragen und dort ausgegeben. Ebenfalls können die angeschlossenen Stationen am EchoLink-Betrieb teilnehmen bzw. entsprechende EchoLink-Verbindungen auf- und abbauen. Die hier dargestellte Variante nutzt sendeseitig einen MultiTx und empfangsseitig die in SvxLink eingebaute Voter-Funktion.

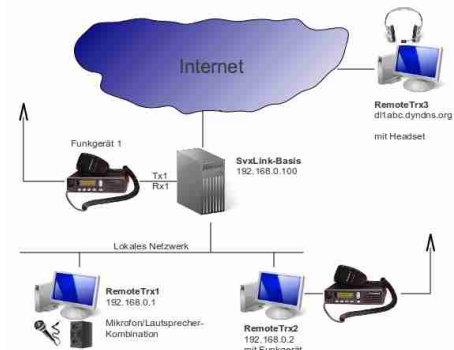
Auf der SvxLink-Basisstation läuft die Applikation svxlink, auf den RemoteTrx(en) die Applikation remotetrx (bei SvxLink mit dabei). Der Verbindungsaufbau erfolgt dabei grundsätzlich von der Svxlink-Basisstation zu den RemoteTrx(en). Achten Sie darauf, dass diese über die IP-Adresse (Achtung bei dhcp) oder per Hostnamen (außerhalb des eigenen Netzes -> dyndns, o.ä) auf dem konfigurierten Port erreichbar sind.

Als Codec kann innerhalb des eigenen Netzes entweder ein RAW-Stream oder der S16-Codec verwendet werden, bei Verbindungen ins Internet über eine DSL-Verbindung sollte man auf den SPEEX und bei geringeren Qualitätsansprüchen auf den GSM-Codec zurückgreifen.

Die Nachteile dieser Konfigurationsvariante sollen aber nicht verschwiegen werden:

- Die Verbindungen zwischen den Stationen sind statisch, also nicht steuerbar (Trennen,Verbinden auf Kommando)
- keine spezifischen Logic-Funktionen für die einzelnen RemoteTrx(e) verfügbar (Metar,Parrot,...)
- Voterfunktionen kosten Zeit, d.h. höhere Latenzen

Eine Weiterentwicklung stellt das LogicLinking dar, welches es das gezielte Verbinden/Trennen einzelner Teilneh-



mer sowie Timeout-Funktionen ermöglichen.

Abbildung 1: Prinzip SvxLink und RemoteTrx(e)

Die nachfolgende Abbildung stellt die einzelnen Konfigurationsabschnitte der SvxLink-Basis und eines RemoteTrx's deutlich(er) heraus.

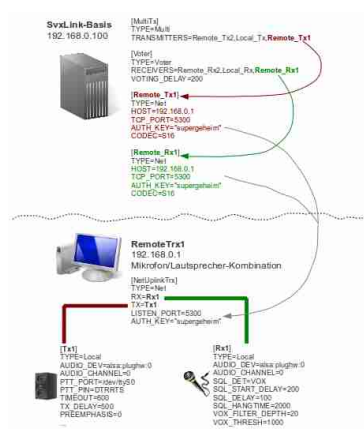


Abbildung 2: SvxLink/RemoteTrx-Konfiguration, Prinzip

Nachfolgend die Konfigurationsdateien für die einzelnen Stationen analog zur Abbildung 1. SvxLink (Basisstation), Konfigurationsdatei svxlink.conf

```

# Konfigurationsdatei svxlink.conf auf der lokalen SvxLink-Basis
[GLOBAL]
MODULE_PATH=/usr/lib/svxlink
LOGICS=RepeaterLogic
CFG_DIR=svxlink.d
TIMESTAMP_FORMAT="%d.%m.%Y %H:%M:%S"
CARD_SAMPLE_RATE=48000

[RepeaterLogic]
TYPE=Repeater
RX=Voter
TX=MultiTx
MODULES=ModuleHelp,ModuleParrot,ModuleEchoLink,ModuleTclVoiceMail
CALLSIGN=DL1ABC
SHORT_IDENT_INTERVAL=60
LONG_IDENT_INTERVAL=60
EVENT_HANDLER=/usr/share/svxlink/events.tcl
DEFAULT_LANG=de_DE
RGR_SOUND_DELAY=0
MACROS=Macros
FX_GAIN_NORMAL=0
FX_GAIN_LOW=-12
OPEN_ON_SQL=50
IDLE_TIMEOUT=2
NO_REPEAT=1

[MultiTx]
TYPE=Multi
TRANSMITTERS=Remote_Tx1,Remote_Tx2,Local_Tx

[Voter]
TYPE=Voter
RECEIVERS=Remote_Rx1,Remote_Rx2,Local_Rx
VOTING_DELAY=200

# zunaechst werden die RX- und TX-Abschnitte für das an der
# SvxLink-Basis angeschlossene Funkgeraet definiert
[Local_Rx]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
SQL_DET=VOX
SQL_START_DELAY=0
SQL_DELAY=0
SQL_HANGTIME=2000
VOX_FILTER_DEPTH=20
VOX_THRESH=1000
CTCSS_FQ=136.5
CTCSS_THRESH=10
SERIAL_PORT=/dev/ttyS0
SERIAL_PIN=CTS:SET
SIGLEV_SLOPE=1
SIGLEV_OFFSET=0

```

```
SIGLEV_OPEN_THRESH=30
SIGLEV_CLOSE_THRESH=10
DEEMPHASIS=0
PEAK_METER=1
DTMF_DEC_TYPE=INTERNAL
DTMF_MUTING=1
DTMF_HANGTIME=100
DTMF_SERIAL=/dev/ttyS0
```

```
[Local_Tx]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
PTT_PORT=/dev/ttyS0
PTT_PIN=DTRRTS
TIMEOUT=600
TX_DELAY=500
PREEMPHASIS=0
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18
```

```
# jetzt folgen die Verweise auf die per Netzwerk anzubindenden
# Stationen (RemoteTrx's)
```

```
[Remote_Tx1]
TYPE=Net
HOST=192.168.0.1
TCP_PORT=5300
AUTH_KEY="supergeheim"
CODEC=S16
```

```
[Remote_Rx1]
TYPE=Net
HOST=192.168.0.1
TCP_PORT=5300
AUTH_KEY="supergeheim"
CODEC=S16
```

```
[Remote_Tx2]
TYPE=Net
HOST=192.168.0.2
TCP_PORT=5301
AUTH_KEY="supergeheim"
CODEC=SPEEX
```

```
[Remote_Rx2]
TYPE=Net
HOST=192.168.0.2
TCP_PORT=5301
AUTH_KEY="supergeheim"
CODEC=SPEEX
```

Nachfolgend die Konfigurationen für die RemoteTrx'e.

RemoteTrx1 (auf 192.168.0.1), Konfigurationsdatei remotetrx.conf:

```
# Konfigurationsdatei remotetrx.conf auf dem ersten RemoteTrx
[GLOBAL]
TRXS=NetUplinkTrx
TIMESTAMP_FORMAT="%c"
CARD_SAMPLE_RATE=48000

[NetUplinkTrx]
TYPE=Net
RX=Rx1
TX=Tx1
LISTEN_PORT=5300
FALLBACK_REPEATER=0
AUTH_KEY="supergeheim"

[Rx1]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
SQL_DET=VOX
SQL_START_DELAY=200
SQL_DELAY=100
SQL_HANGTIME=2000
VOX_FILTER_DEPTH=20
VOX_THRESH=1000
CTCSS_FQ=136.5
CTCSS_THRESH=10
SERIAL_PORT=/dev/ttyS0
SERIAL_PIN=CTS:SET
SIGLEV_SLOPE=1
SIGLEV_OFFSET=0
SIGLEV_OPEN_THRESH=30
SIGLEV_CLOSE_THRESH=10
DEEMPHASIS=0
PEAK_METER=0
DTMF_DEC_TYPE=INTERNAL
DTMF_MUTING=1
DTMF_HANGTIME=100
DTMF_SERIAL=/dev/ttyS0

[Tx1]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
PTT_PORT=NONE
PTT_PIN=DTRRTS
TIMEOUT=300
TX_DELAY=500
PREEMPHASIS=0
DTMF_TONE_LENGTH=100
```

```
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18
```

RemoteTrx2 (auf 192.168.0.2), Konfigurationsdatei remotetrx.conf:

```
# Konfigurationsdatei remotetrx.conf auf dem zweiten RemoteTrx
[GLOBAL]
TRXS=NetUplinkTrx
TIMESTAMP_FORMAT="%c"
CARD_SAMPLE_RATE=48000

[NetUplinkTrx]
TYPE=Net
RX=Rx1
TX=Tx1
LISTEN_PORT=5301
FALLBACK_REPEATER=0
AUTH_KEY="supergeheim"

[Rx1]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
SQL_DET=VOX
SQL_START_DELAY=200
SQL_DELAY=100
SQL_HANGTIME=2000
VOX_FILTER_DEPTH=20
VOX_THRESH=1000
CTCSS_FQ=136.5
CTCSS_THRESH=10
SERIAL_PORT=/dev/ttyS0
SERIAL_PIN=CTS:SET
SIGLEV_SLOPE=1
SIGLEV_OFFSET=0
SIGLEV_OPEN_THRESH=30
SIGLEV_CLOSE_THRESH=10
DEEMPHASIS=0
PEAK_METER=0
DTMF_DEC_TYPE=INTERNAL
DTMF_MUTING=1
DTMF_HANGTIME=100
DTMF_SERIAL=/dev/ttyS0

[Tx1]
TYPE=Local
AUDIO_DEV=alsa:plughw:0
AUDIO_CHANNEL=0
PTT_PORT=/dev/ttyS0
PTT_PIN=DTRRTS
TIMEOUT=300
TX_DELAY=500
PREEMPHASIS=0
```

```
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18
```

Internetstreaming (2012-06-28 12:50)

Hier ist beschrieben, wie man den Funkverkehr des eigenen Repeater/Link im Internet veröffentlichen kann.

2.6 August

(Unwetterwarnung) (2012-08-16 18:42)

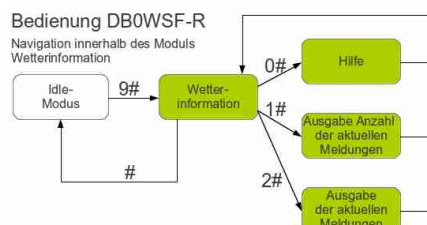
- Kein offizielles Modul -

Dieses Modul gibt eine Unwetterwarnung als Sprachausgabe aus, sobald diese per eMail empfangen wurde. Es wird seit einigen Monaten auf DBOWSF, DM0LEI und weiteren Repeatern bzw. Links getestet. Zum Betrieb ist ein Text-To-Speak-System notwendig, SvxLink selbst generiert keine dynamischen Sprachausgaben. Zum Betrieb sind zwei TCL-Dateien notwendig, die hier heruntergeladen werden können:

WeatherInfo.tcl
ModuleWeatherInfo.tcl

Das File WeatherInfo.tcl nach /usr/share/svxlink/events.d und ModuleWeatherInfo.tcl nach /usr/share/svxlink/modules.d kopieren.

Weiterhin ist ein spool-Verzeichnis anzulegen: /var/spool/svxlink/weatherinfo mit einer Schreibberechtigung für den User svxlink.



Bedienung Unwettermodul (konfigurationsabhängig, hier DBOWSF-R)

Nachfolgend Beispiele für Unwetterwarnungen:

Unwetterwarnung Landkreis Leipzig

Unwetterwarnung Landkreis Göttingen

Kommandos

0 # - Hilfe

1 # - Gibt die Anzahl der aktuellen Unwetterwarnungen aus

2 # - Gibt die aktuellen Unwetterwarnungen in zeitlicher Reihenfolge aus

Beispielkonfiguration in der /etc/svmlink/svmlink.d/WeatherInfo.conf

```
[ModuleWeatherInfo]
```

```
NAME=WeatherInfo
```

```
PLUGIN_NAME=Tcl
```

```
ID=12
```

```
TIMEOUT=15
```

```
DELETE_AFTER=120
```

```
ALERT=0
```

```
SPOOL_DIR=/var/spool/svmlink/weatherinfo
```

```
PLAY_DIR=/usr/share/svmlink/sounds/de_DE/WeatherInfo
```

Beschreibung

NAME

Name des Moduls

ID

Identifikationsnummer für dieses Modul. Diese Identifikationsnummer darf im gesamten System nur einmalig vergeben werden.

TIMEOUT

Zeit in Sekunden, nach der das Modul automatisch deaktiviert wird.

DELETE_AFTER

Zeit in Minuten, nach der die gespeicherten Unwetterwarnungen automatisch gelöscht werden.

ALERT

=1, wenn vor der Unwetterwarnung ein Alarmton ausgegeben werden soll

SPOOL_DIR

Verzeichnis, in dem die Unwetterwarnungen erwartet werden, die Dateinamen entsprechen folgender Struktur:

CALL.Hashcode.wav - die auszugebende Wav-Datei

CALL.Hashcode.info - der Text aus dem die Unwetterwarnung generiert wurde

PLAY_DIR

Verzeichnis, in dem die Meldungen kopiert und verarbeitet werden. Nach der Ausgabe werden sie in das \$PLAY_DIR/archive Verzeichns verschoben, in dem sie noch für DELETE_AFTER verbleiben um dann endgültig gelöscht zu werden.

(Verkehrsinformation) (2012-08-17 07:37)

- Kein offizielles Modul -

Dieses Modul (eigentlich "Gefahrenmeldung") wird seit einigen Monaten getestet und funktioniert relativ gut. Im Release/Trunk wird es allerdings nie erscheinen, es soll nur demonstrieren was mit SvXLink möglich ist.

Es besteht auch eine begrenzte Steuerungsmöglichkeit nach Aktivierung des Modules.

Kommandos:

0 # - Hilfe

1 # - Gibt nur die Anzahl der aktuellen Verkehrsmeldungen aus

2 # - Gibt alle aktuellen Verkehrsmeldungen in zeitlicher Reihenfolge aus

Beispielkonfiguration in der /etc/svmlink/svmlink.d/TrafficInfo.conf

```
[ModuleTrafficInfo]
```

```
NAME=TrafficInfo
```

```
PLUGIN_NAME=Tcl
```

```
ID=11
```

```
TIMEOUT=15
```

```
DELETE_AFTER=120
```

```
HINZ=1
```

```
SPOOL_DIR=/var/spool/svmlink/trafficinfo
```

```
PLAY_DIR=/usr/share/svmlink/sounds/de_DE/TrafficInfo
```

Beschreibung

NAME

Der Name des Moduls, z.B. NAME=TrafficInfo

PLUGIN_NAME

ID

Eindeutige ID des Moduls, mit der das Modul gestartet wird.

TIMEOUT

Zeit in Sekunden, nach der das Modul beendet wird, falls keine Aktivität feststellbar ist.

DELETE_AFTER

Zeit in Minuten, nach der die gespeicherten Verkehrsmeldungen automatisch gelöscht werden

HINZ

Ausgabe des Verkehrsmeldung-Aktivierungstones am Anfang und Ende der Meldung, siehe Hinz-Triller

SPOOL_DIR

Verzeichnis, in der die auszugebenden Verkehrsmeldungen erwartet werden, die Dateien besitzen die Struktur:

CALL.Hashcode.wav - das auszugebende Wav-File

CALL.Hashcode.info - das .info-File einhält den Text aus dem das wav-file generiert wurde

Der Hashcode wird aus dem Text-Inhalt der Verkehrsmeldung generiert, CALL muß dem Rufzeichen entsprechen, unter dem der Repeater/Link konfiguriert wurde.

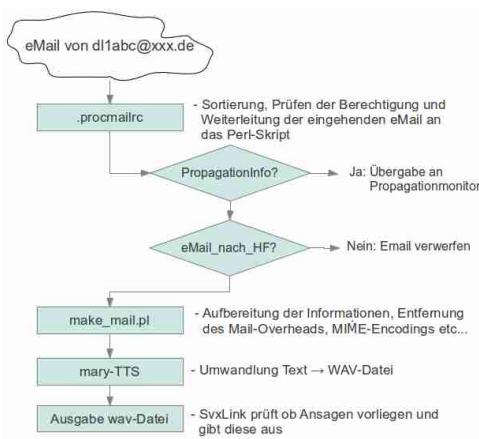
PLAY_DIR

Verzeichnis, in das die Files für die Ausgabe kopiert werden, nach dem Abspielen werden diese Files mit einem öd-äm Anfang des Dateinames versehen und in das \$PLAYDIR/archive-Verzeichnis kopiert in dem sie für DELETE_AFTER verbleiben um danach gelöscht zu werden.

(eMail-2-Rf) (2012-08-17 08:35)

- Kein offizielles Modul -

Dieses Modul gibt einen als eMail empfangenen und in eine wav-Datei umgewandelten Text aus, es läuft seit einigen Monaten auf DBOWSF und DMOLEI.



Email-2-Rf (Prinzip)

Persönliche Begrüßung eingehender EchoLink-Stationen (2012-08-19 16:53)

Seit dem Trunk, Rel. 2223 (ab 18.08.2012) ist es möglich einer eingehenden EchoLink-Station eine persönliche Begrüßung zu übermitteln (rufzeichenspezifisch). Hierzu ist unter Beachtung des locale-Konzeptes eine kleine tcl-Datei im Verzeichnis /usr/share/svxlk/events.d/local zu erstellen, Dateiname z.B. user_greeting.tcl:

```
# Ergänzung zur individuellen Begrüßung
# by Michael Hartje, DK5HH

namespace eval EchoLink {

#
# Executed when an incoming connection is accepted
#
proc remote_greeting {callsign} {
    global langdir; # aus der Hauptebene in events.tcl
    set callsign [string toupper $callsign]; # nur zur Sicherheit
    set callsign [string map {- " "} $callsign]; # Trennzeichen verwenden
    set found [scan $callsign "%s%s" call suffix];
    if {$found > 0 && [file exists
"$langdir/EchoLink/greeting-$call.wav"]} {
```

```

    playSilence 500;
    playFile "$langdir/EchoLink/greeting-$call.wav";
} else {
    playSilence 1000;
    playMsg "greeting";
}
}
}
# End Namespace

```

Die Funktionsweise ist recht einfach, im Verzeichnis /usr/share/svxlink/sounds/de_DE/EchoLink sind für eine deutsche Sprachumgebung die personalisierten Begrüßungsdateien im Format greeting-CALL.wav abzulegen, wobei CALL durch das betreffende Rufzeichen ersetzt wird. Bei einer eingehenden EchoLink-Verbindung (z.B. von DK5HH) wird im Soundverzeichnis nach einer Datei greeting-DK5HH.wav gesucht, ist diese vorhanden wird sie abgespielt.

SvxLink fernsteuern (2012-08-20 08:48)

SvxLink läßt sich per netcat wie folgt fernsteuern:

1) SvxLink auf dem Server mit

```
nc -lk 10000|svxlink
```

starten. Die 10000 ist der Port auf dem netcat lauscht bzw. die Kommunikation zwischen SvxLink und dem Steuer-PC abläuft.

2) Vom Steuer-PC aus kann SvxLink dann mit z.B.

```
echo -n "2 #1 #nc svxlink-server-ip 10000
```

ferngesteuert werden wobei svxlink-server-ip die IP-Adresse/Hostname des Rechners ist auf dem svxlink läuft.

SvxLink Streaming & GNU Radio (2012-08-21 21:55)

Dieser Artikel beschreibt das im August 2012 neu hinzugekommene Streaming von Audiodaten per UDP des Svxlink. Hier wird die von SMOSVX erwähnte Anbindung von SvxLink an GnuRadio mit einem Beispiel gezeigt. GnuRadio [1] ist eine Open Source Entwicklungsumgebung für die OSI/ISO-Ebene "0". Es wird insbesondere für

SDR-Systeme eingesetzt.

Das hier erläuterte Beispiel ist von Michael (DK5HH) und soll Signalnach- und -vorverarbeitung in einem als Repeater betriebenen Svxlink ermöglichen. Das GnuRadio-GRC-Beispiel erzeugt aus dem UDP-Audiostream des Svxlink:

1. eine Oszilloskop-Schirm-Darstellung
2. einen FFT-Plot
3. eine Ausgabe auf die Soundkarte (mit Lautstärkereger)

Die Kommunikation zwischen Svxlink und GnuRadio erfolgt im Beispiel über den UDP-Port 42420.

In der /etc/svxlink/svxlink.conf sind hierfür folgende Erweiterungen notwendig:

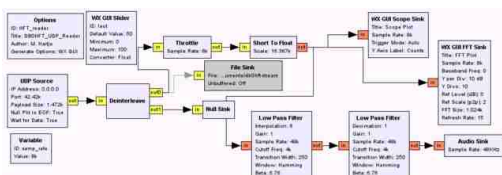
```
[TxUDP]
TYPE=Local
#AUDIO_DEV=udp:127.0.0.1:42420
AUDIO_DEV=udp:100.012.123.123:42420
AUDIO_CHANNEL=0
PTT_PORT=NONE
PTT_PIN=DTR
TIMEOUT=0
TX_DELAY=0
PREEMPHASIS=0
CTCSS_LEVEL=0
DTMF_TONE_LENGTH=100
DTMF_TONE_SPACING=50
DTMF_TONE_AMP=-18
```

```
[TxMult]
TYPE=Multi
TRANSMITTERS=TxHft , TxUDP
```

Weiter oben in der svxlink.conf im Bereich [RepeaterLogic] muss dann noch der Parameter TX=TxMult gesetzt werden, damit dieser Multi-TX auch angesprochen wird.

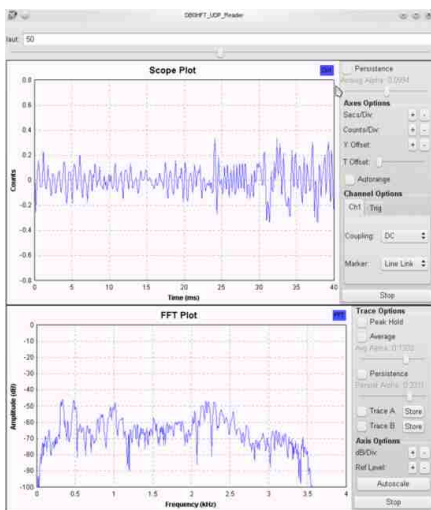
Der von Svxlink im UDP-Paket gelieferte Datenstrom besteht aus 2 Audio-Kanälen zu jeweils 16 Bit. Diese müssen für eine weitere Verwendung entkoppelt werden (deinterlaced) werden. Der unbenutzte Stream geht dann in die Null-Sink (diese gleichzeitige Übertragung von 2 Kanälen ist evtl. noch ein Bug im Svxlink; Stand 21.08.2012, d.Red.).

Für die Abtastratenkonversion an der Soundkarte habe ich zunächst einen digitalen Interpolator und dann anschließend ein Tiefpassfilter vorgesehen, um die unangenehm pfeifenden Oberschwingungen aus dem Audiokanal herauszubekommen.



Schematic Signalfluß in GNU Radio

Noch eine Besonderheit ist die Konversion von short nach float. Dabei nutze ich den Scale-Wert zwischen 16000 und 32000 (hier gewählt $2^{14}-1=16383$) um auf einen Wert zwischen -1 und +1 zu skalieren, so dass die Soundkarte nicht übersteuert wird.



Ausgabebild GNU Radio

Schön ist, dass der Stream von Svmlink nur dann Informationen schickt, wenn wirklich was los ist... in den Ruhezeiten (ohne Input) des sendenden Relais kommt kein Stream. (Ansagen und Rogerpiep werden aber mit übertragen) Für einen Programmierer ist es nun keine große Aufgabe, aus dem UDP-Stream weiteres zu machen. Für die umgekehrte Richtung, die Sendeseite, muss dann entsprechend noch ein UDP-Empfänger im Svmlink (svmlink.conf) aufgesetzt werden.

Es sei auch der Hinweis gestattet, dass mit diesem neuen Stream des SVXLink sich recht leicht jetzt auch der Anschluss an einen Internet-Streamer IceCast erreichen lässt. – Und das geht dann auch für alle Versionen des Linux-Kernels. Des weiteren ließe sich mit diesem neuen Streaming-Konzept des Svmlink die Konstruktion des NetRX und NetTX in Svmlink zumindest in Teilen mit diesem Weg aufgreifen und ggf. auch einfach realisieren; hat allerdings dann die oben beschriebenen unangenehmen Nebeneffekte mit (störendem) unnützem Audio.

