Reverse engineering & hijacking toy quadcopters

All your drones are belong to me…

BSides Vancouver 2017

Agenda

- $whoami
- Introduction
- OSINT
- The hard way
- Over the air
- Conclusion

$whoami

- Yannick Formaggio (@TheLumberjHack)
  - IT Security researcher @ Istuary Innovation Labs (Downtown Vancouver)
  - Originally software vulnerability hunter (presented VxWorks RCE back in 2015)
  - RF/Hardware tickles my curiosity (always learning 😌)

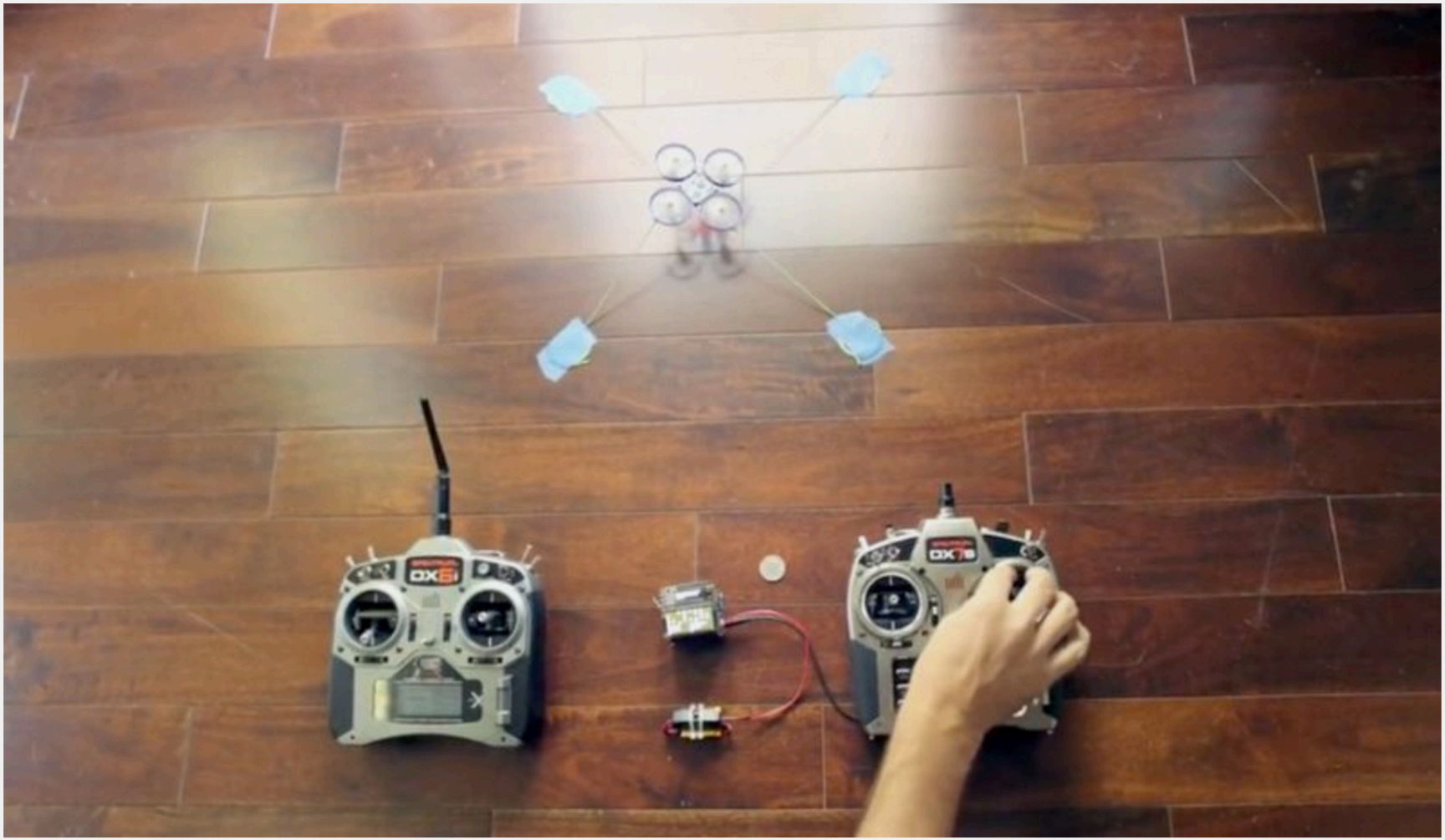
ISTUARY
INNOVATION GROUP

Introduction

There’s a new way to take down drones, and it doesn’t involve shotguns

Not a jammer, device lets hackers fly drones and lock out original pilot.

