

# Dire Wolf APRStt Configuration Example

## Listening on Repeater

December 2015

Seen in the APRSSIG discussion group:

**Robert Bruninga** [bruninga at usna.edu](mailto:bruninga@usna.edu)

*Tue Nov 24 21:00:59 CST 2015*

---

We just put the APRStt (touchtone) transponder on the air in Annapolis for locals to try out the new QIKCOM-2 satellite transponder. But there's more!... WoW!

Here is the announcement, and then at the end is the brand new idea how we will use this terrestrially while waiting for the satellite to be launched!

----- First, the announcement -----

The next Ham DTMF satellite is on the air for testing in Annapolis.

Anyone can use this transponder from their DTMF radio. (don't need an APRS radio). Just enter your DTMF callsign and GRID into a DTMF memory (formatted per the web page below) and send it on the 144.33 uplink and listen for the voice response on 144.39 (amongst all the packet clutter too).

Not only does it convert the DTMF to a voice CQ, but it also sends back an APRS packet copy of the CQ on 144.39 into the global APRS system. If successful, you should see it in the ten minute Bulletin update:

<http://aprs.fi/info/a/QIKCOM-2>

So just program your DTMF memory once, and try it any time you drive by Annapolis (its on my building roof at USNA).

And not only can you send your grid, you can send any of the ARL radiograms both emergency and general, but any of a few dozen more typical messages just by entering the 2 digit message number in DTMF followed by your DTMF callsign...

See how on <http://aprs.org/qikcom-2.html>

We just delivered the flight unit to the launch integrator 18 Nov, and so now we can sit back and actually TEST the prototype on the air and see how it works under user load. It is only 4 Watts, so you probably have to be within a few miles or so. Something to entertain you while sitting in traffic. I have worked it from a high spot in Glen Burnie (bridge over Rt 10) about 12 miles away, but you have to have a good shot to Annapolis.

You only have to enter your grid and call into your DTMF memory once. Then, anytime you are mobile near the Severn bridges or near the Academy, just send out the DTMF memory and listen for the response.

If there are thunderstorms in the area, I should disconnect it... but will try to leave it on.

-----Now the new idea! -----

But here is the NEW idea of how to use its 2 digit message capability locally...

- 1) Have it listen from a very high central metro area location for DTMF on 146.58
- 2) The two-digit messages (01 to 99) can be used to trigger the APRS and voice response of "CALLSIGN, LISTENING ON XXX.XXX MHz"
- 3) then ALL APRS mobiles in the area will see the message on their radios
- 4) AND the messages go worldwide in the APRS-IS system
- 5) You could be called back via Echolink for example
- 6) AND if anyone is monitoring 146.58, they will hear EVERYONE on the air and where they are listening.

In our area with over 50 repeaters, we are so spread out, hardly any repeater is being used. But this way, everyone's "presence" on the air is announced on APRS and the single 146.58 just like APRS was always intended to be... a single clearing house channel for everything happening in ham radio.

But now you only need a DTMF radio. Not a full APRS rig.

Oh, we just make a local list 01 to 99 of the different frequencies. Every local list is different, but so are the repeaters in every local area... Though I am starting to think it can be quite universal! Wow, I'll start now... for example 76 is 146.76 and 94 is 146.94, etc!

You just program your DTMF memory with the few repeaters you use, and then sending out your message is just one button. Send the DTMF memory with your chosen "monitoring frequency!"

for more info on APRStt see <http://aprs.org/aprstt.html>

Bob Bruninga, WB4APR

USNA Satellite Lab

## You *CAN* try this at home

Don't have your own **QIKCOM-2 Satellite Transponder**? No Problem. You can do the same thing with an ordinary computer and the APRStt gateway built into Dire Wolf. Here's how.

First download the Dire Wolf application from: <https://github.com/wb2osz/direwolf/releases>

Refer to the latest documentation: <https://github.com/wb2osz/direwolf/tree/dev/doc>

Some of the features used here are new in version 1.3 (not out yet) so you will want the latest development snapshot.

In this example, I use a Windows PC but you can use Linux instead with only a couple minor changes.

We need two audio interfaces for the two radio channels.

- APRStt / voice channel:

USB audio adapter.

COM1 port RTS line is used to activate the transmitter (PTT).

- APRS channel:

Built-in soundcard on motherboard.

COM1 port DTR line for PTT.

Next, use your favorite editor to construct a configuration file with some meaningful name such as listen.conf.