Kontakt:

Prof. Dr. Michael Hartje, DK5HH - dk5hh {ät } darc . de

Downloads:

GRC-file DB0HFT_reader.grc

aus GRC abgeleitetes Python-Skript HFT_reader.py

Links:

[1] GNU Radio - Project

Selektivruf bei eingehender EchoLink-Verbindung auslösen (2012-08-22 21:26)

In vielen Fällen möchte man den oft langatmigen und weniger interessanten Relais-Gesprächen nicht ausgesetzt sein. Um aber trotzdem für «wichtige» EchoLink-Stationen erreichbar zu sein, kann man bei kommerziellen Geräte wie z.B. dem Motorola GM1200 eine Selektivruffunktion aktivieren (ZVEI1, EIA,...). Für den tcl-Namespace ist

dann eine kleine Anpassungen erforderlich.

- 1) Aktivieren Sie den SelCall-Encoder in der SvxLink-Konfiguration, siehe: http://svxlink.de/?page_id=187
- 2) Erstellen Sie unter Beachtung des locale-Konzeptes eine tcl-Datei, z.B. sel5.tcl mit folgendem Inhalt:

```
#
# This routine sends out a Sel5call over Rf
# when a VIP-EchoLink station was connecting
# the own node
#
namespace eval EchoLink {

    proc remote_connected {call} {

        SelCall::setMode "ZVEI1"; # define the SelCall-mode

        if {$call == "DL1ABC" || $call == "SMOSVX-L"} {

            SelCall::play "01231"; # the own station with call
                                   # "01231" configured
        }

        playSilence 500;
        playMsg "connected";
        spellEchoLinkCallsign $call;
    }
}
```

- 3) Starten Sie SvxLink neu und warten Sie auf die nächste eingehende EchoLinkverbindung von "DL1ABCöder SSMOSVX-L", bitte an die eigenen Erfordernisse anpassen. Stellt eine der konfigurierte Partner-Stationen eine Verbindung zum eigenen EchoLink-Node her, so wird ein Selektivruf über HF ausgesendet, in diesem Falle die Tonfolge zum Ruf "01231".

Informationsansagen zeitlich gesteuert mit cron (2012-08-23 18:26)

Vorbereitete Informationsansagen kann man relativ einfach kurz vor dem Abfallen des Relais aussenden – also zu einem Zeitpunkt, an dem noch jemand zuhört, aber eigentlich niemand mehr sprechen möchte. Zu diesem Zeitpunkt gibt es zumeist mindestens einen Zuhörer, der vorher das Relais vielleicht nur aufgetastet hatte.

Die Informationsansagen sollten nicht mehr als 20 s lang sein. Dazu werden die vorbereiteten WAV-Dateien in einem Verzeichnis unterhalb von `/usr/share/svxlink/sounds/de_DE` abgelegt, z.B. in LocalAudio. Die Dateien sollten alle mit "idle-" beginnen, also z.B. `Idle-00-OV-Abend-Einladung.wav`

Weitere Ansagen zur Benutzung des Relais, den vereinbarten Kurzwahlen, helfen Durchreisenden oder anderen noch nicht umfassend Informierten, sich schneller zurecht zu finden.

Mit der folgenden Prozedur unter /usr/share/svxlink/sounds/de_DE/events.d/local können die vorbereiteten Sounddateien jedes Mal durchgesehen und dann abgespielt werden.

```
#
# Abspielen von Info-wav bei Idle des Relais
#
# Entwurf DK5HH
#

namespace eval RepeaterLogic {

    #Counter which Idle-sound shall be the next
    variable idle_nr 0;
    variable last_idlesound 100000000;

    # minimum time between 2 idle sounds in seconds
    variable min_time_between_idlesound 300;

    proc repeater_idle {} {
        set filelist [get_idle_files];
        set idle_max [llength $filelist];
        set filelist [lsort -ascii $filelist];
        variable idle_nr;
        variable last_idlesound;
        variable min_time_between_idlesound;
        set now [clock seconds];
        # more than 300 s since last idlesound
        if { $now - $last_idlesound > $min_time_between_idlesound } {
            set last_idlesound $now;
            if { $idle_nr >= $idle_max } {
                set idle_nr 0;
            }
            set psound [lindex $filelist $idle_nr]
            set idle_nr [expr { $idle_nr + 1 } ];
            puts "playing Idle-Sound $psound. $idle_nr\/$idle_max";
            playMsg "LocalAudio" $psound;
            playSilence 800;
        }
    }

    proc get_idle_files {} {
        variable dir "/usr/share/svxlink/sounds/de_DE/LocalAudio";
        variable ext "wav";
        variable files "";
        foreach file [glob -nocomplain -directory $dir Idle-*.wav] {
            set foundfile [file tail [file rootname $file]];
            lappend files $foundfile;
        }
        return $files;
    }
}

# end Namespace RepeaterLogic
```

Durch cron kann man die Ansagen durch Umbenennen zu einem bestimmten Zeitpunkt aktivieren oder deaktivieren – also zum OV-Abend eine Woche vorher die Ansage aktivieren und während des OV-Abends wieder deaktivieren. Also: z.B. aus Idle-.... ein idle-.... machen. – Dann kann die Ansage nächste Woche Dienstag ist OV-Abend, willst Du nicht auch kommen?einladend wirken.

Ein beliebiger User der Gruppe svxlink kann einen Eintrag mit crontab -e anlegen. Dabei wird das folgende kleine Shell-Skript idle-activate.sh verwendet:

```
#!/bin/bash
# wird mit cron aufgerufen zum Aktivieren / Dekativieren
# von Idle-Sounds
DIR=/usr/share/svxlink/sounds/de_DE/LocalAudio/
cd $DIR
ACTIVE=$1
if [ -z $ACTIVE ]
then
    echo "usage: Parameter EIN|AUS \"Datei-pattern\""
fi
#U_ACT=tr '[:lower:]' '[:upper:]' $ACTIVE
if [ $ACTIVE = "EIN" ]
then
    for renfile in $(find ./ -name "$2" )
    do
        re1=$(basename $renfile)
        re2=${re1/idl/Idl}
        mv $re1 $re2
    done
fi
if [ $ACTIVE = "AUS" ]
then
    for renfile in $(find ./ -name "$2" )
    do
        re1=$(basename $renfile)
        re2=${re1/Idl/idl}
        mv $re1 $re2
    done
fi
```

das kleine Shell-Hilfsskript idle-activate.sh legen wir im \$HOME-Verzeichnis dieses Benutzers (zB dk5hh) ab und machen es mit

```
chmod 775 idle-activate.sh
```

ausführbar.

Als Eintrag in der crontab wird erforderlich (mit crontab -e aufrufen)

```
# normaler OV-Abend: erster Dienstag
#2 21 * * 2 [ $(date +"%\m") -ne $(date -d 7days +"%\m") ] &&
/home/dk5hh/idle-activate.sh EIN "*OV-abend*"
```



```

#2 21 * * 2      [ $(date +"%\m") -eq $(date -d 7days +"%\m") ] &&
/home/dk5hh/idle-activate.sh AUS "*OV-abend*"
# verschoben auf zweiten Dienstag OV-Abend
2 21 * * 2      [ `date +%e` -gt 0 -a `date +%e` -lt 8 ] &&
/home/dk5hh/idle-activate.sh EIN "*OV-abend*"
2 21 * * 2      [ `date +%e` -gt 7 -a `date +%e` -lt 15 ] &&
/home/dk5hh/idle-activate.sh AUS "*OV-abend*"

```

Die Einträge in der crontab werden jeweils (hier im Beispiel) um 21:02 an jedem Dienstag aufgerufen, wenn der Eintrag nicht mit einem "#äls Kommentar ausgeschaltet werden – hier die ersten beiden Zeilen.

Der Aufruf des auszuführenden Programms ist eine logische Zusammensetzung mit "& &". Ist der erste Teil logisch wahr, wird der zweite Teil ausgeführt.

Auch der erste Teil in [...] ist eine logische Verknüpfung mit a"(and). Es wird das Kommando date verwendet, um den Tag des Monats vom aktuellen Datum zu finden und zu verknüpfen.

Für den OV-Abend am ersten Dienstag des Monats ist es besonders einfach. Wir prüfen einfach, ob der Monat von heutigen Dienstag und der Monat des nächsten Dienstags in 7 Tagen den gleichen Monat haben – wenn nicht, dann schalten wir den Idle-Sound aktiv.

Haben wir sehr viele allgemeine Informationsansagen die nicht zeitabhängig sind und wollen wir einzelne Ansagen häufiger abspielen, legen wir einfach einen Link an auf die Sounddatei Idle-00-OV-abend-Einladung.wav:

```
ln -s Idle-00-OV-Abend-Einladung.wav Idle-10-OV-Abend-link.wav
```

Mehrere Links mit unterschiedliche Zahlen streuen die Ansage zwischen die anderen. Die Liste der abzuspielenden wav-Dateien wird in der oben gezeigten TCL-Prozedur jeweils neu alphabetisch geordnet.

Anmerkung: Aus unserer Erfahrung scheint es gut, für die Aufbereitung der Ansagen ein TTS-System (Text-to-Speech) mit einer angenehmen Stimme zu verwenden.

Informationsansagen zeitlich gesteuert mit cron | SvXLink.de (2012-08-23 18:37:17)

[...] EchoLink-StationenTrennung EchoLink-Verbindung verhindernUnterschiedliche Roger-Beep'sInformationsansagen zeitlich gesteuert mit cronSelektivruf bei eingehender EchoLink-Verbindung auslösenSvXLink für 16k-Sounds selber [...]

2.7 September

SvXLink auf Raspberry-Pi (2012-09-11 10:09)

(Diese Anleitung wurde für das Amateurfunkmagazin CQ DL verfasst und ist in der Urfassung in Ausgabe 10/2012 nachzulesen. Die Veröffentlichung des Beitrags auf dieser Webseite erfolgt mit freundlicher Genehmigung der CQ DL. An dieser Stelle lesen Sie den um neue Erkenntnisse aktualisierten Beitrag mit Stand 23. September 2013)



Ein Echolink-Rechner mit der Leistungsaufnahme im untersten einstelligen Wattbereich, kann das gehen? Ja, es kann: mit dem Raspberry Pi, einem ARM-Prozessor-basierten Minicomputer im Scheckkartenformat. Geringste Stromkosten im Dauerbetrieb sind damit keine Utopie. So benötigt die gesamte Station im obigen Bild (inkl. Rechner, Funkgerät plus zwei Schaltnetzteile und einer vor alles geschalteten IP-Steckdose) bei Rx 10,5 Watt und im Sendefall [1 Watt HF] 18 Watt.

Es war schon eine kleine Sensation, als etwa Anfang des Jahres 2012 folgende drei Schlagworte im Internet las: preiswert, stromsparsam, Linux. All das bezog sich auf einen neuen Scheckkarten-großen Mini-Rechner zum sagenhaften Preis (inkl. Versand) von nur ca. 39 € mit der folgenden technischen Ausstattung:

Technische Daten des Raspberry Pi's



Platinengröße: 85,6 mm × 53,98 mm × 17 mm

Prozessor (CPU): ARM1176JZF-S (700 MHz, übertaktbar)

Speicher (RAM): 512 MB, nicht erweiterbar

Grafikprozessor (GPU): Broadcom VideoCore IV

System-on-Chip (SoC): Broadcom BCM2835

Peripherie: 1 × Ethernet 10/100 MBit, 2 × USB, Audio-Out, HDMI

Strom: über Micro-USB, ca. 300...700 mA, (mit USB-Geräten ca. 3,2 W vor Schaltnetzteil gemessen)

Betriebssysteme: Linux (Debian, Fedora, Arch Linux)

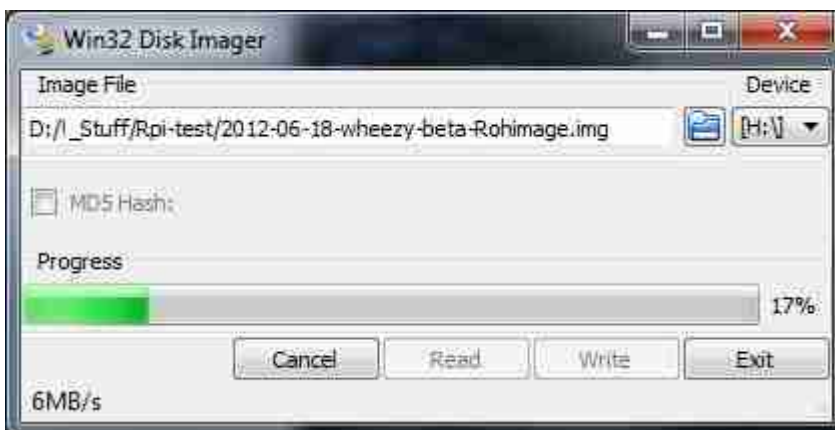
Entstanden ist der „Raspberry Pi“ innerhalb der gleichnamigen Foundation, eine in Großbritannien als Wohltätigkeitsorganisation eingetragene Stiftung. Ihr Ziel ist es das Studium der Informatik zu fördern. Während die Entwickler ein preiswertes System u.a. zum Heranführen Jugendlicher im Sinn hatten, so wurde bereits in Foren früh klar: Der Minirechner hat sicher Potenzial für Amateurfunkwendungen bei wenig Leistungsaufnahme. Schon früh

dachte ich daran, meinen stromhungrigen Pentium-3-PC für sporadischen Echolink-Gateway-Betrieb zu ersetzen, der sich mitsamt Peripherie stolze 60 W aus dem Stromnetz gönnt. Der Einsatz von solchen Mini-Rechnern wie dem Raspberry Pi, oder einem so genannten Embedded-System, ist nichts Neues [1]. Beim Raspberry Pi bestanden anfangs aufgrund der weltweit hohen Nachfrage zunächst Lieferschwierigkeiten. Mitte Juli 2012 wurden erst ca. 10 000 Stück ausgeliefert - dem gegenüber standen ca. 250 000 Vorbestellungen. Beziehen kann man ihn per Internet, z.B. [2, 3]. Die Konfiguration des Linux-Systems auf dem Raspberry Pi bedarf etwas Zeit, ist aber – notfalls mit etwas Google-Hilfe – zu bewältigen. Generell stellt das Internet die beste Bedienungsanleitung für das freie Betriebssystem dar. Es gibt zahlreiche Webseiten, die bei Linux- und auch einigen Raspberry-Pi-Fragen weiterhelfen. Nachfolgend meine Herangehensweise, die ausdrücklich kein Kochrezept sein kann. Es gibt mehrere Konfigurationswege. Außerdem schreitet die Linux-Softwareentwicklung für das Board zügig voran, sodass hier zu lesende Textpassagen schnell überholt sein können.

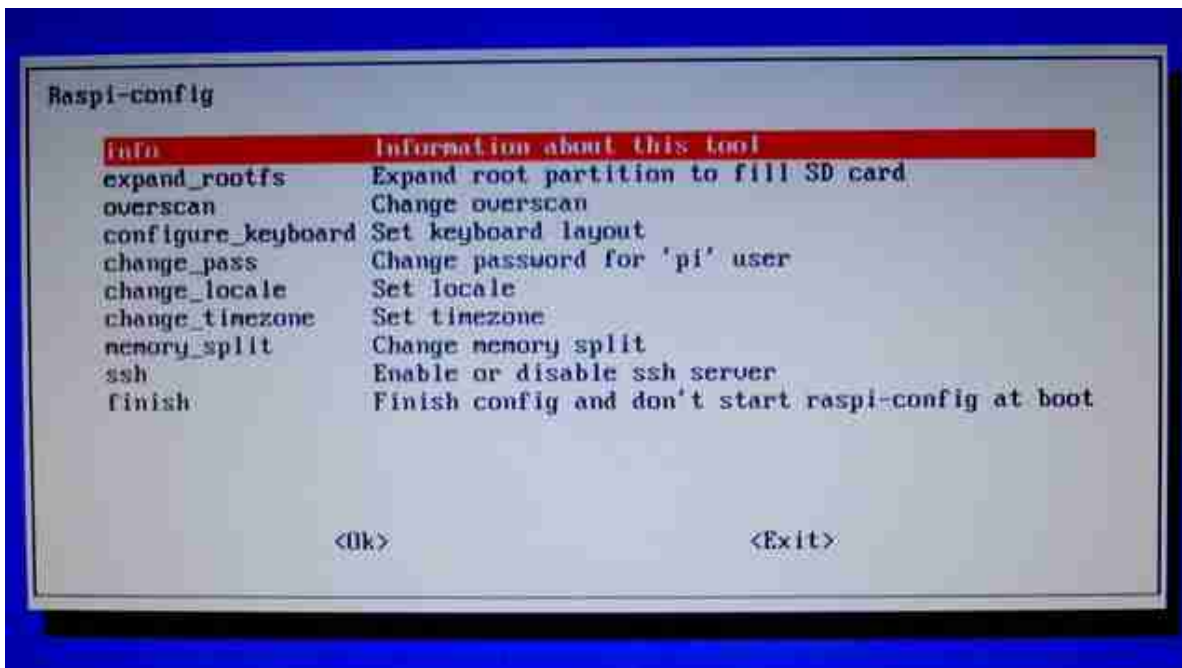
Es geht an die Software

Die Echolink-Anwendung wird als Modul innerhalb des SvxLink-Pakets realisiert. Mein Anwendungsfall sieht nur die Anbindung einer Simplex-Frequenz vor, sodass die „Exil-OMs“ weiterhin auf der OV-Welle präsent sein können und Bekanntschaften bestehen bleiben. Weiterhin macht es großen Spaß auf Reisen z.B. aus Nordamerika oder Australien Kontakt mit den Daheimgebliebenen zu halten.

Zunächst müssen wir Linux für das Board herunterladen [7]. Die Raspberry Pi Foundation bietet dazu die eigene Distribution Raspbian, ein Debian-„WheezyDerivat. Diese Image-Datei ist ausgepackt ca. 1,8 GB groß, wird auf eine mind. 2 GB große SD-Karte kopiert. Unter Windows gelingt dies mit dem Win32Imager am einfachsten. Detaillierte Anleitungen zu diesem Schritt finden sich im Netz.



Zur Erstbedienung schließt man den Monitor, internetfähiges Netzwerk und eine USB-Tastatur an. Der Systemlogin gelingt mit dem Nutzer pi und Passwort raspberry – Achtung, Y und Z sind wegen fehlender lokaler Tastatureinstellung möglicherweise noch vertauscht. Nach dem Login öffnet sich die „Raspi-config“-Oberfläche zur Grundkonfiguration des Systems. Hier gibt es folgende Optionen:



- Partitionierung von SD-Karten, die größer als 2 GB sind (Wenn man das Debian-Image drauf kopiert, werden ca. 2 GB belegt. Speicherplatz darüber verbleibt beim Einsatz von größeren SD-Karten zunächst frei)
- Einstellung des Overscanmodus für optimale Darstellung auf dem Bildschirm
- Keyboardlayout (hier definieren Sie bitte die deutsche Tastatur)
- Passwort-Änderung für den Standard-Nutzer „Pi“
- Zeitzone einstellen Wahl des Memory-Splits. CPU und Grafik teilen sich den Speicher, für unsere Anwendung braucht die CPU den größten Wert (Standardeinstellung)
- SSH-Server ein/auschalten. Den SSH-Server hier aktivieren, damit man sich künftig ohne Monitor über das Netzwerk einloggen kann. Dies gelingt z.B. mit der Freeware Putty [8] vom PC aus. Im Heimnetzwerk vorhandene Internetrouter weisen dem Rasperry Pi per DHCP automatisch eine interne IP-Adresse zu.

Sofern nötig, lässt sich die Oberfläche später wieder durch

```
sudo raspi-config
```

starten. Nochmal zurück zur SD-Kartengröße. Für die SvxLink-Anwendung ist eine 2-GB-Karte knapp, besser wäre mind. eine 4-GB-Karte. Soll dennoch eine 2-GB-Karte zum Einsatz kommen, kann man Teile der grafischen Oberfläche und weitere Pakete deinstallieren, um Platz zu schaffen. Folgende Zeilen sind kein Optimum, aber ein Anfang, der später in bis zu 750 MB zu Gunsten unserer Arbeit resultieren kann. „Sudo“ verleiht dabei Administratorrechte und wird uns noch häufiger begegnen, da der Nutzer „pi“ nur Anwenderrechte hat.

```
sudo rm /opt/vc/src/hello_pi -r
sudo apt-get remove x11-common midori lxde python3 python3-minimal lxde-common lxde-icon-theme omxplayer
libraspberrypi-doc manpages
sudo apt-get autoremove 'sudo dpkg --get-selections | grep -v "deinstall grep x11 | sed s/install/'
sudo apt-get autoremove
sudo apt-get clean
```

Da fast täglich Systemupdates verfügbar sind, wird die Vorinstallation mit Hilfe des Paketmanagers Aptitude auf den neuesten Stand gebracht:

```
sudo apt-get update
sudo apt-get upgrade
```

Aptitude lädt nun einige Pakete nach:

```
Get:10 http://http.debian.net wheezy/contrib armel 2012-06-18-0227.32.pdiff [31
B]
Get:11 http://http.debian.net wheezy/non-free armel 2012-06-19-0222.33.pdiff [71
9 B]
Get:12 http://http.debian.net wheezy/contrib 2012-06-18-0227.32.pdiff [31 B]
Get:13 http://http.debian.net wheezy/main 2012-06-18-0227.32.pdiff [6,013 B]
Hit http://archive.qmh-project.org unstable/main armel Packages
Get:14 http://http.debian.net wheezy/non-free 2012-06-21-0214.11.pdiff [48 B]
Ign http://archive.qmh-project.org unstable/main Translation-en_GB
Ign http://archive.qmh-project.org unstable/main Translation-en
Get:15 http://http.debian.net wheezy/contrib armel 2012-06-18-0227.32.pdiff [31
B]
Get:16 http://http.debian.net wheezy/non-free armel 2012-06-19-0222.33.pdiff [71
9 B]
Get:17 http://http.debian.net wheezy/contrib 2012-06-18-0227.32.pdiff [31 B]
Get:18 http://http.debian.net wheezy/main 2012-06-18-0227.32.pdiff [6,013 B]
Get:19 http://http.debian.net wheezy/main armel 2012-06-18-1416.08.pdiff [2,297
B]
Get:20 http://http.debian.net wheezy/contrib armel 2012-06-19-0222.33.pdiff [382
B]
Get:21 http://http.debian.net wheezy/non-free armel 2012-06-20-0231.53.pdiff [46
2 B]
Get:22 http://http.debian.net wheezy/contrib 2012-06-20-0231.53.pdiff [587 B]
100% [18 Translation-en rred 0 B]
```

Zum komfortablen Weiterarbeiten installiere ich zügig den Midnight Commander – seine Ähnlichkeit zum Norton Commander aus vergangenen DOS-Tagen wird man schnell erkennen. Die Kommandozeile mit Aptitude sieht also so aus:

```
sudo apt-get install mc
```

Nun ist die Debian-Distribution aktuell und nutzbar. Was noch fehlt, ist ein Update des Betriebssystemkerns, also der Kernel. Alle damit verbundenen Dateien bezeichnet man bei dem Raspberry Pi und seinen Debian-Distributionen gern als Firmware. Diese kann man manuell updaten, aber auch automatisiert. Dafür hat der Computerfreak Hexxeh seine Rpi-Updater-Software entwickelt [9]. Damit diese funktioniert, laden wir zunächst folgende Pakete:

```
sudo apt-get install ca-certificates git-core
```

Nun laden wir den Rpi-Updater selbst:

```
sudo wget http://goo.gl/1BOFJ -O /usr/bin/rpi-update
```

Damit man das Tool ausführen kann, müssen die entsprechenden Rechte gesetzt werden. Dies erledigt chmod. Anschließend starten wir den Update-Vorgang:

```
sudo chmod +x /usr/bin/rpi-update
sudo rpi-update
```

Nun haben wir genug geändert. Ich empfehle hier einen Neustart mit

```
sudo reboot
```

Programmcode übersetzen

Fürs erste wäre die Installation abgeschlossen. Kommen wir zur angedachten Anwendung: einem sparsamen Echolink-Server. Von SvxLink gibt es zwar fertig installierbare Softwarepakete. Da das Raspberry Pi aber mit einer ARM- anstelle einer X86-basierten CPU ausgestattet ist, müssen wir das Softwarepaket selbst auf dem Board kompilieren, also die Quelltexte in ausführbaren Programmcode übersetzen. Die Rechenleistung der mit 700 MHz getakteten CPU ist begrenzt, man nehme sich also etwas Zeit – es dauert ca. 25 Minuten!