DAN GOODIN - 10/26/2016, 5:47 PM



Jonathan Andersson

You purchased this item on Nov 17 2016.
View this order

Roll over image to zoom in

Cheerson CX-10WD-TX Real Time WiFi FPV **New Version: Altitude Hold and Transmitter** Mini Quadcopter Android / iOS RC 4CH 2.4GHz 6 Axis Nano Drone HD Video Camera (Golden)

by iNovaDirect

3 customer reviews

Price: CDN$ 63.99 **FREE Shipping** (4 days) for Prime members Details

In Stock.

Want it Tuesday, March 14? Order it in the next **3 hours and 34 minutes** and choose **Standard Shipping** at checkout.

Sold by iNovaDirect and Fulfilled by Amazon. Gift-wrap available.

- The World's smallest High-Tech FPV Quadcopter Drone
- Upgraded Version of CX-10W **New features: Altitude Hold and Transmitter** Real-Time Video Transmission on your smartphone.
- One touch Take-off / One touch Landing / One touch Return - Altitude holds automatically - Very Stable, Easy to Maneuver: Up/Down, Left/Right, Forward/Backward, Left/Right Sideward Flying, 3D 360 Eversion - You can throw it and fly - Night flying with colorful LEDs - Miniature size makes it perfect for small spaces.
- Download Software "CX-10wifi" by scanning QR code or directly in APP store / Google Play for Wifi Camera transmission - Transmitter and/or Mobile Phone Dual Operating Mode - Record your flights and take pictures directly on your smartphone.
- 4 Spare Blades included + Extra: Propeller Guard