## Configuration File – listen.conf

```
#
# Sample Dire Wolf configuration file to approximate
# the behavior described here:
#
# http://www.tapr.org/pipermail/aprssig/2015-November/045030.html
#
# We listen for 10 digit callsigns followed by 2 more
# digits representing a pre-defined local voice repeater.
#
# The voice response, on the same channel, is something like:
#
#     W B 2 O S Z listening 146.955
#
# An object report is then sent out on the APRS channel.
# It looks something like this:
#
# WB2OSZ-15>APDW13:;WB2OSZ-12*201915z4235.  N/07126.  WC146.955MHz T074 http://www.wb1gof.org !T95!
#

##### Define the Radio Channels #####

#
# Channel 0 is used for the APRStt frequency.
# Touch Tone reception and Speech response.
#
# Espeak doesn't seem to have a way to direct sound to a
# particular device.  It always uses the default audio device.
# When we plug in a USB audio adapter, it becomes the default.
# Therefore, the default audio device must be used for the
# channel with voice response.
#

ADEVICE0 USB
CHANNEL 0
PTT COM1 RTS

#
# Channel 2 is for the normal APRS channel.
#
# In my case, "Realtek High" is used to select the "soundcard"
# built into the motherboard.
#

ADEVICE1 "Realtek High"
CHANNEL 2
MODEM 1200
PTT COM1 DTR

MYCALL WB2OSZ-15

##### Configure the APRStt Gateway #####

# Define locations for Position Reports.
# Not actual repeater locations but should be in right town.

TTPOINT B995 42.59 -71.44
TTPOINT B912 42.6 -71.29

# Define format for position ambiguity.

TTAMBIG BAx

# Rules for handling tone sequences of exactly 12 digits,
```

```

# terminated by the final # of course.
# The first 10 digits are the user's callsign.
# The last 2 digits get mapped into a frequency and location.

# We generate an object with:
#   The user's callsign.
#   Canoe & horse symbols just for fun.
#   Approximate location of the repeater.
#   Remove two digits lat/long digits for ambiguity.
#   Frequency of the specified repeater.
#   CTCSS tone. (integer part, leading 0 if necessary to make 3 digits.)
#   Comment with repeater information.

TTMACRO xxxxxxxxxx95 ACx*AB{canoe}*B995*BA2*C146955*C074*CA{http://www.wblgof.org}
TTMACRO xxxxxxxxxx12 ACx*AB{horse}*B912*BA2*C147120*C103*CA{http://www.wlhh.org}

# Handle the shorter 5 digit suffix form as well.
# Full call is substituted if available.

TTMACRO xxxxx95 ACx*AB{canoe}*B995*BA2*C146955*C074*CA{http://www.wblgof.org}
TTMACRO xxxxx12 ACx*AB{horse}*B912*BA2*C147120*C103*CA{http://www.wlhh.org}

# Receive touch tones on channel 0 and respond with voice there.
# Transmit APRS object reports on channel 2 via path specified.
# IGates should not send it to Internet Servers.

TTOBJ 0 2 WIDE1-1,RFONLY

##### Scripts to generate speech responses #####

# This one selects the specific speech synthesizer we are using.
# It is included with the standard distribution.

SPEECH dwespeak.bat

# Custom to produce the desired words when valid tone sequence is received.
# For errors, the specified responses listed below are used.

TTCMD listen.pl

##### Finally specify voice responses #####

TTERR OK SPEECH "Message Received O K."
TTERR D_MSG SPEECH "D type not implemented."
TTERR INTERNAL SPEECH "Internal error."
TTERR MACRO_NOMATCH SPEECH "No matching pattern for digit sequence."
TTERR BAD_CHECKSUM SPEECH "Bad checksum on call."
TTERR INVALID_CALL SPEECH "Invalid call sign."
TTERR INVALID_OBJNAME SPEECH "Invalid object name."
TTERR INVALID_SYMBOL SPEECH "Invalid symbol."
TTERR INVALID_LOC SPEECH "Invalid location. "
TTERR NO_CALL SPEECH "No call or object name."
TTERR INVALID_MHEAD SPEECH "Invalid Maidenhead Locator."
TTERR INVALID_SATSQ SPEECH "Satellite square must be 4 digits."
TTERR SUFFIX_NO_CALL SPEECH "Send full call sign before using suffix."

```

Next construct a file to generate the desired speech response.

## Speech Response Script – listen.pl

```
#!/usr/bin/perl

# Simple script to generate custom speech responses for

# In this case we want to speak:
#   (callsign as separate letters) listening (frequency)

# First, gather information of interest.

$call = $ENV{TTCALLSP};
$freq = $ENV{TTFREQ};

# Remove "MHz" from the frequency.

$freq =~ s/MHz//;

# Combine into string to be spoken.

print "$call listening $freq\n";
```

## Let's try it!

Convert your callsign to the 10 digit representation with the included "text2tt" utility. I happen to be using a Cygwin command window which explains the \$ prompt.

```
$ text2tt wb2osz

Push buttons for multi-press method:
"922A222266677779999"    checksum for call = 3
Push buttons for two-key method:
"9A2B26C7D9D"          checksum for call = 4
Push buttons for fixed length 10 digit callsign:
"9226711598"
```

Remember that last line.

