

An Introduction to Software Defined Radio

Pamela O'Shea, @pamoshea



Contents

1. Hardware
2. Software
3. Planes
4. Ships
5. Pagers
6. Home devices



powerpig /for/ GIZMODO

1. Hardware

- RTL-SDR Dongle: Great value receiver!
- Realtek RTL2832U/R820T Tuner Receiver
- Freq range: 24 MHz - 1.8 GHz
- Receive only
- \$10 AUD



1. Hardware

- HackRF One
- Transmit & Receive
- Half Duplex
- Freq Range: 1 MHz - 6 GHz
- \$520 AUD



1. Hardware

- HackRF One Portapack
- Add on
- Come see at our equipment table after the talks



1. Hardware

- USRP
- New model USRP E312
- Freq range: 70 MHz - 6 GHz
- \$5,220 AUD



2. Software

- **GNU RADIO**

The screenshot shows the GNU Radio wiki homepage. At the top, there's a browser header with back, forward, and search buttons, and the URL gnuradio.org/redmine/projects/gnuradio/wiki. Below that is a dark navigation bar with links for Home, Projects, and Help. The main content area has a header "Welcome to GNU Radio!" and a sub-header "Introduction". A text block explains that GNU Radio is a free & open-source software development toolkit for signal processing. It's licensed under the GNU General Public License (GPL) version 3. The "Content" sidebar on the right lists sections like "Getting started", "Documentation", and "Community & Communicating". At the bottom, it says "GNU Radio has two manuals: one for the C++ API and another for the Python API. The majority of the documentation comes from using [Doxygen](#) markup comments in the public header files. These are the basis for both manuals. The Python documentation uses [Sphinx](#) to pull in both the Doxygen documentation as well as any formatted comments present in any Python files."

gnuradio.org/redmine/projects/gnuradio/wiki

Home Projects Help

GNU Radio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

Overview Activity Roadmap Issues News Wiki Files Repository

Welcome to GNU Radio!

History

Introduction

GNU Radio is a free & open-source software development toolkit that provides signal processing blocks to implement software radios. It can be used with readily-available low-cost external RF hardware to create software-defined radios, or without hardware in a simulation-like environment. It is widely used in hobbyist, academic and commercial environments to support both wireless communications research and real-world radio systems.

GNU Radio is licensed under the GNU General Public License (GPL) version 3. All of the code is copyright of the Free Software Foundation.

Content

I. Getting started

If you've never touched GNU Radio before, these pages will get you started with a running installation of GNU Radio and will show you how to take your first steps with this software radio tool.

- [What is GNU Radio and why do I want it?](#) - Read this if you really have no idea what this project is about.
- [Installing GNU Radio](#) - This will explain all the steps to get a working installation of GNU Radio.
- [How do I use GNU Radio?](#) - A short introduction to the possibilities you have as a GNU Radio user.
 - [Utilities and tools that come with GNU Radio](#)
- [Tutorials](#)
 - [Guided Tutorials](#)
 - [How to write Python applications](#) - This includes a guide on how to read and use the Doxygen-generated API docs.
 - [A quick guide on doing simulations with GNU Radio](#)
 - [How to write an out-of-tree \(OOT\) module](#)
 - [Tutorial on how to configure OOT packages to find and link against GNU Radio](#)
- [Frequently Asked Questions](#) - Check this page before asking questions on the mailing list.

II. Documentation

GNU Radio has two manuals: one for the C++ API and another for the Python API. The majority of the documentation comes from using [Doxygen](#) markup comments in the public header files. These are the basis for both manuals. The Python documentation uses [Sphinx](#) to pull in both the Doxygen documentation as well as any formatted comments present in any Python files.

2. Software

- **Linux**

- **GNU Radio Live:**

- <https://gnuradio.org/redmine/projects/gnuradio/wiki/GNURadioLiveDVD>

- **Pentoo:** <http://www.pentoo.ch>

- **Kali:** <https://www.kali.org>

- **Ubuntu:** <http://www.ubuntu.com>

- **Windows:**

- Not covering Windows this meetup but SDR# is a good free tool of choice for Windows (same as gqrx on Linux)

- Lots of easy to use tools on Windows

- Check out Hak5 for tools on Windows

2. Software

- **GQRX** on Linux
- This is also supplied in our Live USB Dongles
- Tune into FM radio

3. Planes



3. Planes – ADSB

- **ADSB:** Automatic dependent surveillance – broadcast
- Radar replacement
- Aircraft gets position from satellite and broadcasts it for tracking
- No encryption or authentication
- 1090 Mhz (or 978 Mhz)

3. Planes – dump1090: list aircraft

- Tool: **dump1090**

<https://github.com/antirez/dump1090>

```
dump1090 --interactive --aggressive
```