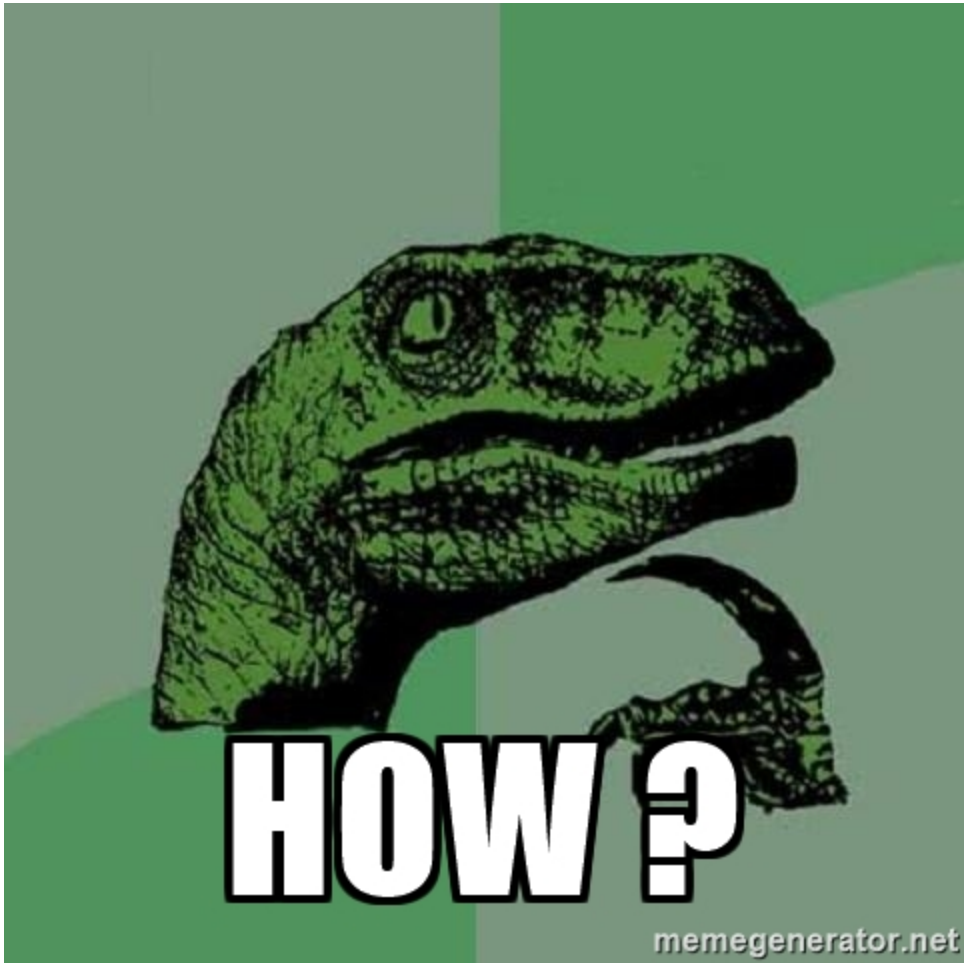
My ultimate goal

- Detecting the flying drone using RF
- Take over the control
- Bring it down/Push it away

Let’s reverse it !

What do we need?

- **Understand TX Communication protocol:**
  - Determine modulation
  - Determine data/symbol rate
  - Determine frequencies of operation
  - Frequency hopping behavior if any
  - Determine packet format (size, header, payload, CRC…)
- How to glue things together to make it happen


HOW ?
memegenerator.net

OSINT

Finding info without opening the box…

Searchable FCC ID Database

The information resource for all wireless device applications filed with the FCC.

Check Today's FCC ID Filings

FCC ID Search:

FCC ID: XXX-XXXXXXX Search

What is an FCC ID?

An FCC ID is a unique identifier assigned to a device registered with the United States Federal Communications Commission. For legal sale of wireless deices in the US, manufacturers must:

- Have the device evaluated by an independent lab to ensure it conforms to FCC standards
- Provide documentation to the FCC of the lab results
- Provide User Manuals, Documentation, and Photos relating to the device
- Digitally or physically label the device with the unique identifier provided by the FCC (upon approved application)

FCC IDs are required for all wireless emitting devices sold in the USA. By searching an FCC ID, you can find details on the wireless operating frequency (including strength), photos of the device, user manuals for the device, and SAR reports on the wireless emissions.

deviation

Main Menu

- Home
- Forum
- Wiki
  - Getting Started
  - FAQ
  - Supported Models
- User Manual
  - User Manual for DEVO 6/8/12
  - User Manual for DEVO 7E/10
- Downloads
  - Release Notes
- Bug Tracker

Index | Recent Topics | Search | Log in

Forum › Development › Protocol Development › JD 395 cx-10

JD 395 cx-10

Start | Prev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | ... | 22 | Next | End

Search

kamueone

TOPIC AUTHOR

Offline

More

17 Jul 2014 14:39 #24636

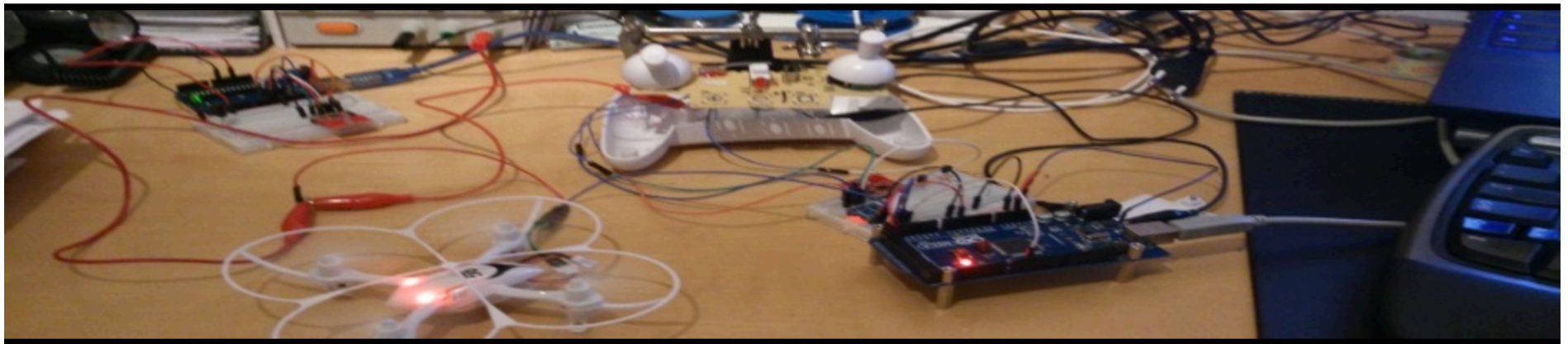
kamueone created the topic: JD 395 cx-10

The new green Board on the Cheerson cx-10 uses the same protocol as the JD 395.