Start up Dire Wolf with the proper configuration file:

```
$ direwolf -c listen.conf -a 60
```

On the APRStt / voice frequency, transmit your 10 digit callsign, either 95 or 12 for the repeater, and finally # to indicate end of tone sequence.

```
9226711598 95 #
```

Watch the screen to see how this gets transformed.

```
DTMF audio level = 3 tt
[0.dtmf] DTMF>APDW13;t922671159895#
Raw Touch Tone Data, DireWolf, WB2OSZ
922671159895#
Macro tone sequence: '922671159895'
Matched pattern 3: 'xxxxxxxxx95', x=9226711598, y=, z=
Replace with: 'ACx*AB135*B995*BA2*C146955*C074*CA728484802615158787871487661771797014798271'
After substitution: 'AC9226711598*AB135*B995*BA2*C146955*C074*CA728484802615158787871487661771797014798271'
[APRStt] WB2OSZ-15>APDW13;;WB2OSZ-12*201915z4235. N/07126. WC146.955MHz T074 http://www.wb1gof.org !T95!
[0.speech] "W B 2 O S Z listening 146.955 "
[2L] WB2OSZ-15>APDW13,WIDE1-1,RFONLY;;WB2OSZ-12*201915z4235. N/07126. WC146.955MHz T074 http://www.wb1gof.org !T95!
```

What is this trying to tell us?

- [0.dtmf]... Indicates channel 0 DTMF reception.
- The incoming tone sequence matched the xxxxxxxxxxx95 pattern.
- After substitution, it becomes the equivalent sequence you could send over the air but wouldn't want to because it is so tedious.
- [0.speech]... means it is transmitted on channel 0 as synthesized speech.
- [2L]... means it is transmitted on channel 2 as an AX.25 frame.

Let's try running that through the "decode\_aprs" utility to see if it was formed correctly.

```
$ echo 'WB2OSZ-15>APDW13,WIDE1-1,RFONLY;;WB2OSZ-12*201915z4235. N/07126. WC146.955MHz T074 http://www.wb1gof.org !T95!' | decode_aprs
WB2OSZ-15>APDW13,WIDE1-1,RFONLY;;WB2OSZ-12*201915z4235. N/07126. WC146.955MHz T074 http://www.wb1gof.org !T95!
Object, "WB2OSZ-12", Canoe, Direwolf, WB2OSZ
N 42 35.0000, W 071 26.0000, APRStt location 95 of 100, 146.955 MHz, PL 74.4
http://www.wb1gof.org<0x20>
```

Looks good.

After the full callsign is memory, we can use the shorter suffix notation. Let's try it.

```
67162 12 #
```

Watch the screen to see how this gets transformed.

```
DTMF audio level = 4 tt
[0.dtmf] DTMF>APDW13;t6716212#
Raw Touch Tone Data, DireWolf, WB2OSZ
6716212#
Macro tone sequence: '6716212'
Matched pattern 6: 'xxxxx12', x=67162, y=, z=
Replace with: 'ACx*AB169*B912*BA2*C147120*C103*CA72848480261515878787148717727214798271'
After substitution: 'AC67162*AB169*B912*BA2*C147120*C103*CA72848480261515878787148717727214798271'
Suffix "OSZ" was converted to full callsign "WB2OSZ"
[APRStt] WB2OSZ-15>APDW13;;WB2OSZ-12*202018z4236. N/07117. We147.120MHz T103 http://www.w1hh.org !T12!
[0.speech] "W B 2 0 S Z listening 147.120 "
[2L] WB2OSZ-15>APDW13,WIDE1-1,RFONLY;;WB2OSZ-12*202018z4236. N/07117. We147.120MHz T103 http://www.w1hh.org !T12!
```

If there was not a full callsign in memory, matching the suffix, we hear the spoken message, "Send full call sign before using suffix."

```
DTMF audio level = 3 tt
[0.dtmf] DTMF>APDW13;t2772212#
Raw Touch Tone Data, DireWolf, WB2OSZ
2772212#
Macro tone sequence: '2772212'
Matched pattern 6: 'xxxxx12', x=27722, y=, z=
Replace with: 'ACx*AB169*B912*BA2*C147120*C103*CA72848480261515878787148717727214798271'
After substitution: 'AC27722*AB169*B912*BA2*C147120*C103*CA72848480261515878787148717727214798271'
Couldn't find full callsign for suffix "APR"
[0.speech] "Send full call sign before using suffix."
```

## Summary

Very few hams have portable equipment for APRS but nearly everyone has a handheld radio that can send DTMF tones. APRStt allows a user, equipped with only DTMF (commonly known as Touch Tone) generation capability, to enter information into the global APRS data network.

The Dire Wolf implementation of APRStt is very flexible and is easily customized for a wide variety of different situations by changing the configuration file.