3. Planes – dump1090 Viewing on a Map

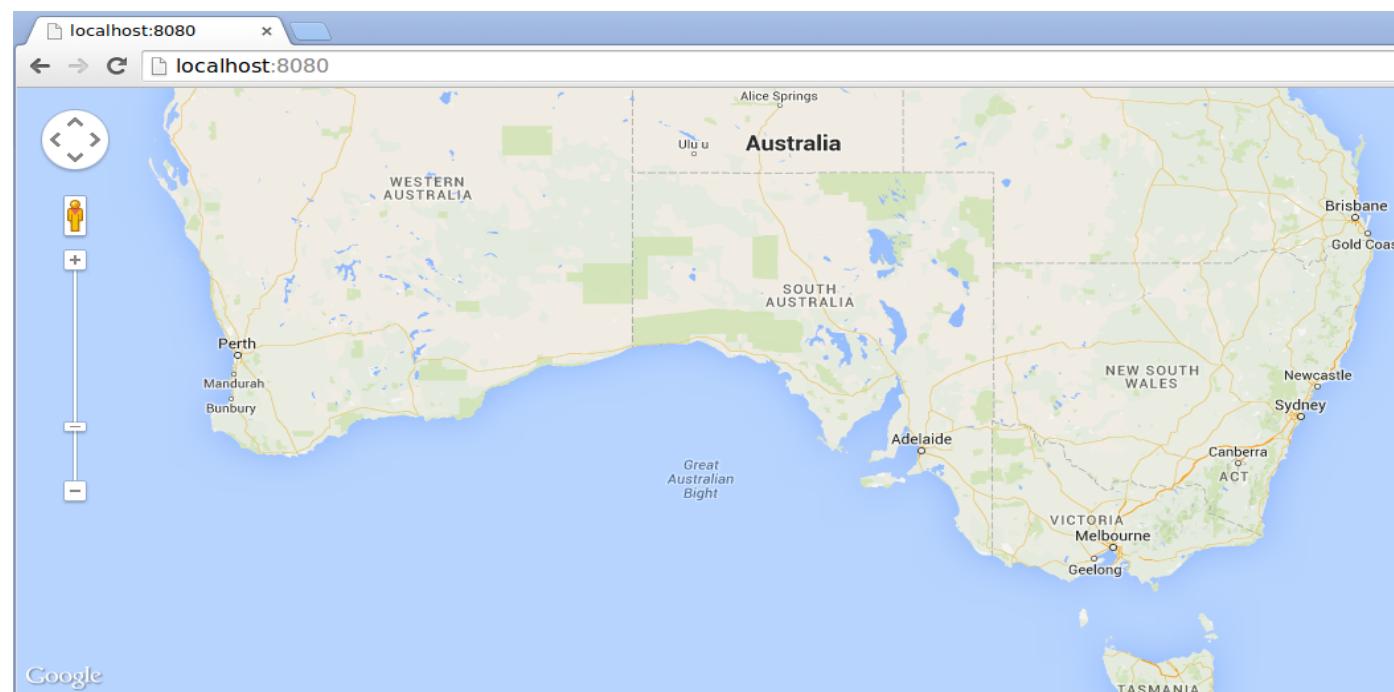
dump1090 --interactive --aggressive --net

Open browser on **http://localhost:8080** for map

3. Planes - dump1090: Viewing on a Map

Hex	Flight	Altitude	Speed	Lat	Lon	Track	Messages Seen	.

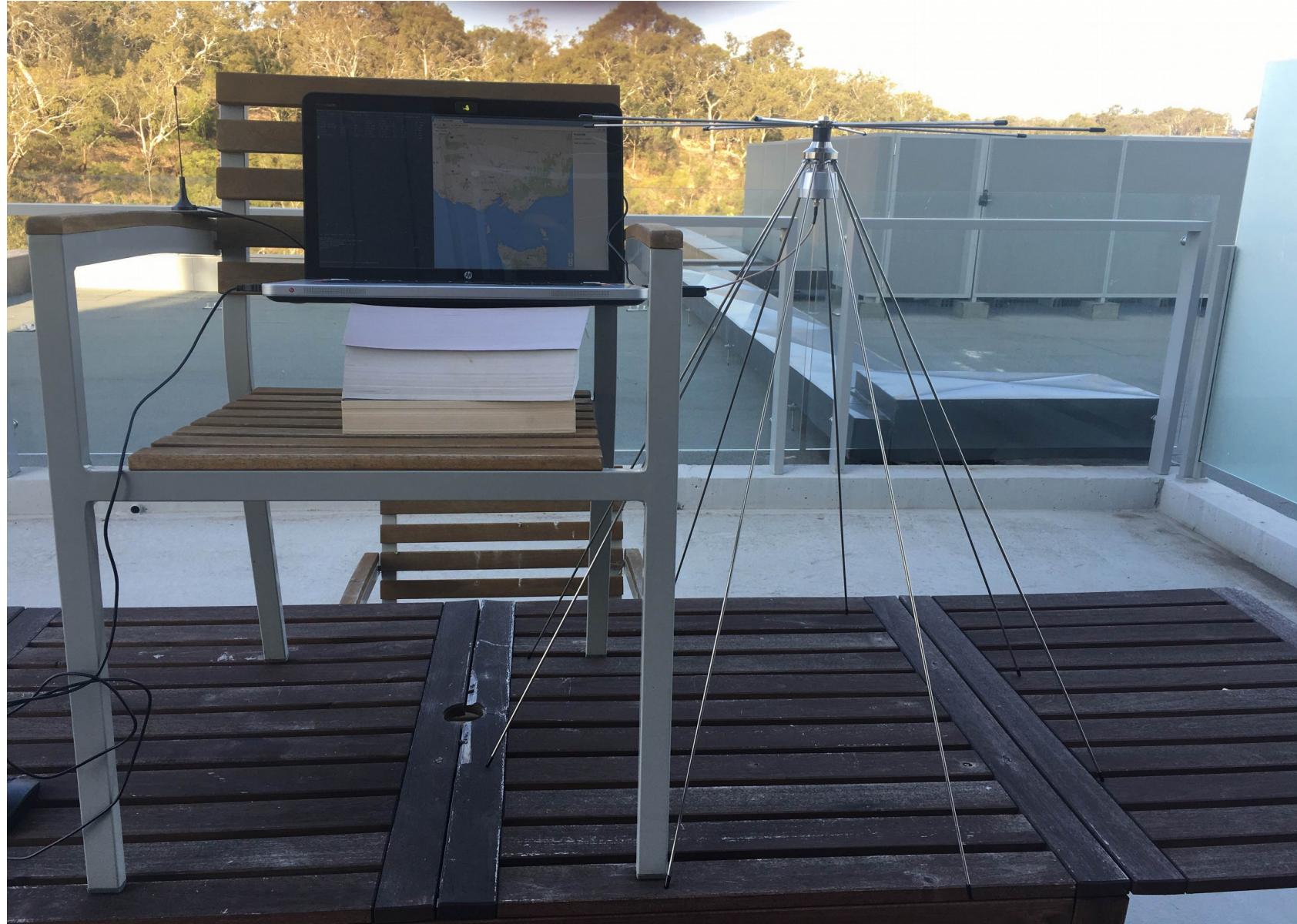
c319ce	0	0	0.000	0.000	0	1	32	sec
0c1a1f	0	0	0.000	0.000	0	1	37	sec
bdf973	0	0	0.000	0.000	0	1	41	sec
cb09bf	0	0	0.000	0.000	0	1	52	sec