Zunächst benötigen wir wieder einiges an Software zur Nachinstallation:

```
sudo apt-get install subversion libsigc++-2.0-dev g++ make libsigc++-1.2-dev libgsm1-dev libpopt-dev tcl8.5-dev
libgcrypt-dev libspeex-dev libasound2-dev alsa-utils
```

Dann wechseln wir mit

```
cd /home
```

in ebendieses Verzeichnis und laden uns SvxLink in der (automatisch aktuellsten) Trunk-Version vom Server herunter mit

```
sudo svn co svn://svn.code.sf.net/p/svxlink/svn/trunk svxlink-trunk
```

Zur Konfiguration künftiger Dateien ist ein Texteditor nötig. Ich bevorzuge den internen des Midnight Commanders. Da auch Systemdateien betroffen sind, die wir nur mit Administratorrechten bearbeiten dürfen, starten wir letzteren durch Eingabe von

```
sudo mc
```

Der interne Texteditor des Midnight Commanders muss indes auch aktiviert werden. Man drücke F9, „Options“, „Configuration“, „Other options“ und setze einen Haken bei „Use internal edit“.

Im Midnight Commander wechseln wir in das heruntergeladene Source-Verzeichnis von SvxLink. Im dortigen Hauptverzeichnis liegt die Konfig-Datei `makefile.cfg`, die wir mit F4/Edit öffnen. Sie listet standardgemäß einen Eintrag, der SvxLink im Betrieb veranlassen wird, das interne Soundmixing mit 16 kHz zu leisten. Dies kann dem Raspberry Pi mehr CPU-Last abfordern. Ich habe gute Erfahrungen gemacht, den Wert auf 8000 (Hz) zu reduzieren. Meiner Meinung nach fällt die geringere Audio-Qualität infolge der Übertragung durch Funkgeräte und ihren NF-Passbandbreiten nicht groß auf. Suchen und Ändern Sie den Eintrag von `"16000"` auf `"8000"`:

```
CXXFLAGS += -DINTERNAL_SAMPLE_RATE=16000
```

Für eine geringe CPU-Last im laufenden Betrieb schlagen DL1HRC und DC5PI zudem folgende Parameterzeile in gleicher Datei vor:

```
RELEASE_CFLAGS=-g -O2 -march=armv6 -mfloat-abi=hard -mfpu=vfp
```

Der Kompilierungsvorgang wird nun mit

```
sudo make release
```

gestartet. Newcomer und Linux-Fremde bängen an dieser Stelle immer und hoffen, dass nun alles fehler- und abbruchsfrei durchläuft. In Problemfällen gilt: Google oder Linux-affine Bekannte sind deine Freunde! Hat alles

geklappt, wird das SvxLink-Paket mit Hilfe von

```
sudo make install
```

auf dem Raspberry Pi installiert.

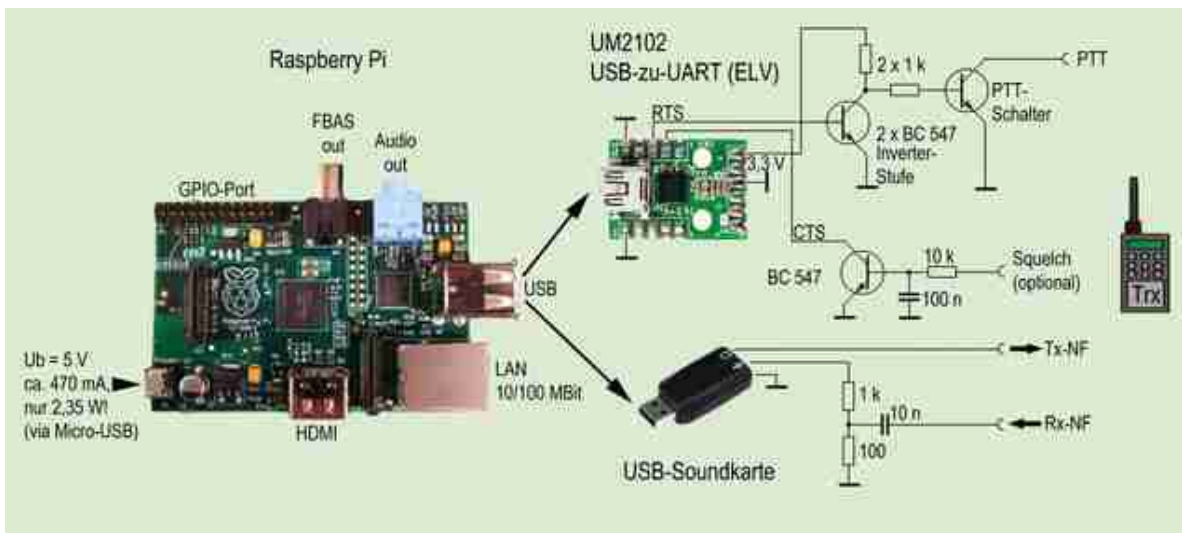
Blick auf die Hardware

Bevor wir mit der weiteren Konfiguration fortfahren, zunächst ein Blick auf die Hardware. Für ein Echolink-Gateway benötigt man einen Soundein- und Ausgang sowie die Möglichkeit der PTT-Tastung des angeschlossenen Senders. Über einen Soundausgang verfügt das Raspberry Pi von Hause aus, nicht jedoch über einen Eingang in Form von Line-in oder Mikrofon-Input. Man wird also zu einer USB-Soundkarte greifen, die bereits im einstelligen Eurobereich erhältlich ist. Kaufempfehlungen kann ich nicht aussprechen, dafür habe ich zu wenige zum Testen vorrätig. Als funktionierend hat sich bei mir eine USB-Soundkarte von Sennheiser herausgestellt, die vor einiger Zeit zusammen mit dem USB-Headset PC105 ausgeliefert wurde:



Die USB-Vendor-ID lautet: „1395:3556 Sennheiser Communications USB Headset“. Ein Internet-Wiki empfiehlt als weitere Alternativen eine Creative Sound Blaster Play!, eine Edirol UA-1A oder eine Logilink UA0053 [10].

Die PTT-Tastung erfolgt meist über eine serielle Schnittstelle, welche beim Raspberry Pi nicht mit den nötigen Handshakeleitungen auf die Standard-GPIO-Pinleiste gelegt ist. Erst seit der neuen Board-Revision des B-Modells (zu erkennen an den Bohr-/Befestigungslöchern und 512 MB RAM), ist unterhalb der GPIO-Pinleiste ein zweites Pin-Feld hinzugekommen, das weitere GPIO-Pins zur Verfügung stellt. Wie man daraus eine vollständige serielle Schnittstelle erstellt, beschreibt dankenswerter Weise DL8SCU auf [14]. Darüber kann die PTT-Tastung direkt erfolgen. Zu meiner Entwicklungszeit des Raspberry Pi's war diese Möglichkeit noch nicht bekannt, weshalb ich auf das USB-Modul UM2102 von ELV [11] zurückgegriffen habe, was vom Linux-Kernel automatisch erkannt wird. Soundkarte und das UM2102-Modul belegen fortan beide vorhandenen USB-Ports des Raspberry Pi's. Für eine Tastatur verbleibt ohne zusätzlichen USB-Hub kein Steckplatz mehr. Spätestens jetzt sollte man sich vom PC per SSH einloggen und ohne Tastatur weiterarbeiten oder eben einen Hub einsetzen.



Sound für SvxLink

Es wird Zeit, die USB-Soundkarte softwareseitig gangbar zu machen. Die folgenden Schritte sollten dabei für diverse Modelle gelten (Generic-Treiber). In der Datei /etc/modules binden wir den Treiber für diesen Hardware-Typ ein. Das gelingt durch Hinzufügen des Eintrags „snd-usb-audio“. Da der Onboard-Audioausgang des Raspberry Pi's nicht benötigt wird, sollte man die Zeile – sofern vorhanden – „snd-bcm2835“ mit einem „#“ vorweg auskommentieren oder löschen. Die zweite Änderung betrifft die Datei /etc/modprobe.d/alsa-base.conf. Sie ist für das Linux-Alsa-Soundsystem zuständig. Hier müssen wir ihm mitteilen, dass es angesteckte USB-Soundkarten ansprechen und nicht ignorieren soll. Die Zeile „options snd-usb-audio index=-2“ wird hinten auf „0“ geändert.

Eine Dritte Änderung hat infolge des fortentwickelten Linux-Soundsystems ALSA nun VK2MEV ins Spiel gebracht: Man passe die Datei /etc/asound.conf an. Im Original sieht sie wie folgt aus:

```
pcm.mmap0 {
type mmap_emul;
slave {
pcm "hw:0,0";
}
}
```

```
#pcm.!default {
# type plug;
# slave {
# pcm mmap0;
# }
# }
```

Ihr Inhalt wird nun wie folgt geändert:

```
pcm.!default {
type plug
slave {
channels 1
pcm "hw:0,0"
}
}
pcm.low {
type plug
slave {
```



```
pcm "hw:0,0"  
}  
}
```

Mit Hilfe dieser Einträge gelingt es, selbst 5-€-USB-Soundkarten in Betrieb zu nehmen. Am besten steckt man jetzt die zuvor in Tests als funktionierend gefundene USB-Soundkarte und das UM2102-Modul an und vollzieht der Gewissheit halber einen Neustart, woraufhin die USB-Soundkarte nun als Alsa-Device „0“ mit Hilfe der Abfrage `cat /proc/asound/cards`

mit dem Kartenindex „0“ angezeigt werden sollte.

SvxLink lernt laufen

Für die Parametrierung des SvxLink-Pakets müssen wir im Wesentlichen zwei Dateien in den Verzeichnissen `/etc/svxlink` und `/etc/svxlink/svxlink.d` anpassen. Alle Parameter hier zu beschreiben, würde den Rahmen sprengen, außerdem gibt es dafür im Internet ausführliche Beschreibungen, z.B. [12]. Deshalb hier in Kürze, worauf es mindestens ankommt (Änderungen unterstrichen).

Die Datei `ModuleEchoLink.conf` bestimmt Logindaten – ja, einen Echolink-Account sollte man bereits vorher einmal erstellt haben –, und Connect-Text. Neuere SvxLink-Versionen bieten hier zudem den Parameter `USE_GSM_ONLY=1`, den man möglichst auf 1 setzen sollte. SvxLink bietet auf schnelleren Systemen den Vorteil einer besseren Übertragungsqualität: Wird auf der Gegenseite ebenfalls ein SvxLink-System erkannt, wird vom GSM- auf den SPEEX-Codec gewechselt. Dieser ist jedoch für den Raspberry Pi zu Rechenaufwändig, weshalb es zu Aussetzern kommt. Der Parameter belässt das System im Ressourcen-sparsamen GSM-Codec.

Auszug `ModuleEchoLink.conf`

```
[ModuleEchoLink]  
NAME=EchoLink  
ID=2  
TIMEOUT=60  
#ALLOW_IP=192.168.1.0/24  
#DROP_INCOMING=^() $  
#REJECT_INCOMING=^() $  
#ACCEPT_INCOMING=^(.*) $  
#REJECT_OUTGOING=^() $  
#ACCEPT_OUTGOING=^(.*) $  
SERVER=europe.echolink.org  
CALLSIGN=IhrCall  
PASSWORD=IhrPasswort  
SYSOPNAME=IhrOpName  
LOCATION=[Svx] Ihr QTH & QRG  
MAX_QSOS=3  
MAX_CONNECTIONS=3  
LINK_IDLE_TIMEOUT=300  
USE_GSM_ONLY=1  
DESCRIPTION=SSvxLink Node,\n"  
"Weitere Infos zum Node\n"
```

Die Möglichkeiten für Einstellungen in der `svxlink.conf` sind dagegen umfangreicher [12]. Hier kann man u.a. auch einstellen, ob das System im Vox-Betrieb arbeiten soll. Recht viel Zeit musste ich mit der korrekten Angabe des Audio-Device verbringen. Viele gefundene Konfigurationsanleiten gaben an, dass man „`AUDIO_DEV=alsa:plughw:0`“ angeben sollte, was anfangs zum Kernel-Panic führte. Erst die Angabe von „`alsa:default`“ für den Rx-Weg und „`alsa:hw:0`“ für den Tx-Weg führte zum Erfolg. Für weitere Einstellungen möchte ich zum Experimentieren anregen. Gerade beim VOX-Betrieb für den Sende-/Empfangsfall lassen sich die vordefinierten

Werte noch optimieren. Wer einige Millisekunden Umschaltzeit sparen und damit letztendlich einen flüssigeren QSO-Verlauf ermöglichen will, wird den Squelch des Trx hardwareseitig auswerten und mit dem angedeuteten Schaltungszweig auf den CTS-Pin des UM2102 gehen.

Und auch wenn SvXlink nicht für 16000 kHz kompiliert wurde, ist in einigen Fällen noch die Variable CARD_SAMPLE_RATE anzupassen. Hier muss man mit dem Wert spielen, solange, bis SvXLink anstandslos in Verbindung mit der eingesetzten Soundkarte startet.

Auszug SvXlink.conf

```
CARD_SAMPLE_RATE=48000
```

```
...
```

```
[SimplexLogic]
```

```
TYPE=Simplex
```

```
RX=Rx1
```

```
TX=Tx1
```

```
MODULES=ModuleEchoLink
```

```
CALLSIGN=IhrCall
```

```
...
```

```
[Rx1]
```

```
...
```

```
AUDIO_DEV=alsa:default
```

```
...
```

```
SERIAL_PORT=/dev/ttyUSB0
```

```
...
```

```
[Tx1]
```

```
...
```

```
AUDIO_DEV=alsa:hw:0
```

```
...
```

```
PTT_PORT=/dev/ttyUSB0
```

SvXLink lernt sprechen

Jetzt bringen wir SvXLink das Sprechen bei. Dafür laden wir uns z.B. von [13] Sprachdateien herunter. Im Archiv enthalten sind solche für diverse Samplingraten. Im Sinne der CPU-Auslastung des Raspberry Pi's kommt die 8-kHz-Variante zum Einsatz. Über Funk klingen meiner Meinung nach selbst 8k-Samples durchaus akzeptabel. Ausgepackt werden die Dateien nach /usr/share/svXlink/sounds/en_US, sodass sich darunter die im Paket enthaltenen Unterverzeichnisse „Core“, „Default“, „EchoLink“ usw. befinden.

Fertig zum Start!

Nach der Installation wollen wir endlich Früchte dessen ernten, was wir gesät haben. Auf der Konsole starten wir SvXLink. Nachfolgend ergibt sich durch Eingabe von svXlink

das Gewünschte (je nach Konfiguration der Module):

```
pi@raspberrypi $ svXlink
```

```
SvXLink v0.13.0 (Jul 11 2012) Copyright (C) 2011 Tobias Blomberg / SM0SVX
```

```
SvXLink comes with ABSOLUTELY NO WARRANTY. This is free software, and you are  
welcome to redistribute it in accordance with the terms and conditions in the  
GNU GPL (General Public License) version 2 or later.
```

```
Using configuration file: /etc/svXlink/svXlink.conf
```

```
Starting logic: SimplexLogic
```

```
Loading module ModuleHelp into logic SSimplexLogic"
```

```

Module Help v0.7.0 starting...
Loading module ModuleParrot into logic SSimplexLogic"
Module Parrot v0.7.0 starting...
Loading module ModuleEchoLink into logic SSimplexLogic"
Module EchoLink v0.10.1 starting...
Loading module ModuleTclVoiceMail into logic SSimplexLogic"
Module Tcl v0.3.0 starting...
Loading module ModuleMetaInfo into logic SSimplexLogic"
Module MetaInfo v0.0.2 starting...
Event handler script successfully loaded.
EchoLink directory status changed to ON
— EchoLink directory server message: —
EchoLink Server v2.5.9997
ECHOEC2-3: Herndon, VA USA

```

Was man noch herausfinden muss, ob die Soundlautstärke für Empfang und Senden ausreichend ist. Ich empfehle eine zweite Konsole, sprich man loggt sich ein zweites Mal auf dem Raspberry Pi per SSH ein und ruft das Mischpult auf der Konsole auf mit

alsamixer



Hat man alle Einstellungen getroffen, gilt es diese für alle künftigen Systemstarts zu speichern. Das geht nicht automatisch, sondern mit

```
sudo alsactl store
```

Mit <Strg+X> kann man SvxLink fürs erste beenden. Der letzte Schritt für die Konfiguration betrifft die Schönheits-Optik. Damit das System nach einem Neustart SvxLink startet, fügen wir z.B. in der /etc/rc.local mit Sudo-Rechten

folgende Zeilen vor dem „exit 0“ ein:

```
# Starte Svxlink
sleep 10
svxlink -daemon
sleep 5
echo „SvxLink gestartet ...“
```

Der Sleep-Befehl versetzt den Rechner für kurze Zeit in Warteposition. Dies ist nötig, damit kurz nach dem Start alle Systemparameter am Laufen sind, ehe SvxLink seinen Dienst aufnehmen kann. Wem der Autostart per rc.local unkonventionell erscheint, kann SvxLink alternativ über die Runlevel starten lassen.

Troubleshooting: Was tun, wenn der Sound nicht will?

Möglicherweise werden Sie Probleme mit dem USB-Sound bekommen: Entweder funktioniert nur die Sound-Ausgabe, aber nicht die Soundeingabe; oder die Soundausgabe funktioniert und die Eingabe ist mit softwarebedingten Störgeräuschen behaftet. Im Gegensatz zum Zeitpunkt der Erstveröffentlichung dieser Anleitung im Oktober 2012 ist das heutige Softwarepaket für den Raspberry Pi weiterentwickelt worden, was die Nachbausicherheit seitdem deutlich erhöht hat. Selbst 3-€-Soundkarten bekommt man eigentlich zum Laufen. Grundsätzlich besteht ein Zusammenspiel mit dem Linux-Kernel und der Firmware des Raspberry Pi's. Dies sind auch genau die Schrauben, woran man vermutlich drehen kann, wenn's klemmt. Weiterhin ist ein Austausch per Forum unter Gleichgesinnten immer hilfreich [5]. Als weitere Alternativen bleibt nur zu empfehlen, Soundkarten einfach durchzuprobieren und auch immer die Änderungen an der Raspberry-Pi-Firmware im Auge zu behalten.

Zusammenfassung und Tipps

Da wir doch einiges an Aufwand hineingesteckt haben, empfehle ich, von der Karte am PC mit dem Win32Imager ein Backup-Image (Funktion „Read“) zu erstellen. Generell sagt man SD-/Flashkarten nur eine gewisse Anzahl an Schreib-Zugriffen nach; sollte eine Karte also mal ausfallen, hat man eine Systemkopie.

Das Raspberry Pi wird ohne Gehäuse geliefert. Reichelt bietet z.B. unter der Bezeichnung TEK-BERRY ein einfaches Plastikgehäuse an.

Bei einem Board gab es ein Problem mit der Stromversorgung, die auf eine offenbar defekte Mikro-USB-Buchse zurückzuführen war. Das System lief instabil, die Netzwerkverbindung wurde immer wieder verloren und die LEDs flackerten leicht im Takt der CPU-Last. Mit dem Oszilloskop wurde eine Defekte Buchse festgestellt. Ein Lösungsansatz: Auf dem Board befinden sich zwei Testpunkte, über die man GND und +5 V direkt einspeisen kann (vgl. unbedingt den Schaltplan wegen Möglichkeit der Falschpolung). Eine weitere Möglichkeit besteht darin über die GPIO-Pinleiste die Spannungsversorgung direkt einzuspeisen, auch hierzu sollte der Stromlaufplan des Boards konsultiert werden.

Durch den Einsatz eines modernen Embedded-Systems ist ein kompaktes und vor allem stromsparendes Echolink-Gateway entstanden. Die CPU-Auslastung liegt mit 30 % im Durchschnitt und bietet noch etwas Luft nach oben. Der Stromverbrauch dieser Komponente konnte von ca. 60 W auf <3 W gesenkt werden. Wer zum LötKolben greift, kann auch das Board nochmals modifizieren: Im Spannungs-Eingangsbereich sitzt ein Linearregler von 5 V auf 3,3 V. Diesen könnte man durch einen Schaltregler ersetzen und sicher nochmals einige mA sparen. So geschehen bei einem Projekt, bei dem der Raspberry Pi auf einem Wetterballon mitflog und das Energiebudget limitiert war.



Der folgende Screenshot zeigt nochmals die CPU-/Systemlast. Im Gegensatz zu ersten Installationsversuchen läuft das System über einen recht langen Zeitraum stabil:

```
pi@raspberrypi ~ $ uptime
 17:10:25 up 51 days, 18:16,  1 user,  load average: 0.54, 0.33, 0.29
pi@raspberrypi ~ $
```

Echolink bzw. SvxLink ist nicht die einzige Amateurfunkanwendung - Hamnet, APRS, oder allgemeine Steuerungsaufgaben sind möglich. Die Redaktion CQ DL freut sich auf Bauvorschläge – wir sind gespannt! Bedanken möchte ich mich bei Frank Köditz, DD9UG, und Michael Hartje, DK5HH, für den Gedankenaustausch zu Schaltungstechnik und SvxLink.

73 und viel Erfolg beim Stromsparen de Stefan, DH5FFL

Links

- [1] Markus Heller, DL8RDS, hat die Installation eines Echolink-Systems mit der SvxLink-Software auf dem Beagleboard beschrieben: www.dl8rds.de/index.php/A_SvxLink_Server_based_on_the_BeagleBoard
- [2] Bezug über element14 (Farnell): www.element14.com/community/groups/raspberry-pi
- [3] Bezug über R & S: <http://uk.rs-online.com/web/generalDisplay.html?id=raspberrypi>
- [4] Original SvxLink-Seite des Entwicklers SMØSVX: <http://svxlink.sourceforge.net>
- [5] Deutschsprachige SvxLink-Seite (diese hier :-)) : www.svxlink.de
- [6] Hinweise zu SvxLink, auf der Webseite der Firma KCT: www.firma-kct.com/cgi-bin/show.pl?page=SvxLink-Tips
- [7] Download der Raspbian-Wheezy-Distribution für den Raspberry Pi: <http://www.raspberrypi.org/downloads>
- [8] SSH-Client Putty: www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- [9] Rpi-Updater von Hexxeh: <https://github.com/Hexxeh/rpi-update>
- [10] Empfohlene Hardware für das Raspberry Pi, z.B. auch Soundkarten: http://elinux.org/RPi_VerifiedPeripherals
#USB_Sound_Cards
- [11] Mini-USB-Modul UM2102, Best.-Nr. 68-09 18 59, 5,95 €, ELV Elektronik AG, Maiburger Straße 29-36, 26789 Leer, www.elv.de
- [12] Ausführliche Beschreibung der Konfigurationsdatei svxlink.conf: <http://svxlink.sourceforge.net/man/man5/>

svxlink.conf.5.html

[13] Synthetische OpenSource-Sprachdateien: <http://svxlink.de/download/stimmen-mary.tar.bz2>

[14] Blog von DL8SCU, Erweiterung zur vollständigen seriellen Schnittstelle auf dem Raspberry Pi: <http://dl8scu.wordpress.com/2013/02/10/are-you-lucky-no-usb-to-serial-converter-needed/>

2.8 Oktober

Startskripte (2012-10-04 20:39)

Die meisten Distributionen (und jetzt sogar auch Debian) unterstützen den systemd, einen Dienst der über das Starten bzw. über gestartete Applikationen wacht. Im Folgenden eine Kurzanleitung, wie Sie diesen Dienst nutzen können um SvxLink automatisiert und überwacht zu betreiben.

Die Applikation svxlink sollte nicht als root laufen, das ist absolut nicht notwendig! Legen Sie daher eine Gruppe svxlink und den User svxlink an, führen Sie als root aus:

```
groupadd svxlink
useradd -G svxlink -g svxlink svxlink
```

Vergeben Sie dem Nutzer ein Passwort, bitte nicht so etwas wie äbc123”!