Is there a way to use that protocol with a devo 7e?

Michael Melchior

Ein schoener Tag!



Home About me Disclaimer

← Kids Music Box

Reverse Engineering a Quadcopter RC, or: How to not miss the needle while throwing the haystack in the air (Part 2) →

Reverse Engineering a Quadcopter RC, or: How to not miss the needle while throwing the haystack in the air (Part 1)

Posted on June 6, 2016

Search

Recent Posts

- Tu(r)ning a 40MHz RC Ferrari (from digital into a 2.4GHz analog one) (Part 1)
- Reverse Engineering a Quadcopter

This repository Search Pull requests Issues Gist

m-melchior / QC-360-A1

Watch 4 Unstar 30

Code Issues 1 Pull requests 0 Projects 0 Wiki Pulse Graphs

Reverse engineering the QC 360 A1 quadcopter RC protocol https://mmelchior.wordpress.com/2016/...

Hacking the CampCopter

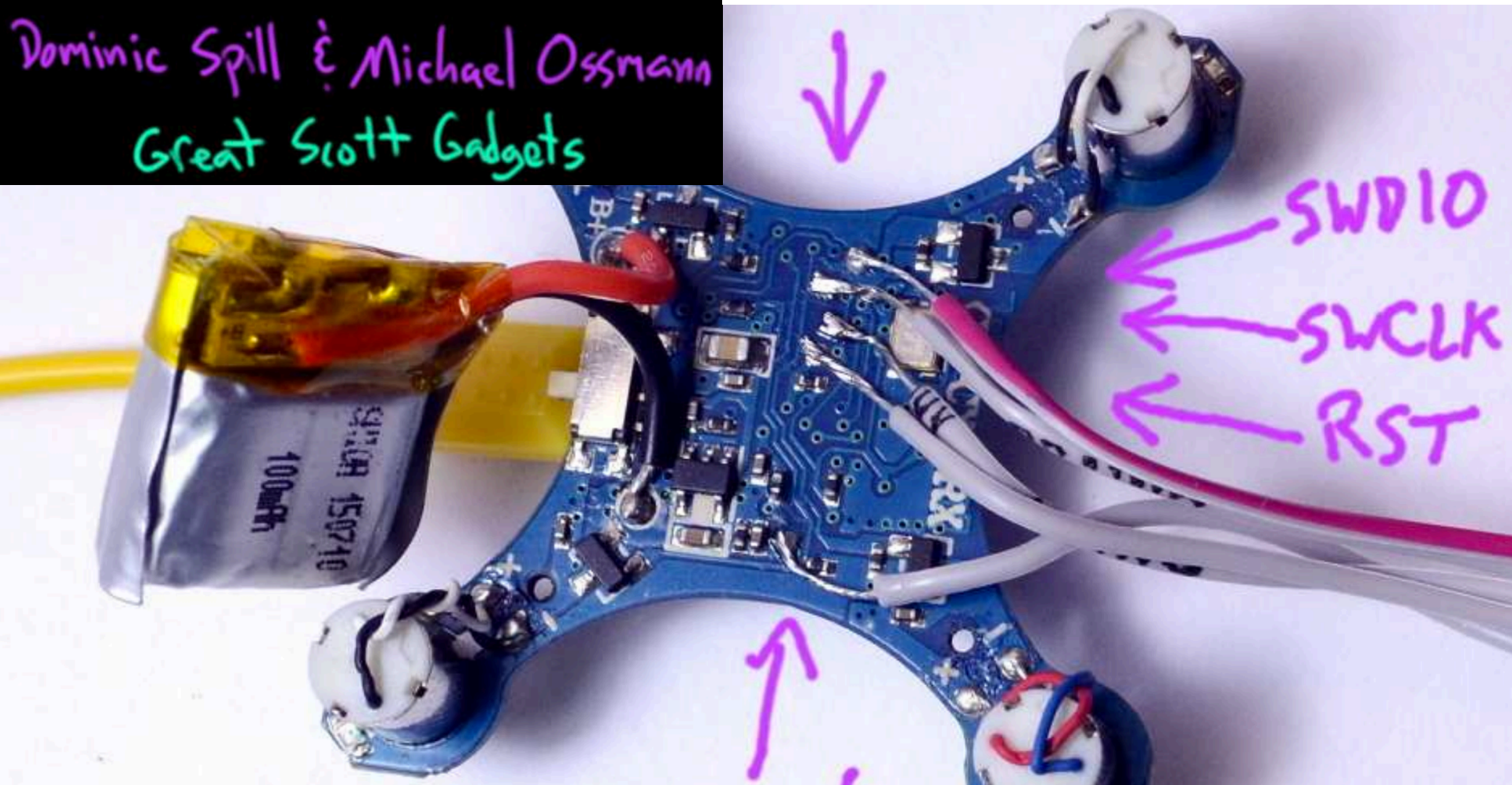
Dominic Spill & Michael Ossmann
Great Scott Gadgets