3. Planes – dump1090: Viewing messages

```
*8ce3b19f2f463e3c1d81330e61f8;  
CRC: 0e61f8 (ok)  
Single bit error fixed, bit 27953  
DF 17: ADS-B message.  
    Capability : 4 (Level 2+3+4 (DF0,4,5,11,20,21,24,code7 - is on ground))  
    ICAO Address : e3b19f  
    Extended Squitter Type: 5  
    Extended Squitter Sub : 7  
    Extended Squitter Name: Surface Position  
    Unrecognized ME type: 5 subtype: 7  
  
*888a7fbe2e7e50c939f662bc85ce;  
CRC: bc85ce (ok)  
Single bit error fixed, bit 27999  
DF 17: ADS-B message.  
    Capability : 0 (Level 1 (Surveillance Only))  
    ICAO Address : 8a7fbe  
    Extended Squitter Type: 5  
    Extended Squitter Sub : 6  
    Extended Squitter Name: Surface Position  
    Unrecognized ME type: 5 subtype: 6  
  
*8d2b1817222070fcbd253d4f78d;  
CRC: d4f78d (ok)  
Single bit error fixed, bit 18760  
DF 17: ADS-B message.  
    Capability : 5 (Level 2+3+4 (DF0,4,5,11,20,21,24,code7 - is on airborne))  
    ICAO Address : 2b1817  
    Extended Squitter Type: 4  
    Extended Squitter Sub : 2  
    Extended Squitter Name: Aircraft Identification and Category  
        Aircraft Type : Aircraft Type A  
        Identification : HGC??ZIS  
  
*890a3ed27f7750e80dc3620f309b;  
CRC: 0f309b (ok)  
Single bit error fixed, bit 27486  
DF 17: ADS-B message.  
    Capability : 1 (Level 2 (DF0,4,5,11))  
    ICAO Address : 0a3ed2  
    Extended Squitter Type: 15  
    Extended Squitter Sub : 7  
    Extended Squitter Name: Airborne Position (Baro Altitude)  
        F flag : even  
        T flag : non-UTC  
        Altitude : 22725 feet  
        Latitude : 29702 (not decoded)  
        Longitude: 115554 (not decoded)
```

3. Planes – Discone Antenna



3. Planes – Discone Antenna



3. Planes – Discone Antenna



3. Planes – Discone Antenna



3. Planes

[Video]

3. Planes

- The basic rubber ducky antenna with the RTL SDR will still give you great results
- Just make sure its in line of sight of the sky
- The higher the better
- Could even try with a drone and present your results here!