Die neusten openSUSE-Versionen nutzen den systemd um das Starten und Stoppen der Applikationen zu realisieren. Startskripte in /etc/init.d sind dann nicht mehr erforderlich. Die Einrichtung ist sehr einfach. Erstellen Sie im Verzeichnis eine Datei svxlink.service mit folgendem Inhalt:

```
[Unit]
```

```
Description=SvxLink Server
```

```
After=network.target sound.target
```

```
[Service]
```

```
Restart=always
```

```
KillMode=process
```

```
ExecStart=/usr/bin/svxlink -config=/etc/svxlink/svxlink.conf -logfile=/var/log/svxlink
```

```
RestartSec=5
```

```
TimeoutSec=5
```

```
User=svxlink
```

```
Group=svxlink
```

```
Environment="HOME=/home/svxlink"
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Aktivieren Sie das Starten des SvxLink-Servers mit:
systemctl enable svxlink.service

und starten Sie svxLink mit:
`systemctl start svxlink.service`

Der systemd überwacht die Funktion dieses Dienstes und startet ihn im Bedarfsfall neu.

Und hier noch ein Skript für Debian, dieses Skript bitte in svxlink umbenennen und nach /etc/init.d/ kopieren:

Download SvxLink-Startskript für Debian

Dieses Skript muß ausführbar sein:

```
chmod 755 /etc/init.d/svxlink
```

und dann noch aktiviert werden, damit es beim Systemstart automatisch ausgeführt wird:

```
sudo update-rc.d svxlink defaults
```

Wieterführende Informationen zu Diensten kann man hier finden <http://wiki.ubuntuusers.de/Dienste>

3. 2013

3.1 Mai

Compileroptionen für ARM, Geode (und andere Prozessoren) (2013-05-29 14:53)

Mit einigen wenigen Compileroptionen läßt sich die durch SvxLink hervorgerufene CPU-Last erheblich reduzieren. Ändern Sie hierfür den Parameter RELEASE_CFLAGS in der Datei makefile.cfg im SvxLink-Source-Verzeichnis wie folgt:

Raspberry Pi (ARM-Architektur):

```
RELEASE_CFLAGS=-g -O2 -mfloat-abi=softfp -mfpu=vfp -mcpu=native
```

Raspberry Pi (ARM-Architektur) unter Raspbian "wheezy" (Hinweis von Uwe/DC5PI):

```
RELEASE_CFLAGS=-g -O2 -march=armv6 -mfloat-abi=hard -mfpu=vfp
```

Futro A240 (Geode):

```
RELEASE_CFLAGS = -g -O2 -march=geode -mtune=generic
```

Diese Angaben wurden experimentell ermittelt und erheben keinen Anspruch auf Vollständigkeit!
Weitere Informationen zum gcc und dessen Compileroptionen können Sie hier finden:

ARM-Prozessor: <http://gcc.gnu.org/onlinedocs/gcc/ARM-Options.html>

Geode (oder andere): http://gcc.gnu.org/onlinedocs/gcc/i386-and-x86_002d64-Options.html

3.2 Oktober

SvxLink mit Tetra-DMO-Repeater (Tetra im Amateurfunk) (2013-10-29 12:41)

Hier ist die aktuelle ISSI-Liste für die Region Halle/Weißenfels (Stand 20.2.2015): [Download](#)

Seit dem 16.10.2013 arbeitet in Halle/Saale ein TETRA-DMO-Repeater mit SvxLink unter dem Amateurfunkrufzeichen DM0SVX. Für dieses kleine Projekt wurde SvxLink um eine TetraLogic auf der Basis einer RepeaterLogic erweitert und damit begonnen, Standard-PEI-Funktionalitäten in diese Logik aufzunehmen um die angeschlossene Hardware entsprechend ansteuern zu können. Ziel ist ein Gateway HFEchoLink mit den bekannten Funktionalitäten bereitzustellen.



Standort DM0SVX

Stationsparameter DM0SVX (DMO-Repeater):

Parameter

Wert / Beschreibung

Standort

Halle/Saale

GPS-Position

51°29.49'N und 11°57.35'E in JO51XL

Frequenz

430.375 MHz (DMO Type 1A)

Bandbreite

25 kHz

Modulationsart

$\pi/4$ -DQPSK

Übertragungsverfahren

TDMA, Zeitschlitzverfahren mit 3 genutzten Zeitschlitzten

138

MCC/MNC/SSI
262/16383/23401

TEA
Class 1 (clear mode)

EIRP
15 W

Antenne
X50N

Höhe NN / GND
76,1m / 39m

TRX
Motorola MTM5400 mit DMO-Repeater-Lizenz

Software
SvxLink mit Tetra-Extension

EchoLink Node-ID
890492

Betriebssystem
openSuSE 12.3

Sysops
Tino / DM2NT und Adi / DL1HRC

Kontakt
dm0svx {ät } svxlink.de

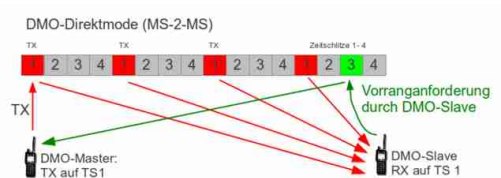
Diese Relaisstelle wird momentan im Experimentierstadium betrieben was heißt, dass sich ständig etwas ändern kann.

Allerdings sind schon einige Funktionen verfügbar wie z.B. Positioning (APRS) und SDS-Empfang. Die Userdaten der Endgeräte und deren aktuellen Betriebsparameter werden in Struct's gespeichert und stehen damit für weitere SvxLink-interne Operationen zur Verfügung.

Ein DMO-Repeater sendet periodisch (ca. jede 10 Sekunden) eine Identifikation aus, die von den MS (mobile

station) erkannt werden. Im Falle eines Rufaufbaus erfolgt die Verbindung dann über den Repeater und nicht auf direktem Weg. Prinzipiell läßt sich ein an den vom GSM-Betrieb her bekannten "hand-over" in rudimentärer Form realisieren, wenn die DMO-Repeater in Scanlisten auf den MS gespeichert werden. Damit entfällt ein händisches Umschalten/Suchen von diversen Repeatern bzw. ein Mitführen von Relaislisten. Das ist aber noch Zukunftsmusik.

Beim Direktmode (ohne DMO-Repeater) wird vom DMO-Master der Zeitschlitz 1 (TS1) zum Senden genutzt. Der oder die DMO-Slaves haben in Zeitschlitz 3 (TS3) die Möglichkeit eine Vorranganforderung auszusenden um dem Master einen Sprechwunsch zu signalisieren.



DMO (MS-2-MS-Betrieb)

Im Gegensatz zum DStar-DPRS wird bei der Aussendung von Positionsbacken auch nicht der über die Station durchgeführte Funkverkehr gestört und durch das TDMA-Verfahren ist praktisch ein Relaisbetrieb über eine Simplex-Frequenz möglich.

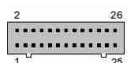
OM's die gern mitarbeiten möchten werden gebeten sich mit den Sysops in Verbindung zu setzen. Diese können u.U. bei der Beschaffung von Technik behilflich sein.

ACHTUNG: Es können nur TETRA-Geräte verwendet werden, z.B. Motorola MTP8x0, MTH700, MTM800, MTM5400, CEP400, Selex Puma T3

Bitte in jedem Fall vor dem eBay-Kauf nachfragen!

Anschlußbelegung Erweiterungsleiste MTM5400 (für SvxLink)

Die Verbindung SvxLink-PC mit dem MTM5400 erfolgt über ein aktives Datenkabel (Motorola-Bestellnummer: PMKN4104) bei dem für die NF-Übertragung noch die Audioleitungen (TX-Audio und RX-Audio) nachgerüstet werden müssen.



MTM5400 AZubehöranschlußbuchse (u.a. PEI), Blick auf Geräterückseite



Aktives Datenkabel MTM800 FuG/MTM5400 vor der Erweiterung der Audioleitungen

Pin

Funktion

Bedeutung

1

UART1_TXD / USBx_D+

USB 1.1 - Standard-Host RS232 oder UART2 - Alternative Einstellung

2

UART1_RXD / USBx_D-

3

UART1_RTS / USBx_VBUS

4

GND_USBx

5

Einadrig

1-adrigger Standardanschluss (über 2,2k auf 5V)

6

KEYFAIL/FLASH

Schlüsselladung (über 10K auf 5V) Flash-Eingang (>10 V löst Flashmode aus)

7

SWB +

A+-Spannung (begrenzt auf 14V) mit 1A-Strombegrenzung

8

GND_MAIN

Haupt- und Leistungsmasse

9

LAUTSPRECHER-

Lautsprecher (PA) Ausgang - (KEINE ERDUNG!)

10

LAUTSPRECHER+

Lautsprecher (PA) Ausgang +

11

TX_AUDIO

TX-Audioeingang

12

GND_ANA

Hauptaudiomasse

13

MIC1 / EXT_MIC

Ext. Mikrofoneingang/MIC1 für Geräuschminimierung dualer Mikrofoneingang

14

RX_AUDIO

RX-Audioausgang

15

MIC2

MIC2 für Geräuschminimierung dualer Mikrofoneingang

16

142

GND_MIC
Masse (für MIC)

17
EXTERNAL_PTT
PTT-Eingang (über 4,7k auf 5V)

18
UART2_DTR/USBy_ID
RS232 oder UART1/UART2 DTR/ 2. USB 2.0

19
HOOK_PA_EN
HOOK_PA_EN Eingang (oder der programmierbare 5V-GPIO)

20
UART2_TXD/USBy_TX
RS232 oder UART2 TXD/2. USB 2.0 (OTG) D+

21
UART2_RTS/USBy_VBUS
RS232 oder UART2 RTS/2. USB 2.0 (OTG) VBUS- 100 mA

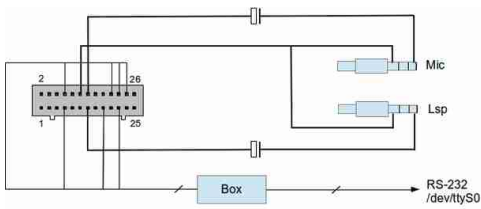
22
UART2_RXD/USBy_RX
RS232 oder UART2 TXD/2. USB 2.0 (OTG) D-

23
Notruf
Eingang für Durchsagen in Notfällen (gezogen über 24,9k auf A+) – zum Einschalten auf Low ziehen

24
UART_CTS
RS232 oder UART1/UART2 CTS-Eingang

25
Zündung
Zündungseingang (über Serie 15K): zum Einschalten > 5V ziehen

26
EXTERNER ALARM
Externer Alarmeingang (gezogen über 4,7k auf A+)

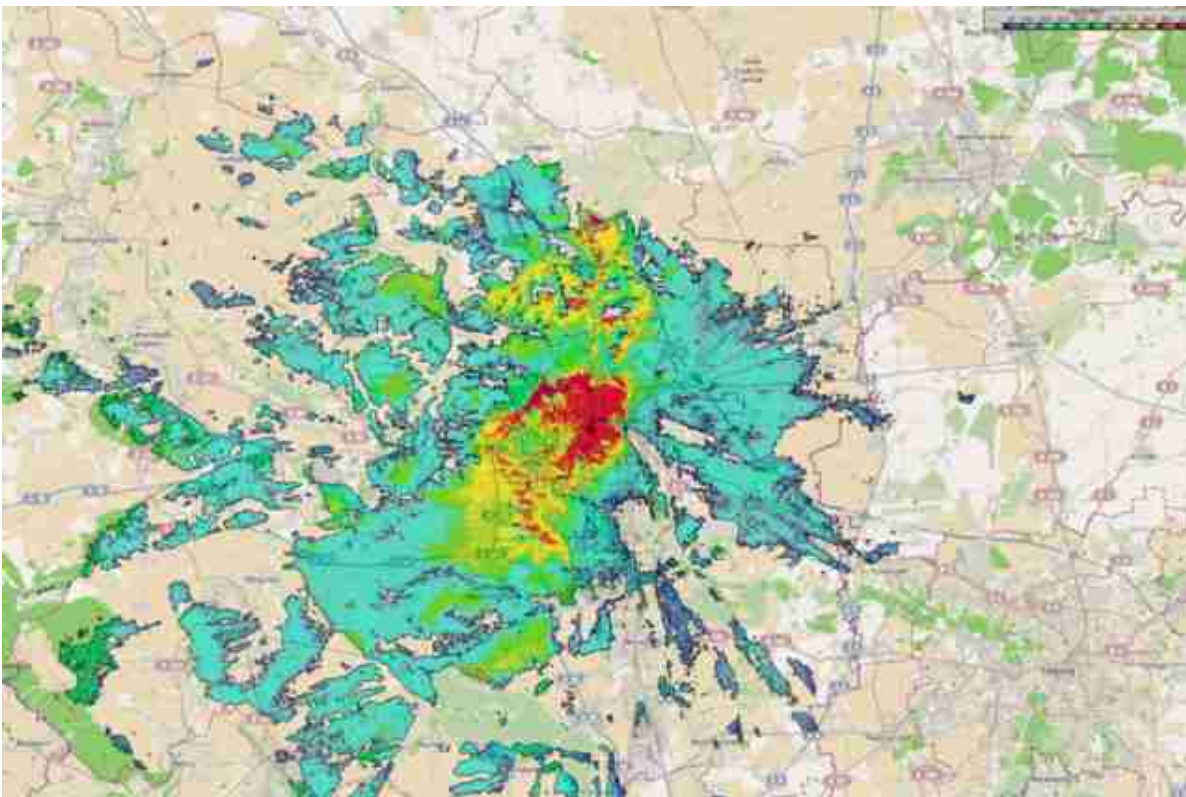


MTM5400-Adapter zum Anschluß an den SvxLink-Server

Warum eigentlich TETRA und nicht z.B. DStar?

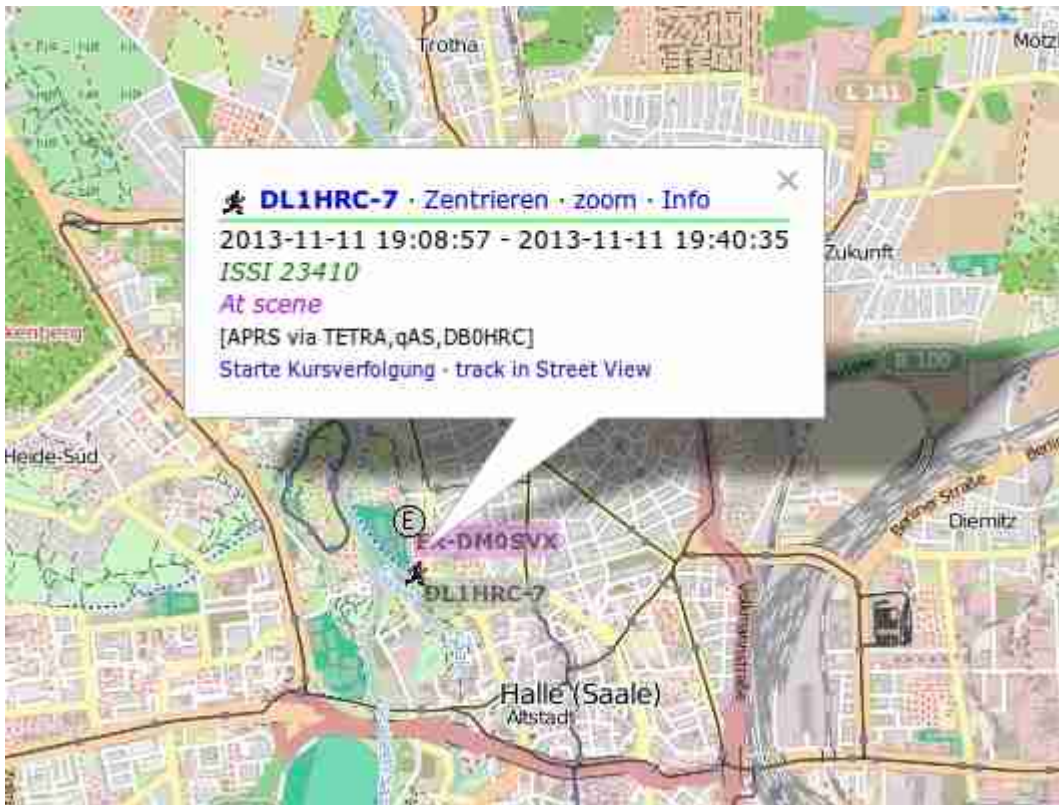
- 1) Amateurfunk ist Experimentalfunk und man muß immer mal was Neues ausprobieren
- 2) wesentlich mehr Möglichkeiten und Funktionen stehen zur Verfügung
- 3) Repeaterfunktionalität auf einer Simplexfrequenz(!)
- 4) Geräte sind für uns wesentlich günstiger zu beschaffen, DStar läuft i.A. auf einen Neukauf hinaus
- 5) TETRA=viele Hersteller, ein Standard (EADS,Selex,Motorola,Sepura,Hytera,R &S,...)
- 6) proprietärer DStar-Voice-Codec verhindert eigene Entwicklungen, der höchstkommerzielle TETRA-Standard ist letztendlich wesentlich offener als DStar(!)
- 7) Positioning (APRS) störungsfrei möglich
- 8) etwas bessere Sprachqualität

Ausbreitungsprognose (Handfunkgerät Feldstärke mindestens S6):



Prognose, Einzugsbereich von DM0SVX (entspricht etwa Erreichbarkeit mit Handfunkgerät im Auto)

Positioning durch Auswertung von LIP-SDS ist bereits implementiert.



Status und Position durch Auswertung der LIP-SDS's

Weitere Informationen zu dieser interessanten SvxLink-Erweiterung folgen...

z.B. auf der Interradio 2013 in Hannover und/oder auf dieser Webseite.

Am 30.11.2013 gibt es ab 13:30 Uhr einen Vortrag, der u.a. auch diese Implementierung erwähnt, siehe Vortragsprogramm Interradio 2013

Hinweis: Das Kartenmaterial entstammt dem OpenStreetMap-Projekt.

Weitere Informationen und Links:

Terrestrial Trunked Radio

MTP850

MTM800

MTM5400

Selex Puma T3

Stationsparameter DMOSVX (APRS-Digi):

Parameter

Wert / Beschreibung

TRX

Motorola MC Micro

EIRP

5 W

Antenne
X50N

Software
aprx

Funktion
Fill-in, IGate

Betriebssystem
openSuSE 12.3

geplant für 2014: POCSAG 439.9875MHz

Audioprobleme beim Raspberry Pi (2013-10-19 23:14)

Falls sich beim Raspberry Pi Audioprobleme zeigen sollten, so kann man den Tip von Wilm (DL4OCH) ausprobieren. Die Ursache liegt vermutlich in Problemen beim USB2.0 und bewirkt dass Audioausgaben gedropped werden (Bitttrigärscheinen).

mögliche Lösung:
Downgrade auf USB1.1

Schritte:
sudo nano /boot/cmdline.txt
dwc_otg.speed=1

Damit wird der USB-Port auf USB 1.1 gezwungen. sudo reboot nicht vergessen.

HowTo (2013-10-22 07:37)

Modul EchoLink:

Wie kann ich bei einer eingehenden EchoLink-Verbindung einen Rufton aussenden um beispielsweise ein Relais zu öffnen?

Erstellen Sie eine tcl-Datei (z.B. el-beep.tcl) nach dem lokale-Prinzip mit folgendem Inhalt:

```
namespace eval EchoLink {
  proc remote_connected {call} {
    playTone 1750 100 1000;
    playMsg "connected";
    spellEchoLinkCallsign $call;
    playSilence 500;
  }
}
```

Starten Sie SvxLink neu.

Repeater-/Simplexbetrieb:

Wie kann die Tonhöhe/-länge des Roger-Pieps geändert werden?

Erstellen Sie eine tcl-Datei (z.B. rogerbeep.tcl) nach dem lokale-Prinzip mit folgendem Inhalt:

```
namespace eval Logic {
  proc send_rgr_sound {} {
    variable sql_rx_id;

    playTone XXXX 500 YYYY; # hier XXXX durch die neue
                           # Tonhöhe ersetzen (Bereich: 300-3000)
                           # und YYYY durch die neue Tonlänge in ms

    playSilence 200;

    for {set i 0} {$i < $sql_rx_id} {incr i 1} {
      playTone 880 500 50;
      playSilence 50;
    }
    playSilence 100;
  }
}
```

Starten Sie SvxLink neu.

Wie kann man periodische Aussendungen (Zeit,Call,...) auf einem Link unterbinden während eine EL-Station bzw. eine lokale Station sprechen?

Erstellen Sie eine tcl-Datei (z.B. stop_announce.tcl) nach dem lokale-Prinzip mit folgendem Inhalt:

```
namespace eval EchoLink {
  variable el_rx 0;
  proc is_receiving {isrec} {
    set el_rx $isrec;
  }
}

namespace eval Logic {
  variable rx_state 0;
```

```

proc send_long_ident {hour minute} {
    if {$EchoLink::el_rx != 0 || $rx_state != 0} {
        exit;
    }
    # kopieren sie den Inhalt der Prozedur send_long_ident der
    # Original-Logic.tcl bzw. Ihre Änderungen hier herein
    #
}

proc sqlch_open {rx_id is_open} {
    variable sql_rx_id;
    set sql_rx_id $rx_id;
    set rx_state $is_open;
}
}

```

Starten Sie SvxLink neu. Achten Sie darauf, dass Sie wahrscheinlich diese Prozeduren schon vorher an Ihre Bedürfnisse angepasst haben und Sie diese Änderungen hier nachziehen müssen.



4. 2014

4.1 Januar

Einfaches EchoLink-Logbuch (2014-01-16 13:55)

Um automatisch ein kleines und einfaches Logbuch der EchoLink-Verbindungen erstellen zu lassen kann ein kleines Perl-Skript [1] genutzt werden. Dieses Skript liest das SvXLink-Log aus und erstellt daraus eine Logbuchdatei im konfiguriertem Verzeichnis.

Bedingungen:

- auf Ihrem System muß perl installiert sein
- Sie müssen SvXLink so betreiben, dass es die Ausgaben in ein Logfile schreibt. Wenn Sie SvXLink auf der Konsole starten werden die Ausgaben NICHT in das SvXLink-Logfile geschrieben

Installation:

- 1) Laden Sie das Skript herunter und installieren Sie es unter z.B. /usr/bin
- 2) machen Sie das Skript ausführbar mit `chmod 755 el_logbook.pl`
- 3) Ändern Sie in der `svxlink.conf` den Parameter `TIMESTAMP_FORMAT="%d. %m. %Y %H: %M: %S"`, damit die Datums/Zeit-Einträge in deutscher Schreibweise erfolgen
- 4) im Skript selbst können die Pfade für das `svxlink-log` und das Logbuch festgelegt werden
- 5) Wenn Sie eine automatische Erstellung des Logbuchs wünschen, sollten Sie entweder einen Eintrag in der `crontab` vornehmen oder `logrotate` nutzen

a) mittels `cron`:

in `/var/spool/cron/tabs/svxlink` einen Eintrag hinzufügen, z.B.:

```
0 0 * * * /usr/bin/el_logbook.pl 1>/dev/null 2>/dev/null
```

Damit erfolgt die Erstellung jeden Tag um 0Uhr.

b) mittels `logrotate`:

Die bessere und sichere Variante ist die mittels `logrotate`. Dann wird immer kurz vor dem Rotieren der SvXLink-Logdatei ein Logbuch erzeugt und somit gehen keine Einträge verloren! Hier ein Beispiel für einen `logrotate`-Eintrag (`/etc/logrotate.d/svxlink`):

```
/var/log/svxlink {
    missingok
    notifempty
    daily
    create 0644 svxlink daemon
    prerotate
        /usr/bin/el_logbook.pl
    endscrip
    postrotate
        killall -HUP svxlink
    endscrip
}
```

Downloads:

[1] EL-Logbuch-Skript (Perl erforderlich)

4.2 Februar

Bootfähiges Image Voyage-Linux (incl. SvxLink) für ThinClients (2014-02-28 15:19)

Michael/DK5HH hat ein bootfähiges Image erstellt, welches auf eine CF-Karte kopiert werden kann. Die kann dann in ein Futro A2xx eingebaut werden und nach dem Booten des Linuxsystems ist man in Besitz eines vorinstallierten und vorkonfigurierten SvxLink's. Nachfolgend die Anleitung von Michael/DK5HH, bei Problemen und Fragen bitte direkt mit Ihm Verbindung aufnehmen.

Bitte auch die Ergänzungen zum Image lesen!!!!

Liebe OM's,

zu meinen Anmerkungen auf der Sysop-Tagung in Hannover nun der Link auf eine von mir zusammengestellte bootfähige Flashkarte. Die gesamte Flashkarte ist mit bzip2 kompaktiert.

<http://svxlink.de/download/voy-092-svx.img.bz2>

Die Datei voy-092-svx.img.bz2 hat (559977256 Bytes)

dann die Datei entpacken - das ergibt eine Datei mit 1,8 GByte

Diese Datei wird als ganzes auf die Flashkarte geschrieben (diese sollte etwa 2 GB haben)

Ich habe eine solche krumme Größe gewählt, die auf jeden Fall noch in eine 2GB-Karte passt.

Äußerste* *Vorsicht*!!! für die folgenden Schritte (erst einmal durchlesen!)

Dazu gibt es mehrere Methoden:

- unter winxp ist das win32DiskImager <https://launchpad.net/win32-image-writer>
- unter Linux ist das "dd"also