This repository Search Pull requests

marcnewlin / drone-duel

Code Issues 1 Pull requests 0 Projects 0

Code used in the Great Drone Duel of 2016



XN297L 系列产品说明书

2.4GHz 单片高速无线收发芯片

概 述

XN297L系列芯片是工作在2.400~2.483GHz世界通用ISM频段的单片无线收发芯片。该芯片集成射频收发机、频率发生器、晶体振荡器、调制解调器等功能模块，并且支持一对多组网和带ACK的通信模式。发射输出功率、工作频道以及通信数据率均可配置。芯片已将多颗外围贴片阻容感器件集成到芯片内部。

主 要 特 性

1、功耗较低

发射模式（2dBm）工作电流19mA；接收模式工作电流15mA；休眠电流2uA。

2、节省外围器件

支持外围5个元器件，包括1颗晶振和4个贴片电容；

支持双层或单层印制板设计，可以使用印制板微带天线；

芯片自带部分链路层的通信协议；配置少量的参数寄存器，使用方便。

3、性能优异

250K / 1M / 2M bps模式的接收灵敏度为-91 / -87 / -83dBm；发射输出功率最大可达13dBm；抗干扰性好，接收滤波器的邻道抑制度高，接收机选择性较好。

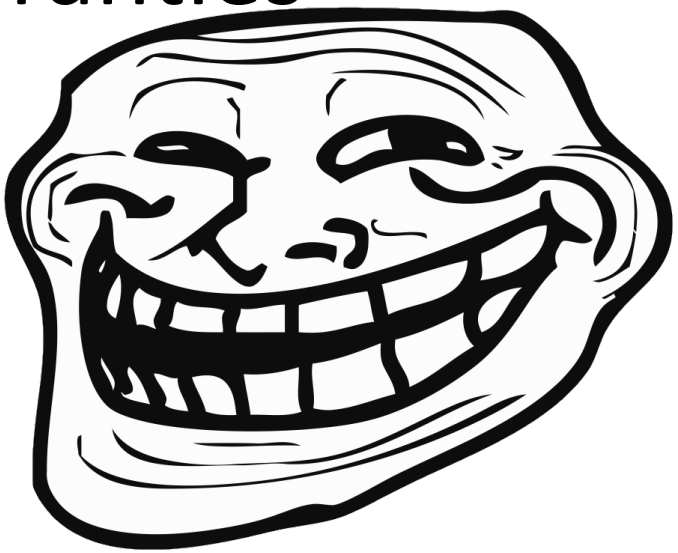
What did we learn?