4. Ships



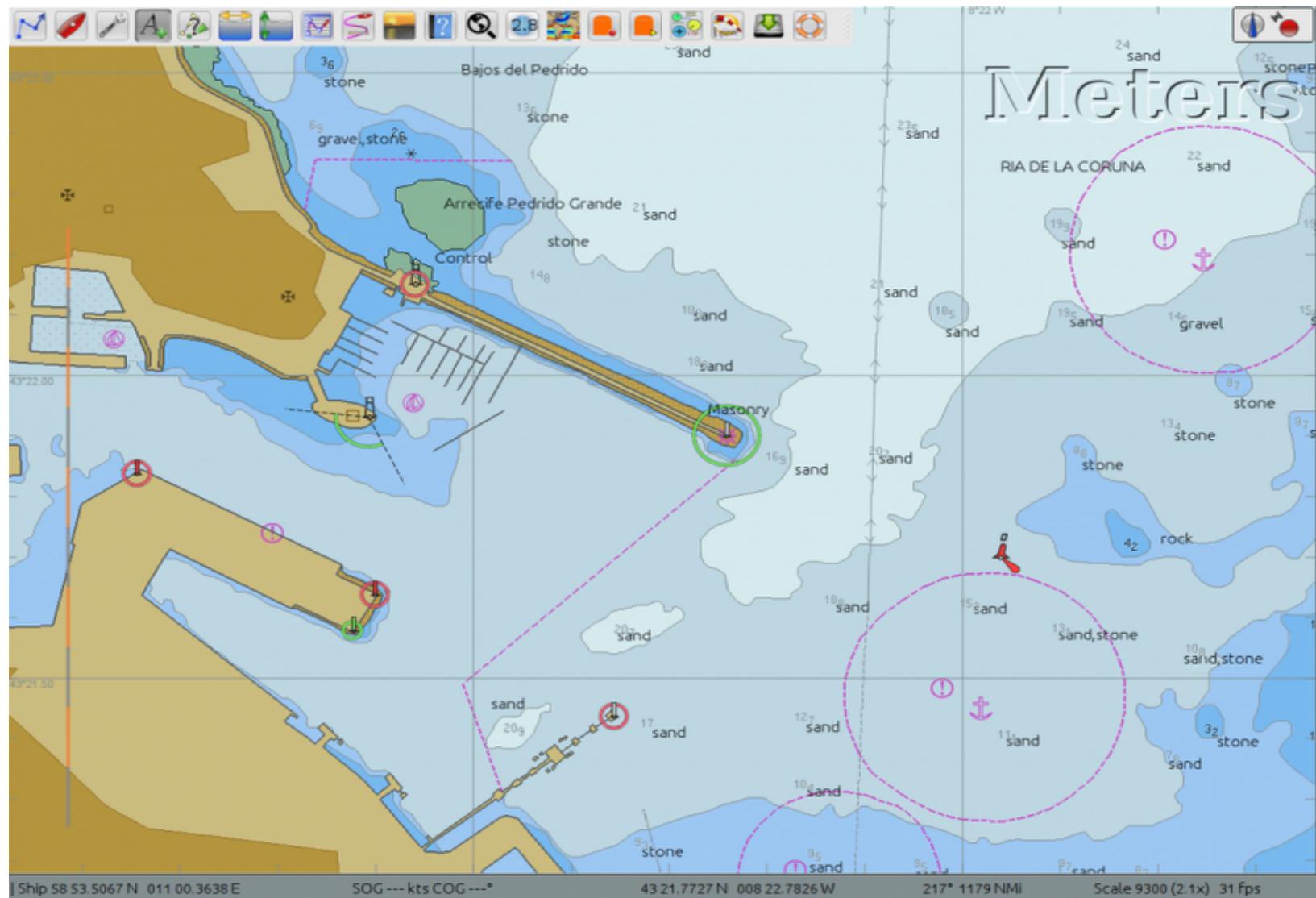
4. Ships - AIS

- AIS: Automatic Identification System
- Tracking systems for ships
- Location
- Messages
- Similar to ADS-B
- 162Mhz

4. Ships

- AIS receiver: **ais_rx**
 - <https://github.com/bistromath/gr-ais/>
 - `./ais_rx -s osmocom`
- Chart plotting tool: **opencpn**
 - <http://opencpn.org/ocpn/>

4. Ships - openCPN



5. Pagers



5. Pagers - POCSAG

- POCSAG: Post Office Code Standardisation Advisory Group
- Other pager protocols include FLEX
- Australia uses:
 - 148.3375 MHz (VHF)
 - 450.375 MHz (UHF)
 - 450.325 MHz (UHF)