```
dd if=voy-092-svx.img of=/dev/sdx bs=1M
```

```
/dev/sdx bitte entsprechend wählen - Achtung nicht /dev/sdx1 sondern /dev/sdx (x ersetzen!)
```

Mit beiden Kommandos wird alles überschrieben! Es entsteht ein neuer Bootsektor mit neuer Partitionstabelle

Das Image sollte auf Futro A2xx booten und normal funktionieren.

Es ist leider nicht für mich möglich gewesen, weitere Rechner zu prüfen. Ich habe einen sehr neuen Kernel (3.10 auf der Karte) verwendet, so dass wohl alle älteren Netzwerkkarten sofort angesprochen werden sollten.

Das Programm svxlink ist in der neusten Version (von heute) übersetzt worden. Ich habe dann die Sounds und die Konfiguration von DBOHFT vollständig übernommen.

Informationen sind auf svxlink.de

Eure individuelle Anpassung sollte mindestens sein

1. Rufzeichen von DB0VVV ändern (/etc/svxlink/svxlink.conf)
2. Sound DB0HFT.wav suchen und mit eigener Begrüßung ersetzen (16 kHz, mono, 16Bit) steht in (/usr/share/svxlink/sounds/de_DE/LocalAudio/DB0HFT.wav)
3. Module zB Echolink aktivieren (in /etc/svxlink/svxlink.conf)
4. Soundlautstärken einstellen (mittels alsamixer – siehe Webseite)

Der Aufbau eines Interfaces ist auf der svxlink.de und im Netz an mehreren Stellen beschrieben. U.a. habe ich im November auf der Interradio einen kleinen Vortrag gehalten und einen weitere einfachen Vorschlag unterbreitet.

Der einfachste (etablierteste) Weg zur Steuerung von Tx und Rx ist mit einer seriellen Schnittstelle – die Futros habe eine feste RS232

Dort sollen Squelch und PTT angeschlossen werden. Der Squelch kann auch im Programm selbst abgeleitet werden, z.B. wenn man direkt an den Diskriminator anschließen möchte – bitte informieren!

Das Linux-Betriebssystem ist Voyage 0.9.2 (voyage.hk), welches optimiert für Flashkarten ist. Es ist derzeit auf einem Debian Wheezy 7.3 aufsetzend. Die Flashkarte ist nach dem Booten nur lesbar gemountet – Änderungen an Dateien lassen sich erst durchführen, wenn mittels remountrwäls root die Flashkarte schreibbar gemacht wurde.

Einloggen bitte mit:

user: root
Passwort: voyage

Ich empfehle noch, einen unprivilegierten Benutzer einzurichten!

Also als root:
adduser db0xxx

Dann auch diesen User mit den notwendigen Rechten ausstatten

```
addgroup db0xxx audio  
addgroup db0xxx svxlink  
addgroup db0xxx dialout
```

Dann sollte es später auch möglich werden, als db0xxx svxlink zu starten

Wenn Soundkarte und Squelch sowie PTT angeschlossen sind, können wir jetzt schon als root mal starten

```
svxlink
```

Auf dem Terminal läuft dann alles mit – ohne die Anschlüsse gibt es jetzt natürlich Fehlermeldungen. mit der Tastatur können wir "*" eingeben – der Sender sollte getastet werden und die Eröffnungsmeldung kommt

Das Programm wird mit "q" beendet.

Wenn ihr hier seid, dann freue ich mich auf weitere Fragen, die wir gerne dann mittels einer gemeinsamen Terminalsitzung (Skype und screen parallel) unmittelbar ausräumen können – ich würde mich dann auf den Rechner

gerne einmal mit Euch zusammen einloggen. Dazu bitte dann vorher user DK5HH anlegen, so dass ich mich einloggen kann. (hinterher kann man den mit deluser dk5hh wieder löschen)

Noch ein wichtiger Hinweis, nach Erstellung dieses Images wurde im Trunk noch eine kleine Erweiterung eingefügt, bitte den Parameter

```
PTT_TYPE=SerialPin
```

in den Abschnitt [Tx] einfügen.

4.3 März

Beeps bei EchoLink-Verbindung unterdrücken (z.B. I-Phone) (2014-03-02 08:16)

Speziell bei EchoLinkverbindungen über iPhones kann es zu Problemen in der Form kommen, dass während der Übertragung von dieser Station viele Roger-Pieps ausgelöst werden weil eine kurze Unterbrechung des Audiostreams als Ende der Übertragung erkannt werden.

Um das abzustellen bitte folgende Änderungen durchführen:

- 1) Im Verzeichnis /usr/share/svxlink/events.d sollte es ein Verzeichnis local"geben, falls nicht bitte anlegen.
- 2) Erstellen sie in diesem Verzeichnis eine Datei, z.B. EchoLink-user.tcl mit dem folgenden Inhalt:

```
namespace eval EchoLink {  
  
#  
# Executed when a transmission from an EchoLink station is starting  
# or stopping  
#  
proc is_receiving {rx} {  
    #if {$rx == 0} {  
        # playTone 1000 100 100;  
        #}  
    }  
  
# end of namespace  
}  
  
#  
# This file has not been truncated  
#
```


SvxLink @ DBOHAL/DBOPET auf dem Petersberg (Stand: 21.8.14) (2014-03-23 09:46)

Vorbetrachtungen:

DBOHAL (ex Y21H) war eines der ersten 2m-Relais der ehemaligen DDR und wurde bis Ende 2013 in fast unveränderter technischer Ausstattung betrieben [1].

Auf Grund von extrem starken Radiosendern [2] in unmittelbarer Umgebung konnte das Relais in den letzten Jahren leider nicht mehr genutzt werden, da nach dem Auftasten die IM-Produkte ein Abfallen des Relais und das QSO-Fahren bzw. standby-sein verhinderten.

Die bis Ende 2013 verwendete Technik basierte auf einer kommerziellen Relaisfunkstelle UFZ600, gebaut in Ungarn. Bei dieser wurde Anfang der 80er Jahre der Eingang vollständig umgebaut (Quarzfilter für 145.15MHz, anderer Mischer, andere Oszillatorkaufbereitung). Diese Technik hatte sich über viele Jahre bewährt!

Allerdings kam ein Umbau der vorhandenen Relaisstation aus verschiedenen Gründen nicht in Frage, sie wurde durch eine Motorola MC Compact ersetzt in die ein Futro (kleiner PC, sog. Thin client) als Steuerrechner eingebaut wurde. Auf diesem läuft ein Linux-System (openSuSE) und die Relaissteuersoftware SvxLink [3]. Diese Software übernimmt die komplette Ablaufsteuerung (Kennungen, Ansagen, EchoLink, Ausbreitungsinformationen, ...) so dass ab sofort ein erheblich größerer und auch zeitgemäßerer Funktionsumfang für die OM's zur Verfügung steht.

Aktivitäten:

Initiiert von Tino/DM2NT und mit freundlicher Genehmigung von Steffen/DL2HYM (Lizenzinhaber DBOHAL) haben wir uns an dieses Projekt herangewagt.



Bild 1: Inbetriebnahme neue Technik DBOHAL

Durch aufwändige Filtertechnik (Sperrfilter im RX- und TX-Zweig, Netzfilter, ...) konnten wir die Störungen erheblich reduzieren aber leider nicht ganz unterdrücken, so dass zum Arbeiten über DBOHAL bis 16.4. ein CTCSS-Ton von 250.3Hz notwendig war. Der Standort der 2m-Relaisstelle wurde näher an die Antennen verlegt (ca 20m weniger Kabel) und als Zuführung doppelt geschirmtes Kabel verwendet, die Zuleitung zur Sendeantenne ist jetzt sogar Cellflex, leider ohne wesentliche Besserung der Situation. Der Störnebel auf der Eingabe liegt bei etwa -110dBm. Seit heute (16.4.2014) ist ein Arbeiten ohne CTCSS-Ton möglich, trotzdem werden wir den Weg mit der Installation von Remote-RXe weiterverfolgen. Nach Rücksprache mit der Bundesnetzagentur ist es möglich, abgesetzte RXe für eine Relaisstation zu betreiben. Im Einzugsbereich von DBOHAL werden drei Empfänger auf 145.150MHz

an verschiedenen Standorten installiert und per Netzwerk/DSL mit einem Voter verbunden. Der Voter wählt das jeweils beste Empfangssignal aus und schaltet es auf den TX. Bild 2 zeigt den geplanten Endausbau. Die auf allen Systemen eingesetzte Software basiert auf dem SvxLink-Projekt [3].

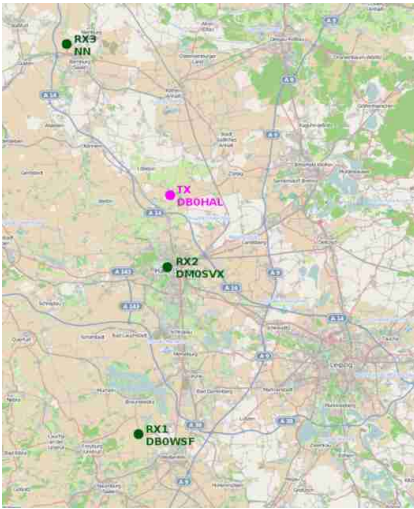


Bild 2: Geplanter Endausbau Remote-RXe/TX DB0HAL

Als Kriterium für die Signalqualität wird von jedem RX ein numerischer Wert aus den Rauschteilen des NF-Signales berechnet und vom Voter verglichen. Welcher RX auf den TX aufgeschaltet wird berechnet der Voter alle 200ms neu. Als Audiostream vom RX zum TX wird der Opus-Codec [5] verwendet der im Vergleich zum Speex-Codec [4] etwas weniger Latenz bei geringerer Bandbreite aufweist.

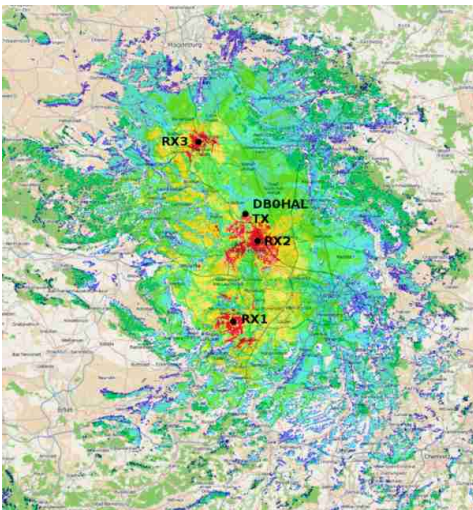


Bild 3: DB0HAL - RX-Prognose bei geplantem Endausbau mit 3 Remote-RX'en (Gegenstelle Handfunkgerät mit 2W)

Vorteile Remote-RXe:

- keine aufwändige Filtertechnik (Duplexer) mehr notwendig, Filterdämpfung momentan >2dB(!)
- Bessere RX-Ausleuchtung bestimmter Bereiche (Innenstädte, Täler, ...) möglich
- Ausgleich geringerer Sendeleistung z.B. bei Handfunkgeräten
- Störausblendung möglich wenn nur ein RX betroffen ist

Nachteile:

- Abhängigkeit vom Netzwerk/Linkstrecken
- u.U. höhere Betriebskosten an verschiedenen Standorten
- Netzwerkübertragung des Audiostreams führt i.A. zu Latenzen

Voraussetzung für die Verbindung der einzelnen RXe mit dem TX ist natürlich eine Netzwerkverbindung. Zu diesem Zweck haben wir eine Linkstrecke vom Petersberg nach Halle (City) aufgebaut. Für eine stabile Verbindung sind nur wenige mW notwendig und auch zulässig. Diese Netzanbindung kann natürlich auch genutzt werden um von der Ferne Betriebsparameter zu verändern oder zu überwachen. So muß man nicht wegen jeder Kleinigkeit auf den Berg fahren.

Am 29.3.2014 wurde am Standort von DB0WSF der erste Remote-RX (RX1) in Betrieb genommen. Dieser wird das südliche Sachsen-Anhalt RXmäßig gut abdecken.

RX2 (DM0SVX) ist speziell für das Stadtgebiet von Halle/Saale vorgesehen und sollte im Zentrum der Stadt ein Arbeiten mit sehr geringer Leistung möglich machen.

Für RX3 haben wir möglicherweise einen sehr guten Standort in der Nähe von Bernburg gefunden. Vielen herzlichen Dank an die OM's vor Ort.

- in Betrieb RX0 auf dem Petersberg - DB0HAL (cw = E)
- in Betrieb RX1 in Pettstädt - DB0WSF (cw = I)
- in Betrieb RX2 in Halle/Saale - DM0SVX (cw = S)
- in Vorbereitung RX3 bei NN

Bild 4 zeigt die Antennenanordnungen am Standort von DB0WSF und deren Nutzung.



Antennenanordnung bei DBWSF

Ausstattung RXe (für alle ähnlich):

- RX = MC Micro (Motorola)
- Antenne = X50N, ca. 7m hoch
- USB-Soundkarte
- PC=Futro,Cubie-Board o.ä
- Software: remotetrx aus dem SvxLink-Projekt
- Betriebssystem: Linux (openSUSE 12.3 und 13.1 [6])

Konfiguration DB0HAL (2m - 145.750MHz):

- Hilfe (0)

- Parrot - Sprachpapagei (1)
- EchoLink (2)
- METAR-Information - Ausgabe von Wetterinformationen (3)
- TCL-Voicemailbox - Sprachmailbox (4)
- PropagationMonitor - Ausbreitungsmonitor, Informationen zu Aurora, F2, TEP,... (5)
- SelCallEncoder - Umwandlung von DTMF in ZVEI-Selektivruf (6)
- WeatherInformation - Ausgabe von Unwetterwarnungen (8)

Konfiguration DBOPET (70cm - 439.000MHz):

- Hilfe (0)
- Parrot - Sprachpapagei (1)
- EchoLink (2) geplant
- METAR-Information (3)
- TCL-Voicemailbox - Sprachmailbox (4)
- SelCallEncoder - Umwandlung von DTMF in ZVEI-Selektivruf (6)
- WeatherInformation - Ausgabe von Unwetterwarnungen (8) geplant

Konfiguration DBOPET (23cm - 1298.325MHz) NOCH NICHT IN BETRIEB!!!

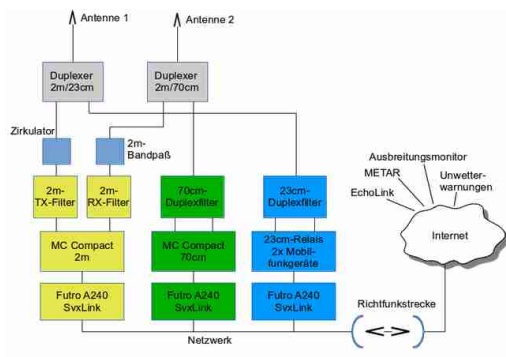
- Hilfe (0)
- Parrot - Sprachpapagei (1)
- EchoLink (2) geplant
- METAR-Information (3)
- TCL-Voicemailbox - Sprachmailbox (4)
- SelCallEncoder - Umwandlung von DTMF in ZVEI-Selektivruf (6)

Für das 23cm-Relais ist eine permanente Verlinkung mit anderen 23cm-Relais angedacht (z.B. in Sachsen) um die Chance zu erhöhen einen QSO-Partner zu finden. Des weiteren sollen sich die einzelnen Relais per Kommando verbinden lassen um z.B. bei Rundsprüchen mehr Zuhörer zu erreichen.

Stand 16.4.2014:

Neben DB0HAL wurde auch DBOPET (70cm) neu aufgebaut und dabei die vorhandene Technik nahezu vollständig ersetzt. Die Richtfunkstrecke wurde neu ausgerichtet, der Link steht jetzt mit -64dBm bei 16,25MBit/s Datenrate sehr stabil so dass sich Einstellungen von der Ferne ändern lassen und verschiedene Dienste erst möglich werden. Alle Kennungen werden zukünftig nur noch als Sprachansagen ausgegeben.

Hier der aktuelle Aufbau schematisch dargestellt, 23cm ist noch NICHT in Betrieb, soll aber bald folgen!



Schematische Übersicht DB0HAL(2m) DB0PET (70/23cm)

Nachfolgend ein paar Bilder vom Umbau am 16.4.2014:



Bild 5: Der Relaisschrank (2m/70cm ohne 23cm)



Bild 6: 2m und 70cm Relaisstationen MC Compact (Motorola)



Bild 7: Richtfunkstrecke nach Halle



Bild 8: Zirkulator zum Schutz des TX bei defekter Antenne



Bild 9: Ein Haufen SZeugsmuß über viele Stufen nach oben gebracht werden

Stand 20.4.2014:

Der Remote-RX 1 hat ein neues Gehäuse im DB0WSF-Schrank bekommen. Es ist noch einiges an Arbeit erforderlich bis zum endgültig geplanten Ausbaustand.

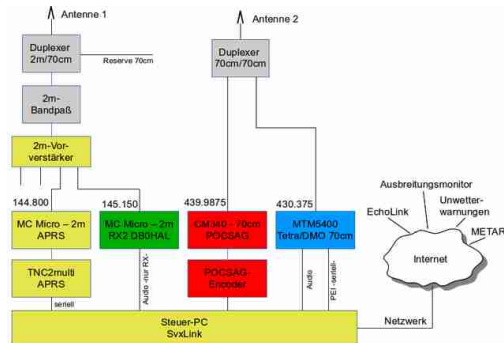


Bild 10: (von oben) Steuer-Rechner, links RX1 für DBOHAL rechts Tetra, Quantar (DB0WSF - 70cm-Relais), ganz

unten APRS

Stand 21.04.2014:

Die Technik für DM0SVX ist fast fertig. Noch ein paar Tage, dann kommt es wieder hoch auf den Turm. Hier die schematische Darstellung, es ist noch etwas mehr geplant als bisher.



Schematische Übersicht der Funktionsweise DM0SVX (POCSAG muß noch beantragt werden)

Und so sieht das dann aus wenn alles zusammengestellt ist:



Technik DM0SVX, von oben: TNC+Netzteil+POCSAG-Encoder, Steuer-PC, Funktechnik, ganz unten Einschub mit Duplexfilter+Vorverstärker+Bandpaß

Stand 24.04.2014:

An dieser Stelle herzlichen Dank an die Firma Funktechnik Dathe [7] in Bad Lausick, die spontan zwei 2m-Rundstrahlantennen zur Verfügung gestellt hat.

Stand 28.04.2014:

Der erste Teil ist wieder oben auf dem Turm (Funktechnik, Netzteil und Server), mit etwas Glück kriegen wir den 2.RX noch vor meinem Urlaub zum laufen, allerdings nur testweise. Es muß leider noch etwas umgebaut werden.

Stand 29.04.2014:

Seit heute erfolgt der Test mit dem 2. Remote-TRX in Halle/City. Es kann sein dass der NF-Pegel noch nicht optimal ist. Eventuell erfolgt die Umstellung der Roger-Beeps von 1-3 Punkten (. | .. | ...) auf veränderte Tonhöhen (440,

540, 640 Hz) zur Erkennung über welchen RX der Durchgang empfangen wurde....mal schauen.

Stand 15.05.2014:

Zu den Zeiten der Rundsprüche erfolgt eine testweise Zusammenschaltung beider Relais (2m und 70cm), beginnend ab Montag, 19.05.2014 jeweils in der Zeit von 19 bis 20Uhr. Es wird empfohlen nur auf einer Station (entweder 2m oder 70cm) zu hören, da die Audiostreams bedingt durch Netzwerklatenzen zeitlich verschoben zueinander sind und somit ein Echoeffekt entsteht.

Stand 31.05.2014:

Einen weiteren Umbau am Schrank bei DB0WSF vorgenommen und bei dieser Gelegenheit eine USB-Soundkarte für den Remote-TRX getauscht die Ärger gemacht hatte.

Stand 30.06.2014:

Die Genehmigung für die Erweiterung um einen Funkruf-Sender (POCSAG auf 439.9875MHz) ist eingetroffen. Der Vortrag auf der HAMRADIO 2014 zum Thema SvxLink (u.a. zum DB0HAL-Voter) ist jetzt online.

Stand 21.08.2014:

Nach einer unvorhergesehenen Sendepause ist seit heute DB0PET (70cm) wieder am Netz. Das 23cm-Relais wird gerade fertiggestellt, mit etwas Glück kann man bald wieder darüber funken.

Bedienungsanleitung SvxLink:

Hier auf dieser Domain findet man alles dazu. Wichtig ist, jedes Kommando mit einer Raute (#) zu bestätigen. Eventuell kann man auch jemanden fragen der sich damit auskennt, das ist allerdings ein bisschen aus der Mode gekommen.

Kontakt:

Tino/DM2NT und Adi/DL1HRC: db0hal {at } svxlink.de

Links:

- [1] DBOLKP - Bilder Technik FM-Relais
- [2] Fernmeldeturm Petersberg bei Halle
- [3] svxlink - Steuersoftware für Relaisstationen
- [4] speex-Codec bei Wikipedia
- [5] Opus-Audiocodec bei Wikipedia
- [6] OpenSuSE - Linux
- [7] Funktechnik Dathe Bad Lausick

4.4 Juni

Qso-Recorder (2014-06-24 19:19)

Mit Hilfe des QSO-Recorders kann man ein laufendes QSO, welches gerade auf dem Relais/Link stattfindet aufzeichnen.

Aktivierung innerhalb eines Logic-Abschnittes durch, z.B.:

[SimplexLogic]

...

QSO_RECORDER=8:QsoRecorder

Konfiguration:

[QsoRecorder]

REC_DIR=/var/spool/svxlink/qso_recorder

MAX_TIME=3600

MIN_TIME=200

SOFT_TIME=300

MAX_DIRSIZE=1024

DEFAULT_ACTIVE=1

TIMEMOUT=100

QSO_TIMEOUT=300

ENCODER_CMD=

Beschreibung:

REC_DIR

Verzeichnis in dem die aufgezeichneten QSO's als WAV-Datei abgelegt werden sollen. Empfehlung /var/spool/svxlink/qso_recorder

MIN_TIME

Wenn die Länge des aufgezeichneten Audiostreams kürzer ist als per MIN_TIME konfiguriert, dann wird das File sofort wieder gelöscht, nachdem die Aufzeichnung beendet ist. Standardeinstellung=0, leere Dateien werden gelöscht.

MAX_TIME

Hier wird die maximale Aufzeichnungszeit pro Audiofile definiert. Wird die hier definierte Aufzeichnungslänge überschritten, so wird das aktuelle File geschlossen und ein neues File geöffnet. Standardeinstellung=0 (unbegrenzt)

SOFT_TIME

Um keine abrupten Unterbrechungen in der Aufzeichnung zu erhalten ist es möglich diese SOFT_TIME zu konfigurieren. Beispiel: Als MAX_TIME wurden 3600 Sekunden (1 Stunde) konfiguriert. Wenn SOFT_TIME=300 konfiguriert wurde, so wird versucht in der Zeit von 55 Minuten bis 60 Minuten ein Schließen der Rauschsperrung auszunutzen um die Aufzeichnung zu beenden und gleich mit einem neuen Audiofile fortzuführen. Hierbei wird errichtet, dass während einer Audioübertragung der Stream in zwei Stücke zerteilt wird.

MAX_DIRSIZE

Definiert die maximale Größe in MegaByte als Summe aller Files im Verzeichnis. Wird diese Größe überschritten, so werden die ältesten Files gelöscht. Die Verzeichnisgröße wird bei jedem Schließen eines Audiofiles überprüft. Hierbei werden nur Files mit dem Präfix "qso_rec_" berücksichtigt. Dies sollte beachtet werden, wenn man ENCODING_CMD verwendet. Das dort konfigurierte Kommando sollte den Präfix "qso_rec" nicht entfernen, wenn sichergestellt sein soll, dass die Verzeichnisgröße nicht überschritten werden soll. Standardeinstellung=0 (keine Begrenzung)

DEFAULT_ACTIVE

Wenn diese Variable =1 ist, dann wird der QSO-Recorder automatisch bei jedem SvxLink-Start aktiviert. Standardeinstellung=0 (inaktiv)

QSO_TIMEOUT

Wenn diese Variable gesetzt ist, wird das momentan geöffnete File nach jedem QSO geschlossen. Definiert wird die Anzahl in Sekunden, die ein Repeater/Link nicht genutzt wird um ein QSO als beendet zu erkennen.
Standardeinstellung=0 (kein QSO Timeout)

ENCODER_CMD

Hier wird ein Befehl (Shell-Befehl) definiert, der jedes Mal ausgeführt wird, wenn die Aufzeichnung eines Files beendet wurde. Er wurde eingerichtet um beispielsweise eine Konvertierung des aufgezeichneten Files von *.wav nach *.mp3 anzustoßen. Allerdings können auch umfangreichere Aktionen ausgeführt werden wie z.B. das Kopieren der Files auf einen anderen Computer.

Dieses Kommando läuft unter der Shell, so dass die normalen Shell-Operatoren (Umleitung, Pipes, ...) verwendet werden können

4.5 August

Direkter SvxLink-2-SvxLink Verbund (SvxServer) (2014-08-12 08:35)

Seit einigen Wochen probieren wir in Mitteldeutschland eine direkte Kopplung verschiedener SvxLink-Nodes aus. Das Problem bei der Remote-TRX-Variante ist ja, dass das Modul EchoLink nur auf dem Hauptnode zur Verfügung steht und beim Ausfall der Netzverbindung die einzelnen Repeater in den Fallback-Mode umschalten. Das wollten wir ändern!



SvxLink-Verbund mehrerer Repeater in Mitteldeutschland

Das Verlinken der Repeater ist natürlich analog des bekannten Remote-TRX-Konzepts per DTMF steuerbar. Verbunden sind alle Stationen per DSL, HAMNET wäre natürlich auch möglich und ist eher das Problem der Standorte und des Preises.

Die Repeater sind per TCP mit einer zentralen Serverinstanz verbunden. Diese nimmt die Audio- und Controlmessages entgegen und verteilt sie an die angeschlossenen Stationen. Auf den Relaisstationen ist neben der RepeaterLogic noch die Einrichtung einer NetLogic erforderlich. Diese hält die Verbindung zur zentralen Serverinstanz aufrecht.

Das Konzept wird momentan auch auf verschiedenen Stationen im Harz (DB0OHA, DB0EIG, DL7ACT) erfolgreich angewendet. Bis diese Variante in den Trunk gelangt wird aber sicher noch einige Zeit ins Land gehen.

Hier mal eine Beispielkonfiguration für die OMs die es mal probieren wollen. Es sind nur die Parameter aufgelistet, die für die Verlinkung eine Bedeutung haben! Zugangsdaten können bei mir (DL1HRC) angefragt werden. Es handelt sich um Änderungen in der svxlink.conf.

```
[GLOBAL]
LOGICS=RepeaterLogic,NetLogic
LINKS=NetLink
```

```
[NetLink]
NAME=NetLink
DEFAULT_ACTIVE=1
CONNECT_LOGICS=RepeaterLogic:34,NetLogic:35
```

```
[NetLogic]
TYPE=Repeater
RX=NetRx
TX=NetTx
CALLSIGN=NONE
EVENT_HANDLER=/usr/share/svxlink/events.tcl
DEFAULT_LANG=de_DE
OPEN_ON_SQL=50
OPEN_SQL_FLANK=OPEN
NO_REPEAT=1
IDLE_TIMEOUT=10
```

```
[NetRx]
TYPE=Net
HOST=svxserver.domain.net
TCP_PORT=5210
AUTH_KEY="geheim"
CODEC=OPUS
```

```
[NetTx]
TYPE=Net
HOST=svxserver.domain.net
TCP_PORT=5210
AUTH_KEY="geheim"
CODEC=OPUS
```

Weiterhin ist ein angepasstes TCL-File notwendig welches hier heruntergeladen werden kann. Dieses ist nach /usr/share/svxlink/events.d zu installieren. Einzige Aufgabe ist die tcl-Prozeduren innerhalb der NetLogic-Umgebung zu deaktivieren, da diese auf der Netzwerkseite nicht gebraucht werden bzw. störend sind.

Wer einen eigenen svxserver aufsetzen möchte kann wie folgt vorgehen.
Bitte beachten, dass dieses Programm nicht offizieller Bestandteil der SvxLink-Applikation ist.

```
git clone https://github.com/dl1hrc/svxlink.git
cd svxlink
git checkout svxserver
mkdir src/build
cd src/build
cmake -DUSE_QT=OFF -DCMAKE_INSTALL_PREFIX=/usr -DSYSCONF_INSTALL_DIR=/etc -DLOCAL_STATE_DIR=/var -DCMAKE_BUILD_TYPE=Release ..
sudo make install
```

Jetzt noch die Konfigurationsdatei svxserver.conf anpassen:

```
[GLOBAL]
LISTEN_PORT=5212
AUTH_KEY="ganz_geheim"
HEARTBEAT_TIMEOUT=10
```

4.6 Dezember

Zeitlich gesteuerte Ausgaben generieren (2014-12-10 09:37)

Viele OM's möchten gern per automatischer Relaisansage beispielsweise auf den nächsten OV-Abend hinweisen. Um so eine Ausgabe zu realisieren gibt es viele Ansätze, im Folgenden sollen drei davon vorgestellt werden. Alle basieren auf dem SvxLink-eigenen TCL-Framework und darin enthaltenen Zeitfunktionen.

Für den weiteren Verlauf wird davon ausgegangen, dass eine Ansage-Datei mit Namen "ÖV-Abend.wav" bereits erstellt und ins Verzeichnis /usr/share/svxlink/events.d/de_DE/Core kopiert wurde (oder unter ../en_US/... je nach Locale-Umgebung).

1) Nutzung der every_minute {}-Prozedur

Im TCL-Framework gibt es eine Prozedur, die zu jeder vollen Minute aufgerufen wird, die every_minute {}-Prozedur. Diese kann man zur Ausgabe heranziehen, allerdings indirekt über addTimerTickSubscriber {}!

Erstellen Sie nach dem locale-Konzept eine Datei meine_ansage.tcl mit folgendem Inhalt:

```
# /usr/share/svxlink/events.d/local/meine_ansagen.tcl
namespace eval Logic {
    Logic::addTimerTickSubscriber relaisansage;

    proc relaisansage {} {
        puts "Ausgabe Relaisansage?";
        source "/home/svxlink/relaisansage.tcl";
    }
}
```

Erstellen Sie eine Datei relaisansage.tcl im /home/svxlink-Verzeichnis mit folgendem Inhalt:

```
# /home/svxlink/relaisansage.tcl
# aktuelle Uhrzeit ermitteln
set now [clock seconds];
set hour [clock format $now -format "%H"];
set minute [clock format $now -format "%M"];
```

```

# eigene Ausgabezeiten definieren
set ansage_Minute 05;
set ansage_Stunde 09;

puts "Es ist $hour:$minute, Ausgabe um $ansage_Stunde:$ansage_Minute";

# prüfen ob Ausgabezeit erreicht ist
if {$hour == $ansage_Stunde && $minute == $ansage_Minute} {
    puts "Ausgabe Relaisansage";
    playMsg "Core" "OV-Abend";
}

```

Alle Zeilen mit puts beginnend dienen nur der Kontrollausgabe und können selbstverständlich weggelassen bzw. auskommentiert werden. Wichtig ist, die gewünschte Ausgabezeit mit führenden Nullen bei Werten kleiner 10 einzutragen, also 09 statt 9!

Wird eine periodisch über den Tag verteilte Ausgabe gewünscht so kann man hier die Modulo-Funktion zielführend einsetzen. Ändern Sie hierzu die Datei /home/svxlink/relaisansage.tcl wie folgt:

```

# /home/svxlink/relaisansage.tcl
# aktuelle Uhrzeit ermitteln
set now [clock seconds];
set hour [clock format $now -format "%H"];
set minute [clock format $now -format "%M"];

# eigene Ausgabezeiten definieren
set interval 6;
set ansage_Minute 22;

# hier prüfen ob Ausgabekriterium erreicht ist
if {($hour % $interval) == 0 && $minute == $ansage_Minute} {
    puts "Ausgabe Relaisansage";
    playMsg "Core" "OV-Abend";
}

```

In diesem Beispiel würde die Ausgabe zur 22. Minute einer ohne Teilungsrest durch 6 teilbaren Stunde erfolgen, also um 0:22, 6:22, 12:22 und 18:22.

2) Nutzung der periodischen Relais-Identifikationsfunktion

Erstellen Sie nach dem locale-Konzept eine Datei meine_ansagen.tcl mit nachfolgendem Inhalt. Achten Sie darauf, dass der Inhalt der Funktion Logic::send_long_ident { } dem Standard nach einer Neuinstallation entspricht. Falls Sie also diese Funktion schon geändert haben (z.B. per locale-Konzept in einer anderen tcl-Datei), so sollten Sie die gewünschten Änderungen dort durchführen!

```

# /usr/share/svxlink/events.d/local/meine_ansagen.tcl
namespace eval Logic {

#

```



```

# Executed when a long identification (e.g. hourly) should be sent
# hour - The hour on which this identification occur
# minute - The hour on which this identification occur
#
proc send_long_ident {hour minute} {
    global mycall;
    global loaded_modules;
    global active_module;
    variable CFG_TYPE;

    spellWord $mycall;
    if {$CFG_TYPE == "Repeater"} {
        playMsg "Core" "repeater";
    }
    playSilence 500;

    playMsg "Core" "the_time_is";
    playSilence 100;
    playTime $hour $minute;
    playSilence 500;

    # Call the "status_report" function in all modules if no module is
active
    if {$active_module == ""} {
        foreach module [split $loaded_modules " "] {
            set func "::$";
            append func $module "::$status_report";
            if {"[info procs $func]" ne ""} {
                $func;
            }
        }
    }
    playSilence 250;
    source "/home/svxlink/relaisansage.tcl";
    playSilence 250;
}
}

```

Erstellen Sie dann wieder eine Datei relaisansage.tcl im /home/svxlink-Verzeichnis. Der Inhalt kann jetzt etwas einfacher sein, muß aber nicht. Es ist jetzt allerdings wichtig geworden, wie oft Sie den LONG_IDENT_INTERVAL ausgeben lassen. Hat man LONG_IDENT_INTERVAL=120 konfiguriert, so würde die Datei relaisansage.tcl nur jede 2h aufgerufen werden:

```

# /home/svxlink/relaisansage.tcl
puts "Ausgabe Relaisansage";
playMsg "Core" "OV-Abend";

```

Natürlich kann man das mit dem Vergleich einer vordefinierten Stunde kombinieren wie unter Punkt 1) beschrieben. Hier muß man einfach ein bisschen probieren. Bitte vorher überlegen, ob das Kriterium überhaupt true werden kann.

Ein grosser Vorteil dieser Varianten ist, dass Änderungen in o.g. Datei sofortige Auswirkungen haben. Man kann diese Ausgaben quasi on-the-fly herausnehmen, indem man in der /home/svxlink/relaisansage.tcl einfach das

```
playMsg "Core" "OV-Abend";
```

auskommentiert. Ein Neustart ist nicht erforderlich. Bitte diese Datei nicht löschen!

3) Ansagen vor dem Äbfallen einer Relaisstation

Diese Variante funktioniert nur bei einer RepeaterLogic. Durch eine Änderung der Prozedur RepeaterLogic::repeater_down {} (locale-Konzept!) kann man auf einfache Weise realisieren, dass die gewünschte Ausgabe z.B. bei jedem 20. Relaisschließen erfolgen soll. Erstellen Sie eine Datei meine_ansagen.tcl nach dem locale-Konzept und prüfen Sie, ob Sie eventuell eine Änderung dieser Prozedur schon irgendwo RepeaterLogic::repeater_down {} vorgenommen haben.

```
# /usr/share/svxlink/events.d/local/meine_ansagen.tcl
namespace eval RepeaterLogic {
#
# Executed when the repeater is deactivated
#   reason - The reason why the repeater was deactivated
#           IDLE          - The idle timeout occurred
#           SQL_FLAP_SUP - Closed due to interference
#
proc repeater_down {reason} {
    global mycall;
    variable repeater_is_up;
    variable count;
    incr count;

    set repeater_is_up 0;

    if {$reason == "SQL_FLAP_SUP"} {
        playSilence 500;
        playMsg "Core" "interference";
        playSilence 500;
        return;
    }

    if {$count > 19} {
        source "/home/svxlink/relaisansage.tcl";
        set count 0;
    }

    set now [clock seconds];
    if {$now-$Logic::prev_ident < $Logic::min_time_between_ident} {
        playTone 400 900 50
        playSilence 100
        playTone 360 900 50
        playSilence 500
        return;
    }
}
```

```
}
set Logic::prev_ident $now;

spellWord $mycall;
playMsg "Core" "repeater";
playSilence 250;
#playMsg "../extra-sounds" "shutdown";
}

# end of namespace
}
```

Erzeugen Sie wieder die bekannte Datei `relaisansage.tcl` im Verzeichnis `/home/svmlink`:

```
# /home/svmlink/relaisansage.tcl
puts "Ausgabe Relaisansage";
playMsg "Core" "OV-Abend";
```

5. 2015

5.1 Januar

Lokaler Simplex-Link mit EchoLink (2015-01-16 08:45)

Wer einen Simplex-Link betreibt, wird seinen Link-TRX am eigenen Standort vermutlich mit einem Handfunkgerät ansprechen. Ein OM, der in einiger Entfernung zum Link über HF in ein Echolink-QSO einsteigen möchte, hört zwar den Gesprächspartner über Echolink, aber oft nicht den Linkbetreiber mit seinem Handfunkgerät – und umgekehrt.

Um nun den Funkverkehr aus dem weiteren Umkreis und den Funkverkehr im unmittelbaren Nahbereich des EL-Hotspots zusammenzuführen, benötigt man nicht zwingend eine RepeaterLogic. Man kann auch zwei Transceiver mit jeweils eigenständigen Simplex-Logiken so verlinken, dass ein Empfangssignal von Rx1 auf Tx2 und Rx2 auf Tx1 wiedergegeben wird.

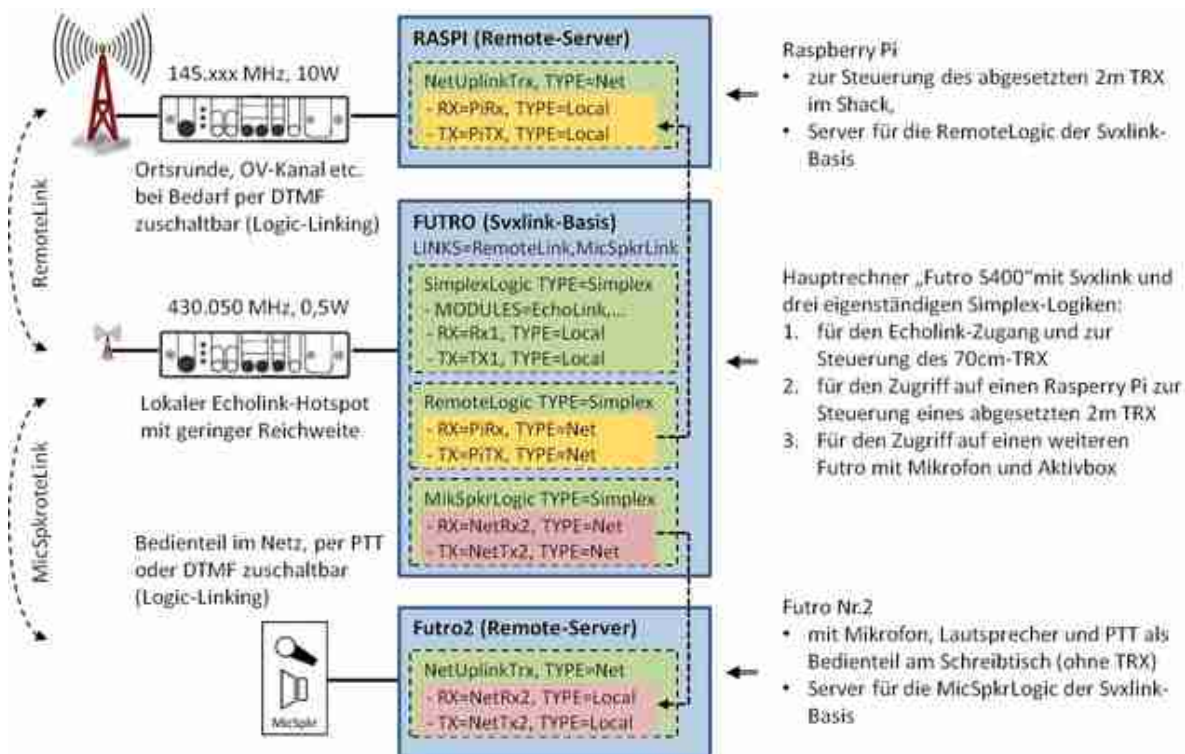
Im Gegensatz zur Funktion X-Band-Repeater einiger Transceiver ist das Logic-Linking bei SvXLink steuerbar: Mit DTMF-Kommandos können die einzelnen Logiken - und damit die TRX - zusammenschaltet oder getrennt werden. Mit einer zusätzlichen Mike-Speaker-Logic lässt sich zusätzlich ein ortsfestes Bedienteil für QSOs vom Schreibtisch oder aus dem Bastelkeller realisieren.

Nachfolgend wird ein erprobter Aufbau beschrieben:

Auf einem Hauptrechner läuft eine aktuelle SvXlink-Version mit drei Simplex-Logiken.

- Die erste Logik ist die bekannte SimplexLogic aus der Standard-Konfiguration mit einem Echolink-Modul. Sie steuert den eigentlichen EL-Hotspot mit Rx1, Tx1 auf 430.050 MHz im Nahbereich.
- Die zweite Logik (RemoteLogic) hat kein Modul EchoLink und greift auf einen 2m-Remotetrx größerer Reichweite zu. Als Remotetrx-Server dient ein Raspberry Pi. Anregung aus http://svxlink.de/?page_id=1228
- Die dritte Logik ist eine MicSpkr-Logik, die hier http://svxlink.de/?page_id=139 schon beschrieben worden ist. Sie läuft ebenfalls remote auf einem Rechner im Büro.

Die hier gewählte Realisierung über Remotetrx ist nicht zwingend erforderlich, der zweite TRX lässt sich auch direkt über den zweiten Kanal der Soundkarte am Hauptrechner betreiben.



Für die Konfiguration sind folgende Schritte durchzuführen:

Zunächst müssen Kopien der Datei SimplexLogic.tcl in /usr/share/svxlink/events.d angelegt werden. Die Dateien werden sinnvoll umbenannt, in der Deklaration von namespace eval ...Logic im Kopf der tcl-Datei muss allerdings der exakte Name der jeweiligen neuen Logik eingetragen werden.

Danach ist die svxlink.conf zu ergänzen. Wer seine alte Konfigurationsdatei erhalten möchte, erstellt davon eine Kopie mit dem Dateinamen „svxtest.conf“ (oder ähnlich) und arbeitet damit weiter.

```
[GLOBAL]
...
LOGICS=SimplexLogic,RemoteLogic,MicSpkrLogic
LINKS=RemoteLink,MicSpkrLink
...
```

Beschreibung der zusätzlichen Simplex-Logiken

```
[RemoteLogic]
TYPE=Simplex
RX=PiRx1
TX=PiTx1
MUTE_TX_ON_RX=0
# restliche Konfiguration aus SimplexLogic kopieren

[MicSpkrLogic]
TYPE=Simplex
RX=NetRx2
TX=NetTx2
MUTE_RX_ON_TX=0
```

```
# restliche Konfiguration aus SimplexLogic kopieren
```

Beschreibung der möglichen Logik-Verlinkungen

```
[RemoteLink]
CONNECT_LOGICS=SimplexLogic:91:DL8BBY,RemoteLogic:92:DL8BBY
DEFAULT_ACTIVE=0
TIMEOUT=60
#AUTOACTIVATE_ON_SQL=RemoteLogic
## Achtung: Bei DEFAULT_ACTIVE=1 wird die Verlinkung bei manueller
## Abschaltung nach 60 Sekunden automatisch wieder hergestellt
```

```
[MicSpkrLink]
CONNECT_LOGICS=SimplexLogic:93:DL8BBY,MicSpkrLogic:94:DL8BBY
DEFAULT_ACTIVE=0
TIMEOUT=60
AUTOACTIVATE_ON_SQL=MicSpkrLogic
# SQL ist hier die PTT und die MicSpkrLogic an den EL-Rechner
```

Beschreibung der zusätzlichen TRX

```
[NetRx2]
#futura2 mit Mic=Rx2
TYPE=Net
HOST=192.168.1.50
TCP_PORT=5212
AUTH_KEY=*****
CODEC=OPUS
# oder CODEC=S16 bei Verlinkung im häuslichen LAN
```

```
[NetTx2]
#futura2 mit Spkr=Tx2
TYPE=Net
HOST=192.168.1.50
TCP_PORT=5212
AUTH_KEY=*****
CODEC=OPUS
```

```
[PiRx1]
#raspi1
TYPE=Net
HOST=192.168.1.36
TCP_PORT=5236
AUTH_KEY=*****
CODEC=OPUS
```

```
[PiTx1]
TYPE=Net
HOST=192.168.1.36
TCP_PORT=5236
AUTH_KEY=*****
```

```
CODEC=OPUS
```

Bei den remote betriebenen Logiken muss jeweils noch die remotetrx.conf auf den abgesetzten Rechnern erstellt werden. Dafür sollte man zunächst die svxlink.conf soweit bearbeiten, dass damit die Funktionen von SQL und PTT sowie die Audioeinstellungen mit der vorgesehenen Hardware am abgesetzten Rechner getestet werden können (z.B. mit Parrot). Aus einer lauffähigen svxlink.conf können dann die Beschreibungen für Rx und Tx in die remotetrx.conf übernommen werden.

```
[GLOBAL]
TRXS=NetUplinkTrx
TIMESTAMP_FORMAT="%c"

[NetUplinkTrx]
TYPE=Net
RX=NetRx2
TX=NetTx2
LISTEN_PORT=5212
AUTH_KEY=*****

[NetRx2]
# hier die Rx-Beschreibung aus der svxlink.conf einfügen
...

[NetTx2]
# hier die Tx-Beschreibung aus der svxlink.conf einfügen
```

Nach gleichem Muster wird die remotetrx.conf für den Remote-Transceiver angelegt (im Beispiel PiRx1 und PiTx1).

Die Konfiguration ist damit abgeschlossen und kann ausprobiert werden. Dazu wird auf den abgesetzten Rechnern remotetrx und auf dem Hauptrechner svxlink gestartet. Wenn die Änderungen in einer Kopie „svxtest.conf“ vorgenommen wurden, startet man SvxLink mit

```
svxlink --config=svxtest.conf
```

Auf der Konsole sollte man jetzt sehen können, wie neben der bisherigen SimplexLogic auch die neuen Logiken hochgefahren werden. Wenn alles geklappt hat, sind nun die Statusmeldungen von SQL und PTT der lokalen und abgesetzten TRX zu sehen, auch wenn die Logiken noch nicht aktiviert sind.

Sind Module aktiv, z.B. bei einer Echolink-Verbindung, können bestimmte Funktionen wie Logic-Linking oder der QSO-Recorder nicht mehr per DTMF gesteuert werden können. In diesen Fällen müssen die DTMF-Sequenzen mit dem Sternchen * eingeleitet werden. Andere Module, z.B. Parrot, lassen sich bei bestehender EL-Verbindung mit diesem Trick nicht starten.

5.2 April

GPIO-Konfiguration (2015-04-17 07:43)

GPIO-Konfiguration für BananaPi

Nachfolgend eine Beispielkonfiguration zum Anschluß von zwei Funkgeräten (jeweils SQL und PTT). Man kann natürlich auch andere Pins verwenden, muß dann aber seine Konfiguration anpassen.

Funkgerät 1

```
PTT1 = Pin 19 = MOSI = PI12 = gpio10
SQL1 = Pin 5 = SCL = PB20 = gpio3
```

Funkgerät 2

```
PTT2 = Pin 21 = MISO = PI13 = gpio9
SQL2 = Pin 3 = SDA = PB21 = gpio2
```

Bevor man die oben genannten GPIO's nutzen kann, müssen diese zunächst initialisiert werden. Eine Möglichkeit besteht durch den Aufruf innerhalb der `/etc/init.d/boot.local` für openSUSE oder in der `/etc/init.d/rc.local` für Debian/Ubuntu:

```
echo "9" >/sys/class/gpio/export
echo "out" >/sys/class/gpio/gpio9/direction
echo "10" >/sys/class/gpio/export
echo "out" >/sys/class/gpio/gpio10/direction
echo "3" >/sys/class/gpio/export
echo "in" >/sys/class/gpio/gpio3/direction
echo "2" >/sys/class/gpio/export
echo "in" >/sys/class/gpio/gpio2/direction
```

Für den Fall, dass der Pegel invertiert werden muß, kann man dieses entweder in der `svxlink.conf` eintragen oder auch hier in der `boot.local` bzw. `rc.local` durch z.B.