5. Pagers – multimon-ng

- Tool: **multimon**

<https://github.com/EliasOenal/multimon-ng>

1. **gqrx**: tune to a pager frequency

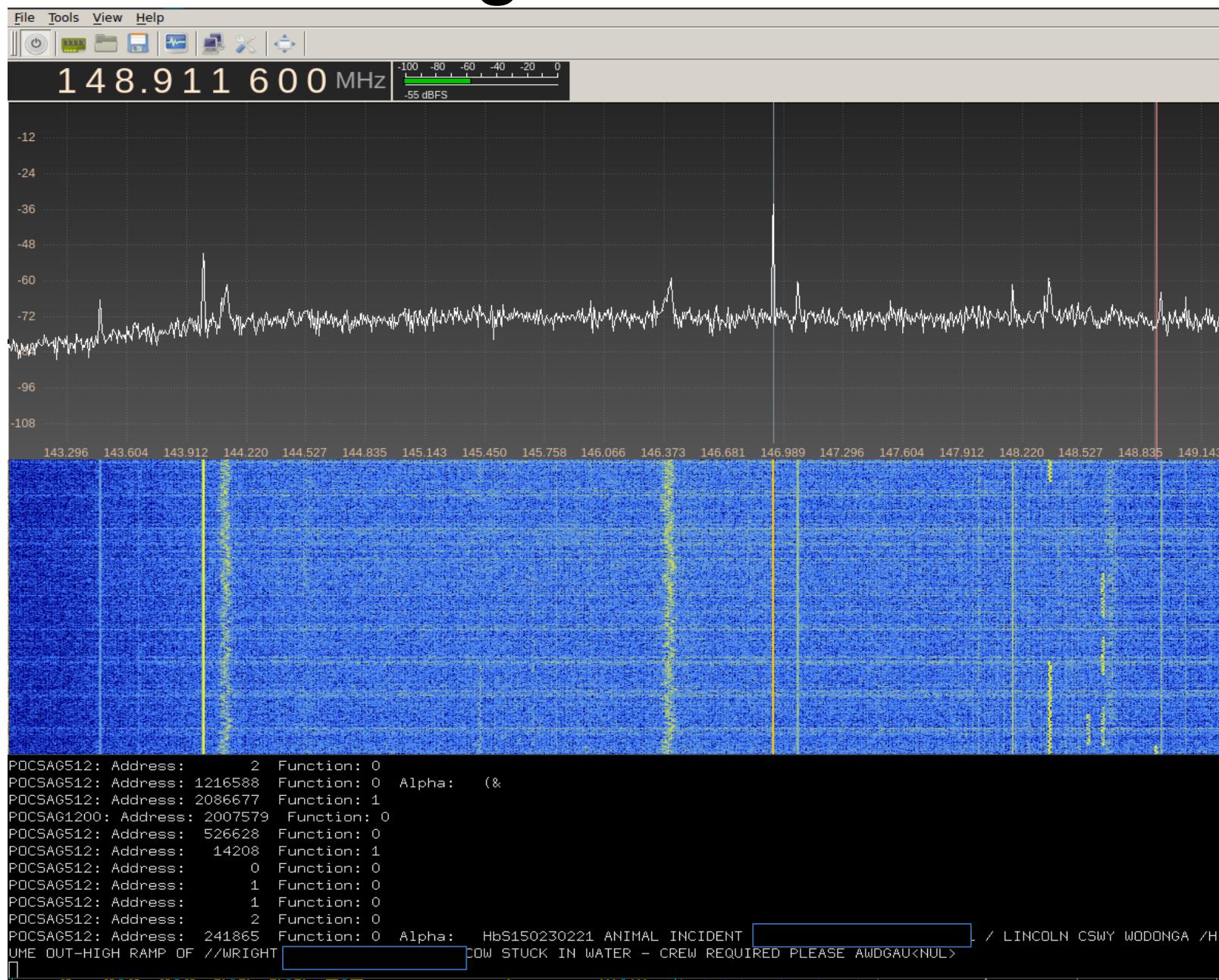
Filter: Wide

Mode: Narrow FM

2. **padsp multimon-ng** -a POCSAG512 -a POCSAG1200 -a
POCSAG2400 -f alpha

3. **pauvcontrol** – enable recording from internal sound card

5. Pagers – VET

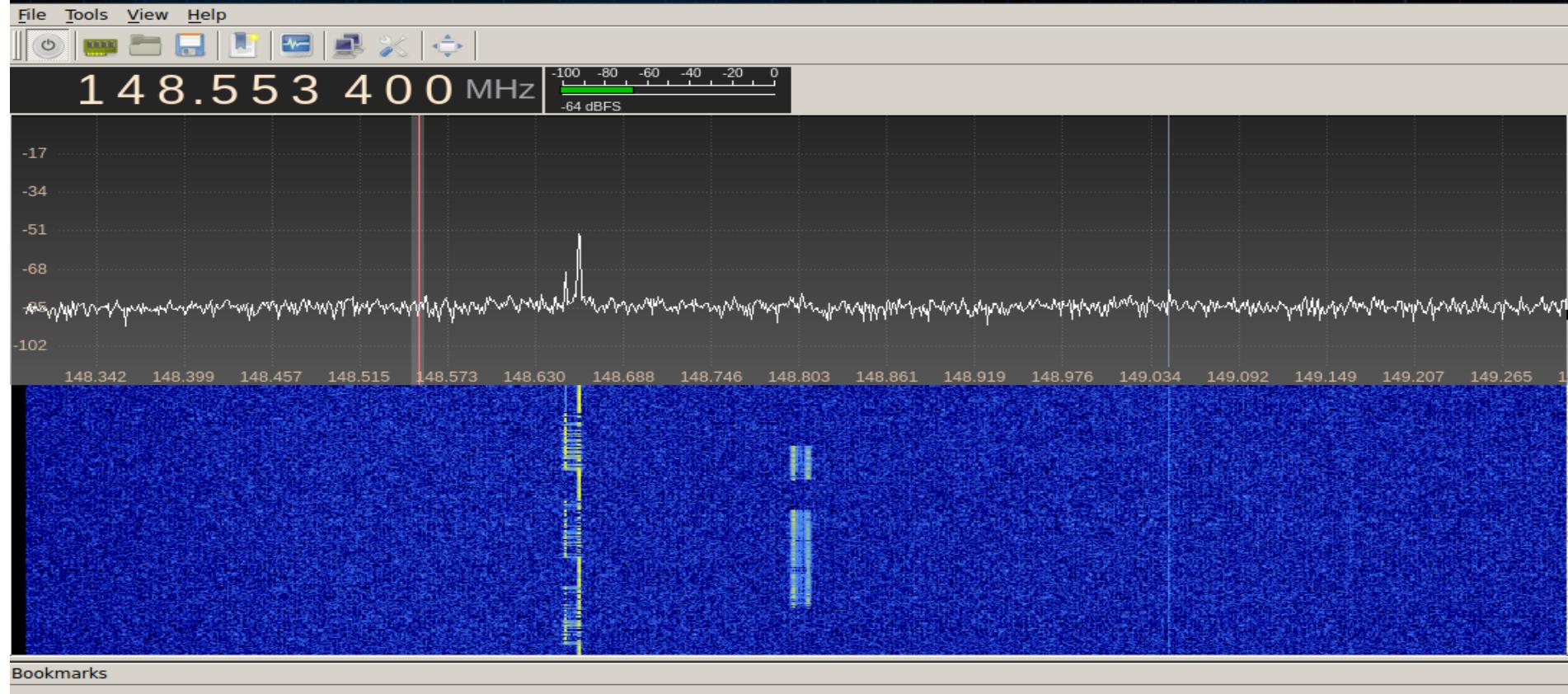


5. Page – VET (zoom-in)

```
2 Function: 0  
216588 Function: 0 Alpha: (&  
086677 Function: 1  
2007579 Function: 0  
326628 Function: 0  
14208 Function: 1  
0 Function: 0  
1 Function: 0  
1 Function: 0  
2 Function: 0  
41865 Function: 0 Alpha: HbS150230221 ANIMAL INCIDENT ROB  
//WRIGHT G2 COW STUCK IN WATER - CREW REQUIRED PLEASE AWDGAU<NUL> / LI
```

5. Pagers – Doors

```
POCSAG512: Address: 1084098 Function: 1
POCSAG512: Address: 862372 Function: 2 Alpha: MLK P1-61 East Foyer Auto Door Push Button has changed state to: Open-Circuit Ta<NUL><NUL>
POCSAG512: Address: 15808 Function: 0
POCSAG512: Address: 0 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 2 Function: 0
POCSAG512: Address: 2 Function: 0
POCSAG512: Address: 862372 Function: 2 Alpha: MLK P1-61 East Foyer Auto Door Push Button has changed state to: Open-Circuit Ta<NUL><NUL>
POCSAG512: Address: 15808 Function: 0
POCSAG512: Address: 0 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 2 Function: 0
POCSAG512: Address: 862372 Function: 2 Alpha: MLK P1-61 East Foyer Auto Door Push Button has changed state to: Open-Circuit Ta<NUL><NUL>
POCSAG512: Address: 15808 Function: 0
POCSAG512: Address: 0 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 1 Function: 0
POCSAG512: Address: 2 Function: 0
POCSAG512: Address: 0
```



5. Page – Doors (zoom-in)

5. Page – Licence Plate Checks

7BDGS H T3 14:48 27/05 SAINT KENTIGERN TRUST 111111 Matthew		24C 78 075	KURANGA ROAD AUCKLAND * DPL CHECK OK
7BDGS H T3 14:48 27/05 SAINT KENTIGERN TRUST 111111 Matthew		24C 78 075	KURANGA ROAD AUCKLAND * DPL CHECK OK
7BDGS H T3 14:48 27/05 SAINT KENTIGERN TRUST 111111 Matthew		24C 78 075	KURANGA ROAD AUCKLAND * DPL CHECK OK
20:21 P260001.ADA			
20:22 P260001.ADA			
20:23 P260001.ADA			
20:24 P260001.ADA			
7BDXX H T3 14:41 27/05 KELVIN ROAD SCHOOL 111111 Jeffrey B 0212		24C 7 8 34259	ROAD PAPAKURA * DPL CHECK OK
7BDXX H T3 14:41 27/05 KELVIN ROAD SCHOOL 111111 Jeffrey B 0212		24C 7 8 34259	ROAD PAPAKURA * DPL CHECK OK
7BDXX H T3 14:41 27/05 KELVIN ROAD SCHOOL 111111 Jeffrey B 0212		24C 7 8 34259	ROAD PAPAKURA * DPL CHECK OK
7BDXX H T3 14:41 27/05 KELVIN ROAD SCHOOL 111111 Jeffrey B 0212		24C 7 8 34259	ROAD PAPAKURA * DPL CHECK OK
MOUNT_BARKER:11103503,SCOTT AND			

5. Pagers – Licence Plate Checks (zoom in)

111111 Matthew Way 006

111111 Matthew Way 006

111111 Matthew Way 006

2 24C 78 07545

2 24C 78 07545

2 24C 78 07545

PAK

PAK

PAK

1 Jeffrey B 0212177208 2072 24C 7 8 34259 200

1 Jeffrey B 0212177208 2072 24C 7 8 34259 200

1 Jeffrey B 0212177208 2072 24C 7 8 34259 200

1 Jeffrey B 0212177208 2072 24C 7 8 34259 200

11103503,SCOTT AND

KELVIN ROAD PAPAKUR

KELVIN ROAD PAPAKUR

KELVIN ROAD PAPAKUR

KELVIN ROAD PAPAKUR

<NUL><NUL>

5. Pagers – Passcodes

500498 Function: 2 Alpha: key safe located on the Right hand side of the front door - code is 1926<NULL>

“key safe located on the Right hand side of the front door – code is 1926”

5. Pagers – Datacentre Servers

```
98 Saturday, 30 May 2015 4:39 PM GigabitEthernet1/0/1 WIRELESS AP on ADLSW01.win.  
96 Saturday, 30 May 2015 4:39 PM GigabitEthernet1/0/1 WIRELESS AP on ADLSW01.win.  
97 Saturday, 30 May 2015 4:39 PM GigabitEthernet1/0/1 WIRELESS AP on ADLSW01.win.  
Saturday, 30 May 2015 4:39 PM GigabitEthernet1/0/1 WIRELESS AP on ADLSW01.win.int  
Saturday, 30 May 2015 4:39 PM GigabitEthernet1/0/1 WIRELESS AP on ADLSW01.win.int
```

```
com.au is Down<NULL>  
com.au is Down<NULL>  
com.au is Down<NULL>  
.au is Down<NULL>  
.au is Down<NULL>
```

```
Ä1/2Ü ÄRepeat #2Ü EM Event: Critical:NBMSP.ar [REDACTED] org.au_NBMSPL-cluster_NBMSPL_3 - Out of memory detected in /apps/oracle/diag/rdbms/nbmsp/NBMSPL<NULL>  
17:59 X 18365 30/05/15 17:59:28  
18:01 P260001.ADA
```

```
84 Saturday, 30 May 2015 5:02 PM<br/>REA-EQX-SQLN1 100 % CPU - Top 10<br/><br/>http://REA-EQX-ORION01:80/Orion/View.aspx?NetObject=N:301<NULL>
```

5. Pagers – Datacentre Servers (zoom in)

```
$ AP on ADLSW01.win.i  
$ AP on ADLSW01.win.i  
$ AP on ADLSW01.win.i  
? on ADLSW01.win.int.  
? on ADLSW01.win.int.
```

```
.com.au is Down  
.com.au is Down  
.com.au is Down  
.com.au is Down  
.au is Down<NU  
.au is Down<NU
```

```
* memory detected in /apps/oracle/diag/rdbms/nbmsp/NBMSPL
```

```
http://REA-EQX-ORION01:80/Orion/View.aspx?NetO
```

5. Pagers – Emergency Services

POCSAG1200: Address: 452370 Function: 2 Alpha:	[REDACTED]	GARDENS,MFS,shed fire, please isolate ,<NUL><NUL>
POCSAG1200: Address: 452368 Function: 2 Alpha:	[REDACTED]	GARDENS,MFS,shed fire, please isolate ,<NUL><NUL>
POCSAG1200: Address: 452369 Function: 2 Alpha:	[REDACTED]	GARDENS,MFS,shed fire, please isolate ,<NUL><NUL>
POCSAG1200: Address: 370377 Function: 2 Alpha:	21:29 X 18379 30/05/15 21:29:28	
POCSAG1200: Address: 370377 Function: 2 Alpha:	21:29 P260001.ADA	
POCSAG1200: Address: 370377 Function: 2 Alpha:	21:30 P260001.ADA	
POCSAG1200: Address: 440500 Function: 2 Alpha:	7B51H H M3 : / AUSGRID DC NORTH RYDE 822000 LENIN +91990 7.3.0.11 GIVING ERROR<NUL><NUL>	AAXR0 23 -25
POCSAG1200: Address: 440501 Function: 2 Alpha:	7B51H H M3 : / AUSGRID DC NORTH RYDE 822000 LENIN +91990 7.3.0.11 GIVING ERROR<NUL><NUL>	AAXR0 23 -25
POCSAG1200: Address: 408384 Function: 2 Alpha:	96 Pole Rqd Lendlease Plse call NOC - re PAW chp job . Refer Ian - cheers.	
POCSAG1200: Address: 405195 Function: 2 Alpha:	Pole Rqd Lendlease Plse call NOC - re PAW chp job . Refer Ian - cheers.	