```
echo "1" >/sys/class/gpio/gpio2/active_low
```

Nun kann die Konfiguration innerhalb der `svxlink.conf` erfolgen.

```
[Rx1]
SQL_DET=GPIO
GPIO_SQL_PIN=gpio3

[Tx1]
```

```
PTT_TYPE=GPIO
GPIO_PTT_PIN=gpio9

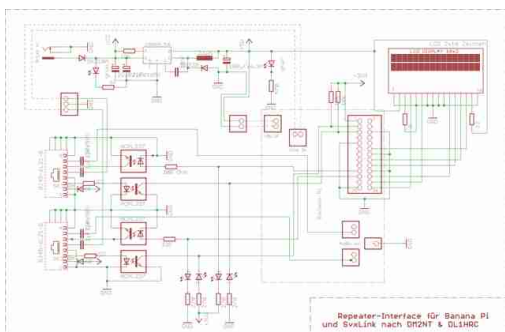
[Rx2]
SQL_DET=GPIO
GPIO_PTT_PIN=gpio2

[Tx2]
PTT_TYPE=GPIO
GPIO_SQL_PIN=gpio10
```

SvxLink-Adapter für BananaPi (2015-04-21 13:25)

Tino/DM2NT hat eine Platine in der Größe des BananaPi's entwickelt, an die zwei Funkgeräte/Relais angeschlossen und unabhängig voneinander angesteuert werden können. Als kleines Addon ist noch ein 2x16-LCD-Display angebracht über das zukünftig Statusinformationen ausgegeben werden sollen.

Es sind einige Platinen vorhanden. Bei Interesse bitte direkt mit Tino/DM2NT (dm2nt {at } svxlink . de) in Verbindung setzen. Hier der Schaltplan:



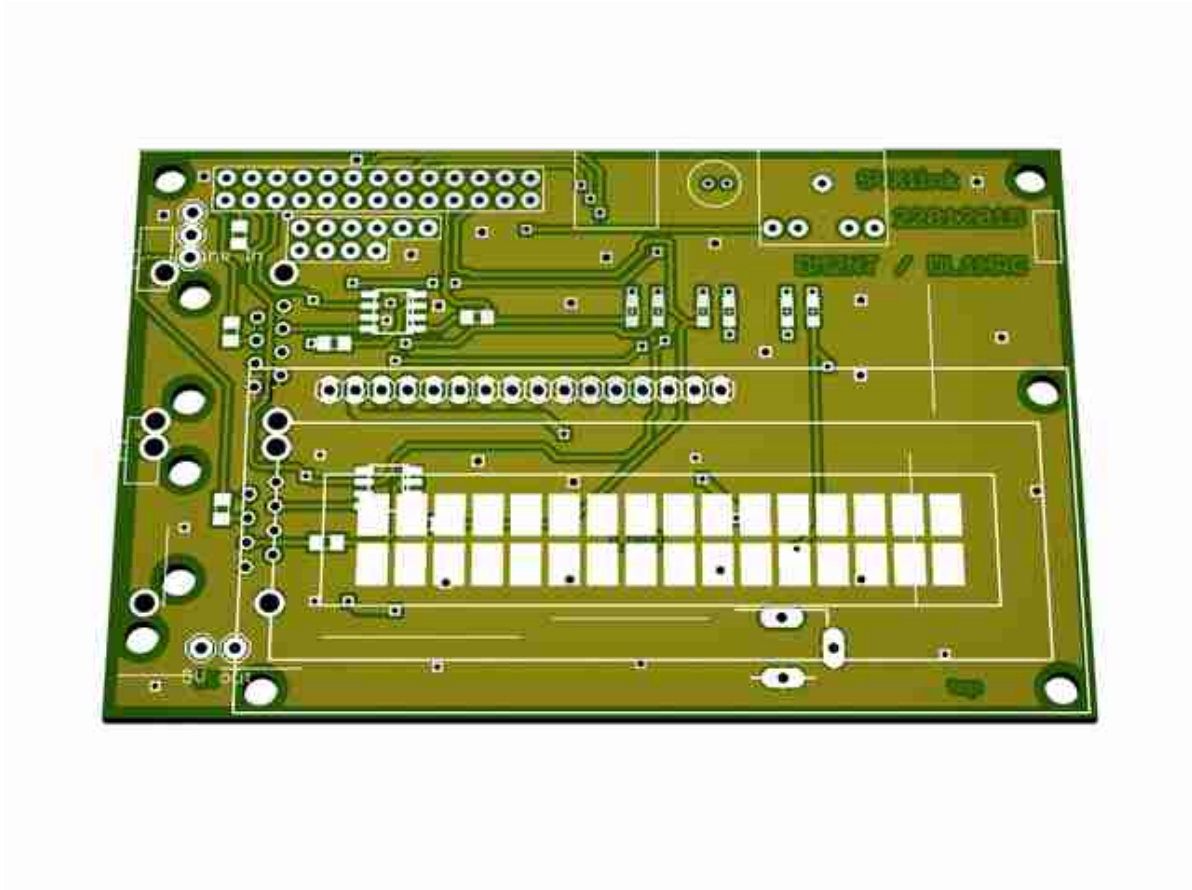
SvxLink-Adapter für BananaPi

Der Adapter wird mit 9-16V versorgt und erzeugt gleich die +5V für den BananaPi.

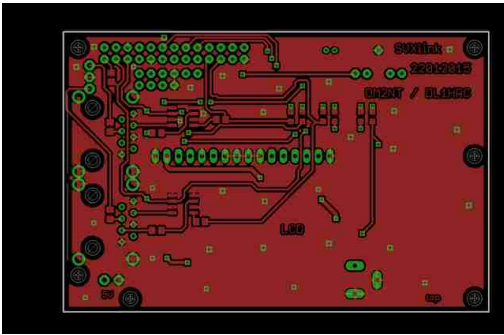


SvxLink-Adapter mit LCD-Statusdisplay

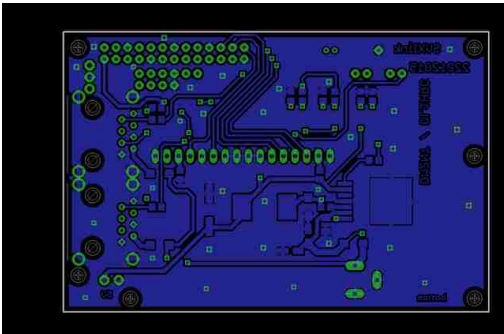
Nachfolgend noch die Layouts:



Adapter - fertige Platine mit Aufdruck



Adapter - Oberseite



Adapter - Unterseite

6. 2011

6.1 Januar

/etc/rc.d/svxlink start (2011-01-20 17:27)

Herzlich Willkommen auf der deutschen SvxLink-Seite.

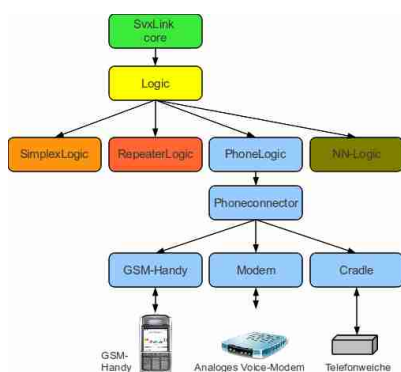
Wir wollen Funkamateure die sich etwas mit SvxLink beschäftigen möchten ein wenig den Einstieg in diese Materie erleichtern. Bitte beachten Sie dass diese Seiten keinen Anspruch auf Vollständigkeit erheben. Alle Basteleien gehen selbstverständlich auf eigenes Risiko. Für Hilfe, Probleme oder Fragen immer das Forum (im Aufbau) nutzen, bitte keine PM.

Viel Spaß beim Lesen

Martin (DF1AMB) & Adi (DL1HRC)

PhoneLogic (2011-01-14 18:52)

Eine ganz neue Idee ist eine PhoneLogic, also ein Interface, welches die Kopplung von SvxLink mit einem Telefonanschluß (Analog oder GSM) erlaubt.



Prinzipielle Darstellung der Phone-Erweiterung

Hier dargestellt ist der vorbereitete Anschluß eines GSM-Mobiltelefones an SvxLink. Voraussetzung ist dabei, dass sich das Mobiltelefon unter Linux als AT-Modem ansprechen läßt und den Anschluß einer externen Hör-/Sprechgarnitur erlaubt. Für den Betrieb ist ein Soundkartenkanal erforderlich (Mikrofon+Lautsprecher), die Ansteuerung der GSM-Mobiltelefones erfolgt über die serielle/USB-Schnittstelle. Die Audioverbindung (Lautsprecher+Mikrofon) kann im einfachsten Fall über den Anschluß einer USB-Soundkarte für 5 Euro erfolgen.



GSM-Telefon (hier Sony Ericsson P910i) zum Anschluß an SvxLink

Diese Logic befindet sich im Experimentierstadium und ist momentan weder im Release noch im Trunk verfügbar. Bei Interesse bitte Kontakt über das Forum aufnehmen.

DTMF-Digit "D" Problem (2011-01-30 10:19)

Kommerzielle Geräte wie z.B. Motorolas GP1200 besitzen keine Taste "D", die den für das Auslösen eines Makros benötigten DTMF-Digit "D" erzeugt. Die Umprogrammierung der Tasten ist zwar möglich, führt aber zu einer nicht besonders ergonomischen Bedienung.

Eine Lösungsmöglichkeit für dieses Problem bietet das SEL5-Mapping. Hierbei wird der in SvxLink vorhandene 5-Ton-Selektivrufdekoder aktiviert und so konfiguriert, dass beim Empfang von 5-Tonfolgen innerhalb eines bestimmten Bereiches die Folgen als Makro erkannt werden.

Und hier die Konfigurationsschritte:

1) in der Repeater/Simplexlogik einen Bereich festlegen, in dem Makros erkannt werden sollen:

[RepeaterLogic]

...

SEL5_MACRO_RANGE=10000,40000

...

2) Im Abschnitt [Macros] wird eingetragen, was beim Empfang einer entsprechenden 5-Ton-Folge passieren soll:

[Macros]

...

10000=EchoLink:9999 #

10001=EchoLink:123343 #

...

39999=MetarInfo:10 #

...

3) im RX-Abschnitt jetzt noch den SEL5-Detector aktivieren, im Moment gibt es nur den eingebauten Software-SEL5-Detector

[Rx1]

...

SEL5_DET=1

SEL5_DEC_TYPE=INTERNAL

SEL5_TYPE=ZVEI1

...

Im GP1200 wird jetzt ein ZVEI-1-Selektivruf "10000" konfiguriert und es kann eine Bezeichnung Testserver hinterlegt werden. Beim Senden dieser 5-Ton-Folge wird SvxLink versuchen eine Verbindung mit dem Echolink-Testserver herzustellen.

Man ist übrigens nicht an 5 Töne gebunden, im Prinzip wären auch 10-Ton-Ruffolgen konfigurierbar.

6.2 Februar

Neuer Rauschperrentyp EVDEV (2011-02-25 08:27)

Tobias hat einen neuen Rauschperrentyp in den Trunk eingebaut: EVDEV. Dieser wertet ein Signal auf /dev/input/eventX aus und erstellt daraus ein Rauschperrenkriterium. Beispielsweise kann so der Status einer Taste auf einer USB-Soundkarte ausgewertet werden (z.B. als PTT bei MicSpkrLogic). Mal schauen wie das so läuft.

6.3 Mai

Release 11.05 veröffentlicht (2011-05-15 20:32)

Tobias hat heute das schon lange erwartete Release von SvxLink veröffentlicht. Dabei wurde die Nummerierung geändert, es handelt sich um die Version 11.05 (analog zum Ubuntu-Format 11 => 2011 und 05 => Mai)
Herunterladen kann man sich die Sourcen von hier:

<https://sourceforge.net/projects/svxlink/files/svxlink/11.05/>

Ich denke die lange Changelog-Liste spricht für sich, es hat sich eine ganze Menge getan. Es ist anzunehmen, dass auch kurzfristig fertige Pakete für die einzelnen Distributionen zur Verfügung stehen werden.

6.4 August

Automatische Installation von SvxLink (2011-08-25 16:53)

OM Martin (DF1AMB) hat eine einfache Möglichkeit geschaffen um ein Betriebssystem mit SvxLink praktisch ohne weitere User-Aktionen installieren zu können. Notwendig hierfür sind:

ein PC mit einer Platte, die komplett gelöscht werden darf
die Möglichkeit vom USB-Stick zu booten
etwas Zeit

Die Installation erfolgt fast von alleine, hier die Schritte:

USB-Stick anschließen
davon booten
1-2h nichts machen
fertig (bzw. SvxLink fertig konfigurieren)

Wir haben diese Variante schon mehrmals erfolgreich getestet. Gedacht ist diese Art der Installation sowohl für Newcomer, die keine Erfahrungen mit Linux haben als auch für Freaks, die eine einfache Lösung suchen um mal eben schnell SvxLink aufzusetzen. Anfragen diesbezüglich bitte per Forum (siehe auch <http://forum.svxlink.de/viewtopic.php?f=4&t=15>), weiterhin werden Beta-Tester gesucht!

vy 73's de Adi, DL1HRC

6.5 September

SvxLink-Abend in Weißenfels (2011-09-19 13:03)

Am 14.10.2011 findet ab 18Uhr der OV-Abend des OV W34 in Weißenfels statt. In diesem Rahmen wird Martin (DF1AMB) einen Vortrag zur automatischen Installation von SvxLink halten. Weiterhin wollen wir über aktuelle Entwicklungen in diesem Bereich informieren.

Gäste sind herzlich willkommen. Da die Räumlichkeiten nur einen begrenzten Teilnehmerkreis erlauben, würden wir uns über eine kurze Info freuen falls jemand aus anderen OV's die Absicht hat dabei zu sein (eMail an dl1hrc@svxlink.de).

Adresse:
Griechisches Restaurant "Äthos"

Markwerbener Str. 24
06667 Weißenfels

73's de Adi, DL1HRC

6.6 November

Neues SvxLink-Release, Version 11.11 (2011-11-06 19:51)

Tobias hat gestern ein neues Release von SvxLink herausgegeben, die Version 11.11.

Ein Changelog findet sich unter:

<http://sourceforge.net/projects/svxlink/files/svxlink/11.11/>

Für Ubuntu 11.10 steht bereits ein Binary zum Download bereit. Felix (WU8K) hat es gebacken:

<https://launchpad.net/felix.lechner/+archive/hamradio>

Ich werde in Kürze die Downloadseiten anpassen.

Viel Erfolg und Spaß mit SvxLink
wünscht Adi, DL1HRC

Neue SvxLink-Version 11.11.1 (2011-11-29 08:37)

Es gibt ein neues Release mit einigen Bugfixes. Nähere Informationen sind hier zu finden:

<https://sourceforge.net/projects/svxlink/files/svxlink/11.11/>

Für Ubuntu-Nutzer hat Felix Lechner Binaries gebacken, welche von hier installiert werden können:

<https://launchpad.net/felix.lechner/+archive/hamradio>

7. 2012

7.1 März

SvxLink-Server unter Ubuntu auf einem Pandaboard (2012-03-10 15:56)

Frank <DL7ATA> hat einen Artikel zur Installation von SvxLink unter Ubuntu auf einem stromsparenden Pandaboard geschrieben, sein Artikel ist unter Projekteßu finden:

SvxLink-Server unter Ubuntu auf einem Pandaboard

7.2 Juni

Internetstreaming ab Kernel Version 2.6.37 (2012-06-08 07:56)

Mathias (DO6EP) beschreibt unter http://svxlink.de/?page_id=1130 die Konfiguration von SvxLink für den Betrieb im Zusammenhang mit einem Streamingserver.

Neuer Artikel Parameter zur Laufzeit ändern”(2012-06-03 08:32)

Unter http://svxlink.de/?page_id=1105 kann man nachlesen, wie man bestimmte TCL-Variablen zur Laufzeit verändern kann.

SvxLink @ Hamradio FDH (2012-06-19 23:10)

Michael (DK5HH) wird am Sonntag, 24.6.2012 um 10Uhr auf der Aktionsbühne einen Vortrag zum Einsatz von SvxLink als Steuersoftware für Relaisfunkstellen halten.

Viel Erfolg Michael es vy 73 de Adi, DL1HRC

7.3 August

Neuer Branch "digitalmodes" (2012-08-08 13:01)

Ich habe heute auf Sourceforge einen neuen branch erzeugt. Er enthält zunächst einen Afsk-Dekoder (1200/2400Baud) um auf dem Sprachkanal ausgesendete APRS-Baken dekodieren zu können. Wer experimentierfreudig ist kann ihn gerne mal ausprobieren:

```
svn co https://svxlink.svn.sf.net/svnroot/svxlink/branches/digitalmodes svxlink-branch-digitalmodes
```

Hinweise: Ich empfehle den Betrieb mit einer internen Samplerate von 16000. Dafür in der Datei makefile.cfg den folgenden Eintrag