5. Pagers – Emergency Services (zoom in)

39244,S21,8 [REDACTED] MFS,shed fire, please isolate
39244,S21,8 [REDACTED] MFS,shed fire, please isolate
39244,S21,8 [REDACTED] MFS,shed fire, please isolate
;29 X 18379 30/05/15 21:29:28
;29 P260001.ADA
;30 P260001.ADA
51H H M3 : /
51H H M3 : /

Pole Rqd Lendlease Plse call NOC - re PAW chp job . Refer Ian - cheer
le Rqd Lendlease Plse call NOC - re PAW chp job . Refer Ian - cheers.

6. Home Devices

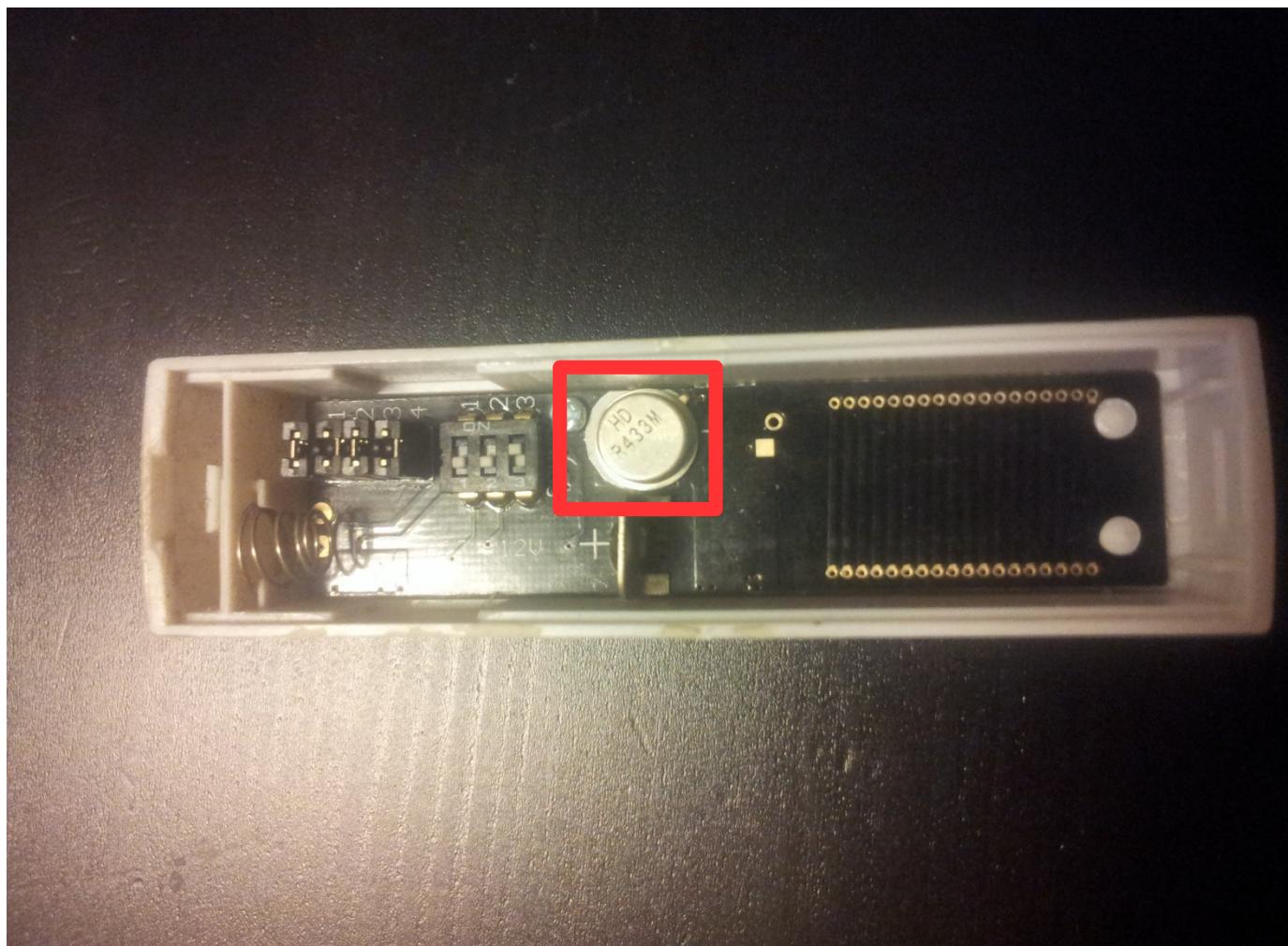


6. Home Devices

- **Doorbells**
- Garage doors
- Baby monitors
- Home automation
- Smart Meters

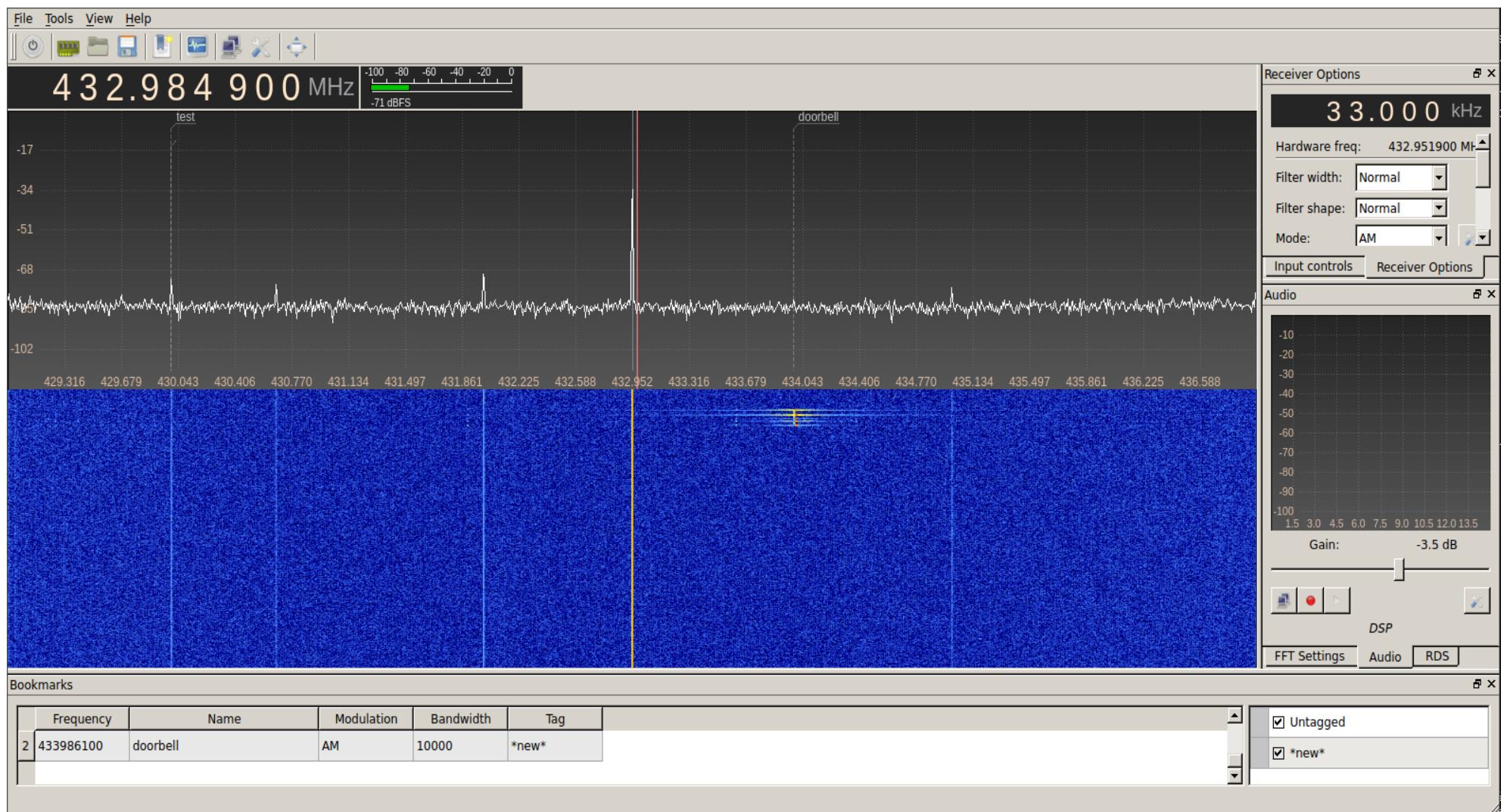
6. Home Devices - Doorbells

1) Identify Frequency: 433Mhz (approx)



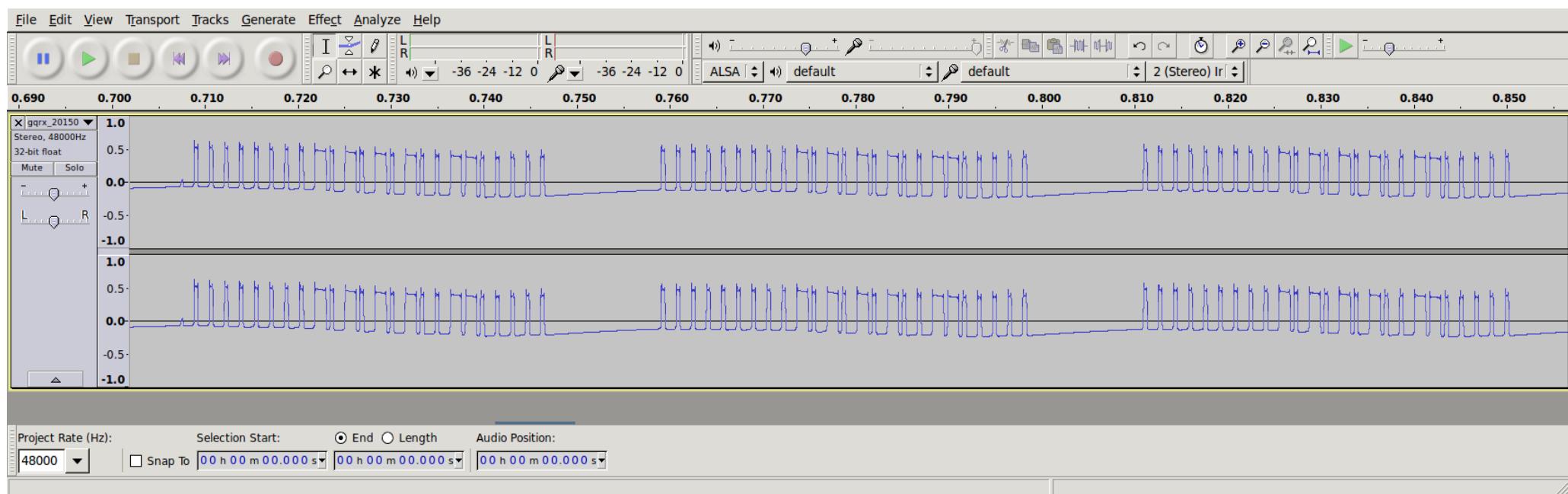
6. Home Devices - Doorbells

2A) Identify Modulation: listening in gqrx



6. Home Devices - Doorbells

2B) Open Recoding in Audacity



6. Home Devices - Doorbells

2B) Open Recoding in Audacity...

- We can clearly see ON/OFF (0 = OFF, 1 = ON): **Amplitude Modulation**
- Shorter pulses are 1, Longer pulses are consecutive ones
- OOK – On Off Shifting Keying

6. Home Devices - Doorbells

3) Capture Raw Data

Check frequency in gqrx and record with a
hackrf:

hackrf_transfer -r 433995700.raw -f 433995700

6. Home Devices - Doorbells

4) Replay without remote

Shift the frequency for transmission down 100 KHz to avoid the carrier spike in middle of our signal

```
hackrf_transfer -t 433985700.raw -f 433985700 -x 20
```

Questions