```
CXXFLAGS += -DINTERNAL_SAMPLE_RATE=8000
```

auf 16000 setzen und dann das Projekt neu kompilieren. Wenn der verwendete Rechner nicht gerade ein 386er ist sollte man generell auf 16k umstellen, die Qualität ist einfach besser...Achtung als Sounds müssen dann 16k-wav-Dateien verwendet werden.

Leider habe ich keine Möglichkeit 2400Bd Datenbursts zu erzeugen und bin deshalb an einem Feedback interessiert.

Die Einrichtung der LocationInfo ist für den Betrieb notwendig, ohne geht gar nichts!

73's de Adi, DL1HRC

SvxLink Streaming & GNU Radio (2012-08-22 11:34)

Om Michael (DK5HH) beschreibt unter http://svxlink.de/?page_id=1489 die Anbindung von SvxLink an GNU Radio über einen UDP-Port. Viel Spaß beim Lesen.

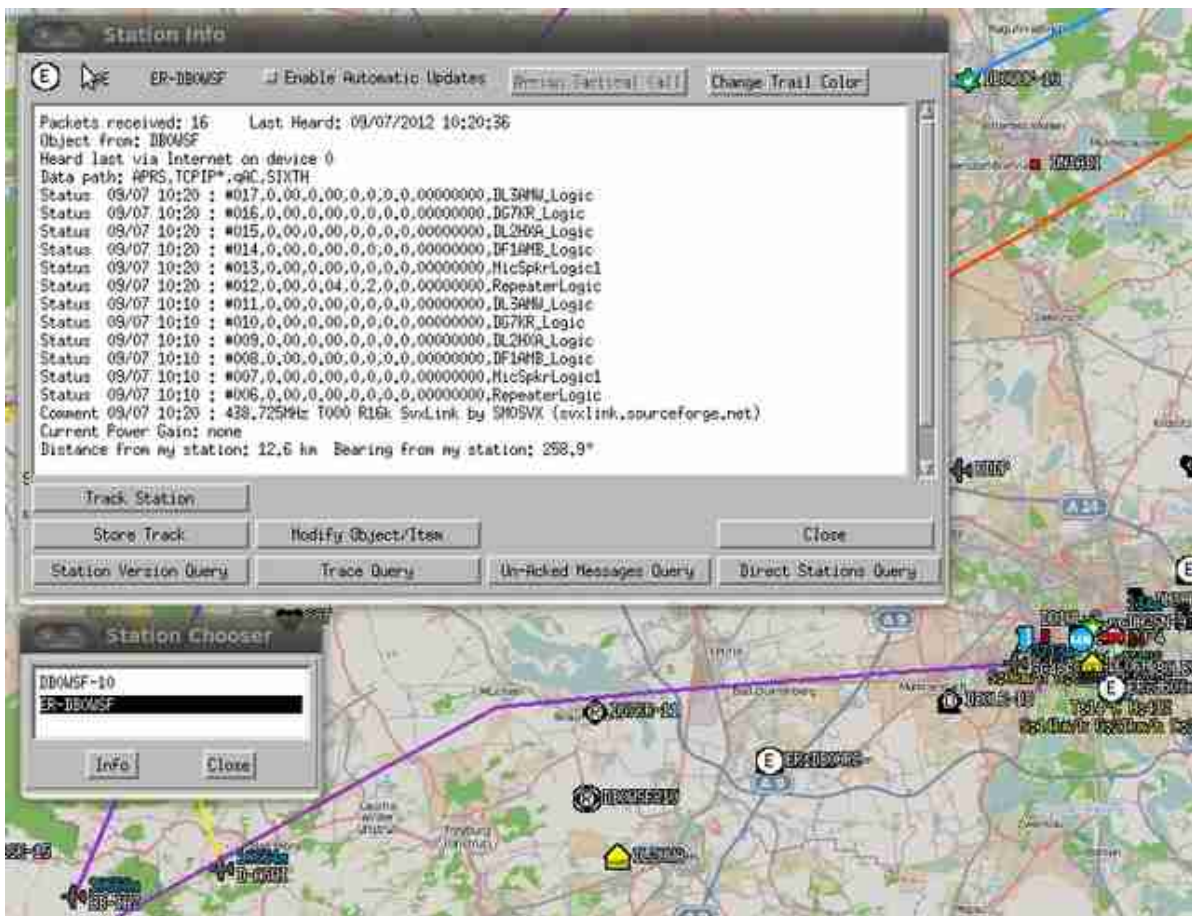
Informationsansagen zeitlich gesteuert mit cron (2012-08-23 18:37)

(monatlich) sich wiederholende Ereignisse, wie die Einladung zum OV-Abend oder andere auch nicht zeitabhängige Ansagen können als vorbereitete .wav-Dateien zu einem geeigneten Zeitpunkt abgespielt werden. – Vorschlag: kurz vor dem Abfallen des Relais, wenn keiner mehr sprechen will. mehr

7.4 September

APRS Statistik im neuen Trunk (2012-09-07 10:29)

Im neuen SvxLink-Trunk werden jetzt, falls LocationInfo aktiviert wurde, APRS-Statistiken erstellt und ins Netz eingespeist, die über die Nutzung des Repeaters/Links Aufschluss geben. In APRS-Programmen (wie z.B. XASTIR) werden diese Statistiken dann auch aufgeführt (Bild).



APRS Statistikdaten, in XASTIR als Status dargestellt

Für jede konfigurierte Logik wird ein einzelner Datensatz erstellt, der die Anzahl der Rauschsperröffnungen, TX-Aktivierungen und die Frequenzbelegung nach Erlang für RX und TX der letzten 10 Minuten logikspezifisch aufzeigt.

Sie Statistiken (in diesem Beispiel von DBOWSF) werden auf <http://aprs.fi/telemetry/a/ER-DBOWSF> aufbereitet

und graphisch dargestellt.

Artikel SvxLink auf Raspberry-Pi (2012-09-24 10:47)

Der Artikel aus der CQ-DL von om Stefan (DH5FFL) ist jetzt auch auf SvxLink.de verfügbar.

http://svxlink.de/?page_id=1606

Vielen Dank Stefan für Deine Arbeit.

8. 2013

8.1 Juni

2. SvxLink-Treffen in Halle/Saale 15./16.6.2013 - abgesagt (2013-06-05 14:28)

Auf Grund der Hochwassersituation muß ich leider das SvxLink-Nutzertreffen absagen. An meiner Hochschule stehen mehrere Gebäude unter Wasser. Wie die Situation nächste Woche aussieht kann ich momentan nicht sagen.



Hochwasser an der Burg

73 de Adi, DL1HRC

9. 2012

9.1 Dezember

Frohe Weihnachten (2012-12-20 17:57)

Das SvxLink-Team wünscht allen HAMs ein frohes und besinnliches Weihnachtsfest und einen guten Rutsch ins SvxLink-Jahr 2013.

73's de Adi, DL1HRC

10. 2013

10.1 März

Happy Birthday, SvxLink (2013-03-03 11:20)



Hi all,

am 8.3.2013 wird das SvxLink-Projekt 10 Jahre alt. Grund genug dem SvxLink-Vater Tobias (SMOSVX) ganz herzlich zu gratulieren und Ihm weiterhin viel Erfolg beim Weiterentwickeln und viele tolle Ideen zu wünschen.

Vy 73's de Adi, DL1HRC

Volodymyr (2013-03-04 18:50:09)

Happy Birthday - SvxLink! Success and further development! 73 de UR3QJW Volodymyr

Tim (2013-03-03 16:44:44)

Alles gute aus dem südlichen Schweden 73 de SA7BNT Tim

Professionelle Stimmen/Ansagen für SvxLink (2013-03-17 21:16)

Hi all,

wir SvxLink-Nutzer haben von der Linguattec GmbH München die Genehmigung zur Nutzung der professionellen Stimmen unter folgenden Bedingungen erhalten:

- 1) Es ist pro Sprache und Amateurfunkstation jeweils eine Home-Lizenz des Voice Reader von Linguattec notwendig.
- 2) Die Stimmen (Werke) werden ausschließlich auf Amateurfunkrelais/Links genutzt und auf den für den Amateurfunk freigegebenen Frequenzen, eine Übertragung in öffentliche Bereiche (Freenet, CB-Funk, LPD, Radio+Fernsehen) ist ausgeschlossen.

3) Die Stimmen (Werke) werden ausschließlich nicht-kommerziell genutzt

4) Die Genehmigung ist an die Nutzung der OpenSource-Software SvxLink gebunden, für andere Programme gilt diese Lizenz nicht.

Der Kauf einer Lizenz kann über den Webshop bei Linatec oder eBay erfolgen.

Für deutsche Nutzer habe ich einen svn erstellt, bei Interesse einfach anmailen.

Neues im Trunk (2013-03-29 08:27)

Tobias hat wieder ein paar Sachen implementiert, hier in Kurzform:

- `USE_GSM_ONLY=1` # verhindert, dass SvxLink bei EchoLink-Verbindungen in den SPEEX-Codec schaltet. Bei schwachen CPU's (Raspberry, Futro) gab es da in der Vergangenheit Probleme
- Verbessertes Logging in der `svxlink.log`
- `DEFAULT_LANG=` in der `ModuleEchoLink.conf` kann Ansagen ausschließlich für die Gegenstation ausgeben
- Qtel kann jetzt mit 16k Samples umgehen und der 100 %-CPU-bug wurde gefixt

Viel Spaß mit SvxLink

10.2 Juni

Neues Release 13.06 herausgegeben (2013-06-06 23:22)

Am 6.6.2013 hat Tobias/SMOSVX ein neues SvxLink-Release herausgegeben, die Version ist 13.06. Wichtige Neuerungen sind:

- standardmäßige Änderung der internen Samplerate von 8000 auf 16000 (Achtung, neue Soundfiles notwendig!), das betrifft auch Qtel!
- EchoLink besitzt von nun an ein Autoconnect-Feature (automatisches Verbinden per EchoLink mit einer vorkonfigurierten Station)
- `USE_GSM_ONLY` verhindert das Umschalten nach SPEEX beim Verbinden mit einem anderen SvxLink-Node (für schwache CPU's)
- verbesserte Pre-/Deemphasis

- Bugfixe in LocationInfo, ModuleEchoLink, Qtel und Async-Framework
- verbesserte Debugausgaben
- verbesserte Fehlerbehandlung für SQL _DELAY, SQL _START _DELAY und SQL _TIMEOUT

Auschecken mit

svn co svn://svn.code.sf.net/p/svxlink/svn/branches/releases/13.06 svxlink-release-13.06

GPIO-Funktionen (Raspberry Pi) sind noch nicht darin enthalten aber bereits im neuesten Trunk.

Viel Spaß es vy 73's de Adi, DL1HRC

HAMRADIO 2013 (2013-06-10 20:49)

Die folgenden Vorträge könnten von Interesse sein:

Samstag, 29.06.2013

09:30 Uhr

Konferenz-Zentrum Ost, Raum Rom

SDR mit GnuRadio

Grundlagen und Anwendung mit Funcube Dongle und DVB-T-Stick als 23 cm-Relais-RX mit SVXLink

Referent: Prof. Dr.-Ing. Michael Hartje, DK5HH

Folien und Dateien sind unter: im Ordner WS_Repeater

13:00 Uhr

Konferenz-Zentrum West, Raum Liechtenstein

Drei Amateurfunk-Anwendungen mit dem Raspberry Pi

Referent: Stefan Hüpper, DH5FFL

16:00 Uhr

Konferenz-Zentrum West, Raum Österreich

Digitale Sprachübertragung mit niedrigen Bitraten mit Hilfe von CODEC2 und Freedv

Referent: Prof. Dr.-Ing. Michael Hartje, DK5HH

Folien und Aduio-Dateien sind unter: im Ordner DVoice

Vielleicht ergeben sich am Rande dieser Vorträge mal Möglichkeiten für ein persönliches "Hallo". Martin/DF1AMB und Adi/DL1HRC haben sich schon mal die passenden T-Shirts besorgt ;)

10.3 Juli

SvxLink-Anfragen und mehr... (2013-07-18 18:10)

Hallo Leute,

ich freue mich wirklich sehr über das große Interesse und die vielen direkten Anfragen per eMail bzgl. SvxLink. Pro Woche bekomme ich momentan zwischen 20 und 30 Anfragen oder Problemlösungswünsche, an manchen Tagen alleine sogar 10 Mails mit teilweise sehr spezifischen Problemschilderungen und Erweiterungswünschen. Sorry, ich schaffe das fast nicht mehr.

Bitte habt Verständnis, dass ich nicht immer gleich eine tcl-Datei mit der gewünschten Funktion zurückliefern kann oder einen Lösungsvorschlag parat habe. Viele SvxLink-Funktionen habe ich selbst nicht probiert.

Bitte noch eine wichtige Sache berücksichtigen: Wenn Ihr Fragen/Probleme habt immer gleich eine mögliche Fehlermeldung mitsenden, das erspart Rückfragen.

Und auch bitte nicht in den Original-TCL-Dateien rumfummeln (Locale-Konzept berücksichtigen)! Ich habe sonst keinen Ansatzpunkt, denn die tcl-Ausgaben/Fehlermeldungen sind leider oft nicht aussagekräftig genug!

Und bitte nutzt auch verstärkt das Forum. Dafür habe ich das eingerichtet, eventuell weiß auch jemand anderes Bescheid. Und/Oder meldet Euch in der svxlink-devel-Mailingliste an, dort wird allerdings Englisch gesprochen. Keine Scheu, ich spreche die Sprache auch nicht ;-)

vy 73's de Adi, DL1HRC

Neues Release 13.07 (2013-07-30 07:19)

Tobias / SMOSVX hat am 29.07.2013 ein neues Release (Version 13.07) herausgegeben. Neuerungen sind:

- Unterstützung für EchoLink-Proxyserver
- ModuleEchoLink: Es ist jetzt möglich mehrere EchoLink-Server zu konfigurieren. SvxLink versucht die Verbindung zu den konfigurierten Servern aufzubauen, bis eine Verbindung hergestellt werden kann. Die Konfigurationsvariable SERVER wurde durch SERVERS ersetzt.
- Unterstützung für PTT an GPIO. Besonders interessant für embedded Systems (Raspberry Pi), die GPIO Pins bereitstellen und somit einen zusätzlichen USB-2-Serial-Adapter ersetzen. Patch von K1FSY / David.
- ModuleEchoLink: Eine neue Funktion `_receiving` wird aufgerufen, sobald eine EchoLink-Übertragung startet oder stoppt. Die Standard-Implementierung gibt einen Roger-beep über den lokalen TX aus. Patch von DL1HRC / Adi.
- Neues SSender-AusKommando. Mit der Variable SHUTDOWN _CMD kann man den Sender per DTMF-Kommando ein- bzw. ausschalten.

vy 73's de Adi, DL1HRC

10.4 August

SvxLink-Webbook 08/2013 (2013-08-28 09:31)

Das neue SvxLink-Webbook mit Stand 28.08.2013 kann unter Links (http://svxlink.de/?page_id=18 ganz unten) als pdf-Datei heruntergeladen werden.

10.5 September

make release" (2013-09-16 08:30)

Hi all,

an dieser Stelle noch einmal der Hinweis, dass beim selbst-kompilieren von SvxLink für ein Echtssystem make release aufgerufen werden sollte. Es werden hier einige Optimierungen durchgeführt. Bitte in jedem Fall diese Option beim Kompilieren auf dem Raspberry Pi und dem Futro anwenden und vorher die Compilerflags setzen! Weitere Informationen finden sich hier: http://svxlink.de/?page_id=1893

Damit kann man eine ganz erhebliche Reduzierung der CPU-Last erreichen (Bsp. Raspberry von >40 % auf ca. 15 %).

10.6 Oktober

Unterstützung des Opus-Codecs im Trunk (2013-10-13 11:48)

Tobias hat seit heute einen neuen Sprach-Codec implementiert. Es handelt sich um den offenen und freien Opus-Codec. Weitere Informationen sind hier zu finden <http://www.opus-codec.org/>

Dieser Codec besitzt eine geringere Latenz als Speex und bietet einen viel größeren Qualitätsbereich, von niedriger Sprachqualität/Schmalbandigkeit bis hin zu hoher Audio-Qualität und höheren Bitraten. Weitere Informationen auf der Wiki-Seite: http://de.wikipedia.org/wiki/Opus_%28Audioformat%29

Diese Unterstützung ist aber momentan auf die Verbindung zwischen SvxLink und den Remote-TRX'en beschränkt. Für die Verbindung zwischen zwei SvxLink-Stationen via EchoLink wird nach wie vor nur GSM und Speex unterstützt (Stand 13.10.2013). Ich denke aber, dass das nicht lange auf sich warten läßt.

SvxLink @ INTERRADIO 2013 (2013-10-25 08:36)

Auf der diesjährigen Interradio am 30.11.2013 in Hannover wird es einen Vortrag zum Thema Moderne Relaissteuerung mit SvxLink"geben. Im Plan steht (noch) Raspberry Pi, allerdings war mir das ehrlich gesagt nicht umfangreich genug und etwas zu einschränkend.

Der Vortrag dauert etwa 1h, danach ist genügend Zeit für einen Meinungs- und Erfahrungsaustausch.

Ort: Hannover-Messe Gelände

Datum: 30.11.2013

Zeit: 13:30 - 15:30

Raum: Glasübergang Ecke Nord/Ost Ende

vy 73's de Adi, DL1HRC

10.7 November

SvxLink @ INTERRADIO 2013 -Update- (2013-11-11 20:02)

Dieser Vortrag wird durch einen eigenen kleinen Vortrag von Michael (DK5HH) ergänzt, in dem er vor allem auf die Konfiguration von Soundkarten incl. Kalibrierung selbiger für den Betrieb unter SvxLink eingehen wird.

noch mal: 30.11.2013, 13:30Uhr, Hannover, Messengelände, Glasübergang Ecke Nord/Ost

SA7BNT (2013-11-12 18:28:04)

...nicht anwesende würden sich über ein Video file freuen. SA7BNT

dl1hrc (2013-11-12 20:13:21)

hm, mal sehen was ich machen kann.

TetraLogic für DMO-Repeater (2013-11-17 18:03)

Seit 16.10.2013 läuft in Halle/Saale ein TETRA-DMO-Repeater auf 430.375MHz mit einer PEI-Anbindung zu SvxLink. Ziel ist, diesen DMO-Repeater mit dem EchoLink-Netz zu verbinden und einen entsprechenden Gateway TETRA/Amateurfunk <-> EchoLink-Netzwerk bereitzustellen.

Positioning (APRS) und SDS-Auswertungen sind bereits implementiert. Etwas detailliertere Informationen sind unter Projekteßu finden.

Weitere Informationen folgen. An- und Rückfragen bitte an dm0svx [ät] svxlink.de

Vorträge INTERRADIO 2013 (2013-11-30 17:49)

Vielen Dank für das Interesse während unserer Vorträge. Michael und ich hoffen, dass es für alle Zuhörer interessant war und wir vielleicht auch ein paar neue Informationen an den Mann bringen konnten.

Download:

Präsentation DL1HRC

Präsentation DK5HH

Eine frohe Adventszeit wünschen Michael, DK5HH und Adi, DL1HRC

10.8 Dezember

Release 13.12 (2013-12-01 17:03)

Seite heute ist das SvxLink-Release 13.12 verfügbar. Hier die Neuerungen:

- verbessertes LogicLinking
- verbesserter QSO-Rcorder
- Support für GPIO-Squelch (z.B. Raspberry-I/O's)
- verbesserte Pre/Deemphasisfilter
- Opus-Support für Rempte-TRX's

Der Dateiname der Soundausgaben wurde wie folgt geändert:

svxlink-sounds-lang _code-voice name-samplerate-svxlink-version.tar.bz2

Beispiel:

svxlink-sounds-en _US-heather-16k-13.12.tar.bz2

Frohe Weihnachten und alles Gute für 2014 (2013-12-24 18:59)

Ich wünsche allen Lesern ein frohes und besinnliches Weihnachtsfest und einen guten Rutsch ins SvxLink-Jahr 2014.

vy 73's de Adi, DL1HRC

DL4OCH (2013-12-25 00:07:33)

Hallo Adi, vielen Dank für die super Website und erst recht für das Forum. Ich wünsche Dir auch eine frohe und besinnliche Weihnacht mit wenig Stress. vy73 de Wilm, DL4OCH SYSOP: DB0EIG und DB0OR-L

SA7BNT (2013-12-25 09:52:54)

Auch dir eine frohe Weihnacht, und einen guten Rutsch ins Jahr 2014. 73 SA7BNT

DG1KWA (2013-12-27 15:00:10)

Dir auch Adi. 73 de Andreas DG1KWA

11. 2014

11.1 Januar

Änderungen (2014-01-30 10:28)

Es hat im neusten Trunk (Version 1.4.0) eine kleine Änderung für das Ansprechen der PTT gegeben, was auf die bessere Integration der GPIO-Ports (Raspberry-Pi) zurückzuführen ist.

PTT_PORT definiert wie vorher die Schnittstelle (z.B. /dev/ttyS1)

PTT_TYPE definiert die Art des Ports, gültige Werte sind: NONE", "GPIO" oder "SSerialPin"

PTT_PIN definiert den Pin (DTR, RTS usw.)

Freier Satz für Italienisch (2014-01-30 10:32)

Für die Freunde der italienischen Sprache biete ich hier den Download einer it-Sprachumgebung für SvxLink (22.05k, 16k und 8k) an:

<http://svxlink.de/download/svxlink-sounds-22k.tar.bz2>

<http://svxlink.de/download/svxlink-sounds-16k.tar.bz2>

<http://svxlink.de/download/svxlink-sounds-8k.tar.bz2>

Diese Umgebung ist frei für die Nutzung innerhalb von SvxLink (Acapella-Sound's).

73's de Adi, DL1HRC

11.2 Februar

HAMRADIO 2014 - Vorträge (2014-02-26 23:03)

Am Samstag, 28.06.2014 gibt es auf der HAMRADIO zwei Vorträge mit einem Bezug zum SvxLink-Projekt bzw. digitaler Sprachübertragung (FreeDV):

Thema: FreeDV

Vortragender: Prof. Dr.-Ing. Michael Hartje/DK5HH

Zeit: 14:00Uhr-15:45Uhr

Ort: Raum 2 Halle A2

Thema: Moderne Relaissteuerung mit SvxLink

Vortragender: Adi Bier/DL1HRC

Zeit: 16Uhr-17:45Uhr

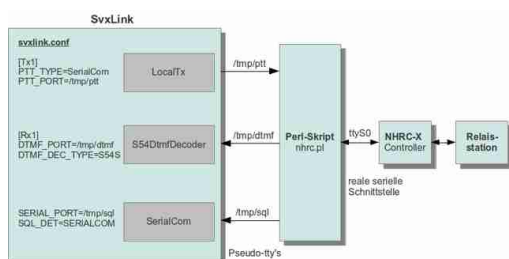
Ort: Raum 2 Halle A2

Das gesamte Tagungsprogramm der HAMRADIO 2014 ist noch nicht veröffentlicht und o.g. Termine verstehen sich unter Vorbehalt!

11.3 März

Erweitertes Konzept für SQL,DTMF und PTT (2014-03-05 17:30)

Momentan wird eine Erweiterung für das Andocken von Repeater-Controllern getestet. Die folgende Abbildung soll das Funktionsprinzip verdeutlichen:



SvxLink-NHRCx-Erweiterung

Das Perlskript wird hier verwendet um die Informationen, die über eine einzelne serielle Schnittstelle vom/zum Controller gesendet/empfangen werden auf die einzelnen Objekte (PTT, DTMF und Squelchdetector) aufzuteilen". Diese Variante ermöglicht auch eine Steuerung von SvxLink durch eigene Skripte. Das Protokoll ist relativ einfach gehalten:

- PTT: T=TX ein, R=TX aus
- SQL: O=Squelch ist offen, Z=Squelch ist geschlossen
- DTMF: 0...9* #ABCD ->Übertragung der DTMF-Zeichen an SvxLink

11.4 Februar

Bootfähiges Image Voyage-Linux (incl. SvXLink) für ThinClients (2014-02-28 15:50)

Unter http://svxlink.de/?page_id=2284 hat Michael/DK5HH eine Anleitung erstellt wie man auf einer CF-Karte ein bootfähiges Linux inklusive einem relativ aktuellem SvXLink installieren kann. Diese Variante wurde auf Thin-Clients wie z.B. Futro A2xx getestet.

11.5 Mai

Erweiterung um einen Control-Port (2014-05-31 19:58)

Die Erweiterung in München bei DB0EL scheint gut zu funktionieren. Mal schauen, wann es im Trunk landet. Ich habe noch eine weitere Extension geplant, die es ermöglicht bestimmte Parameter zur Laufzeit zu ändern ohne SvXLink neu starten zu müssen. Mehr dazu in Friedrichshafen auf der HAMRADIO.

11.6 Juni

HAMRADIO-Vorträge von DK5HH und DL1HRC (2014-06-29 12:41)

Die Vorträge von Michael/DK5HH und Adi/DL1HRC kann man in Kürze von hier herunterladen:

Moderne Repeatersteuerung mit SvXLink (Adi / DL1HRC)

Verzeichnis zum Vortrag FreeDV – SSB-TRX fit machen (Michael / DK5HH)

Verzeichnis zum Beitrag zu SVXLink (Michael / DK5HH)

11.7 Juli

APRS-SysOp Treffen Auersberg (2014-07-10 12:29)

APRS SysOp Treffen beim 3. Relaietreffen DBØERZ/Auersberg

Termin:

Sonnabend, 20.09.2014 ab 10Uhr

ACHTUNG: Durch die Veranstaltung "3-Talsperren Marathon"

verschiebt sich der Zeitplan für geplante Vorträge etwas. Beginn der Vorträge erst 13Uhr

Ort:

Auersberg, JO60HK, 1019m ü. NN

Themen:

- 10:00 Uhr Beginn für Früh-Angereiste, Zeit für Diskussion/Planung neuer Projekt
- bis ca. 12:00 Uhr Mittagspause (<http://www.auersberg-erzgebirge.de/>)
- ab 13:00 Uhr - Beginn der Vorträge

- WX3in1 Plus / Mini ein Standalone Igate/Digi mit WX und Telemetrie live (DH8TOM)
- Wetterstation Davis Vantage Pro2 im APRS Betrieb (DH8TOM)
- APRX als IGate mit Router oder RPi (DM1HF & DH8TOM) / Renaissance alter 19Technik

- 14:00 Uhr - FM-Repeater und -Links mit SVXLink (DL1HRC)
- 14:45 Uhr - Diskussion, Meinungsfindung
- 15:00 Uhr - Digitalrelais: DStar oder DRM? (DM8WP)
- 15:45 Uhr - Diskussion & Meinungsfindung
- 16:15 Uhr - Eine Profi-Wetterstation mit SPS zum Leben erwecken (DL3YK)
- 17:00 Uhr VON und FÜR SysOps: Konfiguration von Igates/Digis auf APRX-Basis (Linux) - (DM1HF & DH8TOM),
Tips für Tetrabetrieb sowie Installation/Einrichtung SvxLink-Link/Repeater unter Linux (DL1HRC)
- 18:00 Uhr - abendlicher Ausklang

Mehr Informationen gibt es in Kürze auf <http://aprs-dl.de/?Aktuell>, www.db0erz.de, www.db0fib.de

Bitte meldet euch vorher per E-Mail sysop-treffen@dh8tom.net an, damit wir und die Bewirtschaftung vor Ort entsprechend vor planen können.

Links:

Relaisstation DB0ERZ - www.db0erz.de

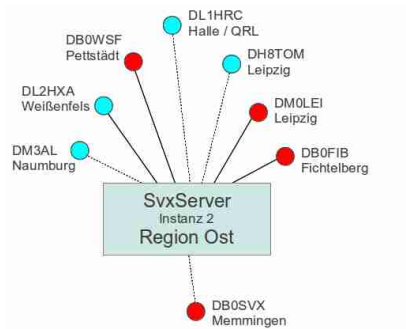
Berggasthof Auersberg - www.auersberg-erzgebirge.de

11.8 September

SvxLink-Relaisverbund (2014-09-24 11:37)

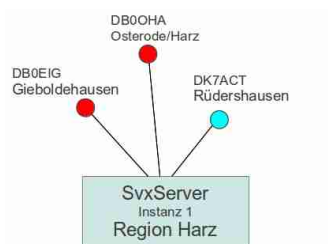
Seit einigen Wochen gibt es einen kleinen IP-basierten Relaisverbund in Mitteldeutschland, an dem temporär auch das süddeutsche Relais DBOSVX in Memmingen angeschlossen ist. Das Ziel der Relais-Belebung wurde damit erreicht.

Ein paar Problemchen sind noch zu lösen aber prinzipiell funktioniert es schon ganz gut.



SvxLink-Relaisverbund Ost

Einen weiteren Verbund gibt es im Harz, Ansprechpartner hier ist Wilm (DL4OCH).



SvxLink-Relaisverbund Region-Harz

11.9 November

Interradio 2014 Hannover -Vorträge- (2014-11-09 09:30)

Hier die Vorträge von Michael/DK5HH und Adi/DL1HRC zum Thema Audioverarbeitung und SvxCLink.

DK5HH kommt noch
INTERRADIO _Hannover _2014 _svxCLink

Vielen Dank für das große Interesse, die Zeit war leider doch etwas knapper als ich gedacht habe. Die OM's die mich gezielt nach Software (wie SvxCLink-Server) gefragt hatten, bitte ich mir mal direkt eine eMail zu schreiben (MEINCALL@gmx.de oder @darc.de geht auch).

vy 73's de Adi, DL1HRC

11.10 Dezember

Eine frohe Weihnachtszeit (2014-12-10 15:56)

Es hat sich wieder einiges getan im letzten Jahr. Mal schauen was das SvxLink-Jahr 2015 so alles bringt. Allen Nutzern wünsche ich eine tolle Adventszeit, ein besinnliches Weihnachtsfest und schon mal einen guten Rutsch!

vy 73's de Adi, DL1HRC

12. 2015

12.1 März

Workaround bei EchoLink-Problemen (2015-03-23 06:21)

In letzter Zeit wird von vielen Nutzern vermehrt über Probleme bei der Nutzung von EchoLink-Gateways über Proxyserver welche sich auch im HAMNET befinden berichtet. Hier ich eine Anleitung (En) von Jann, DG8NGN.

Download



gads

BlogBook v0.5,
L^AT_EX 2_ε & GNU/Linux.
<http://www.blogbooker.com>

Edited: 29. April 2015