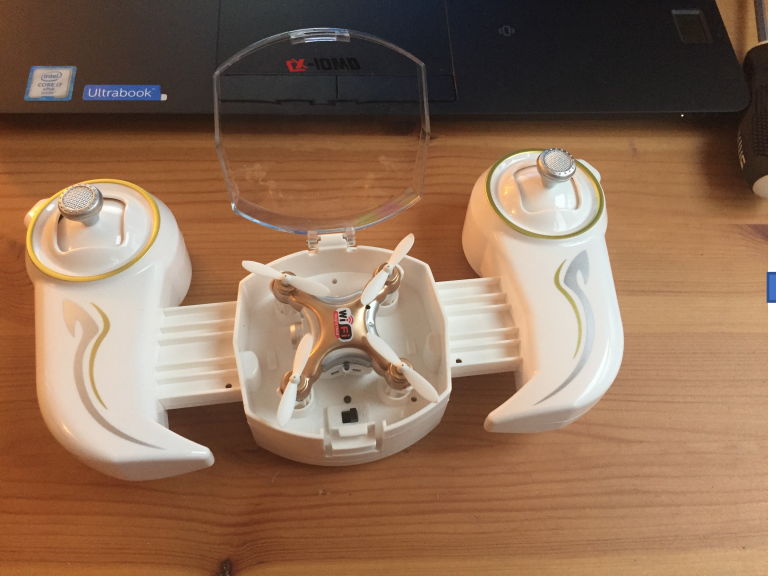
其它特性

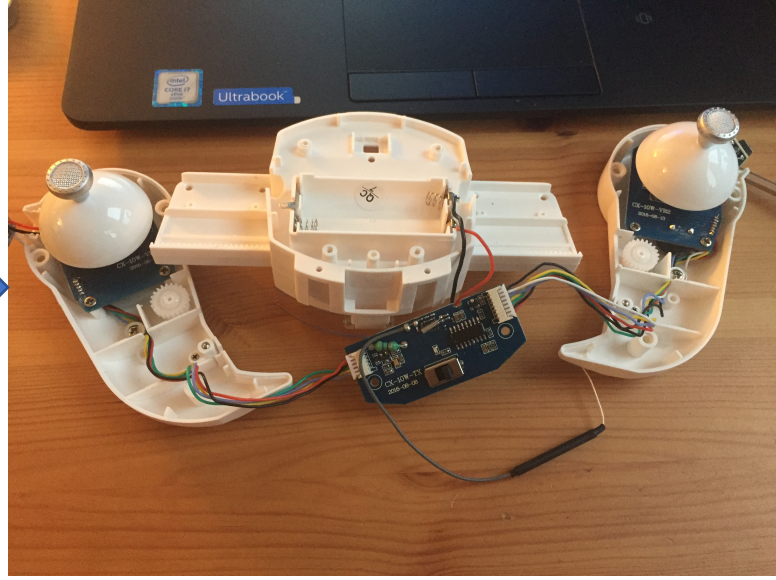
四线 SPI 接口通信	SPI 接口速率最高支持4Mbps
支持最大数据长度为32字节（两级FIFO） 或者 64字节（单级FIFO）	QFN20L0303 / SOP16 / SOP8封装
1M / 2Mbps模式，需要晶振精度 ±40ppm 250kbps模式，需要晶振精度 ±20ppm	工作电压支持2.2~3.3V 工作温度支持-40~+85℃
GFSK通信方式	支持自动应答及自动重传
支持RSSI检测功能	带自动扰码和CRC校验功能

The hard way

Let’s void some warranties

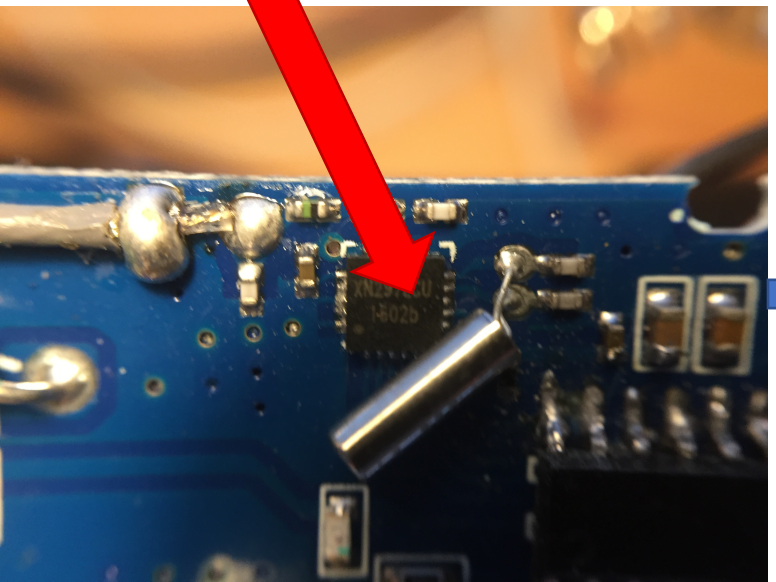


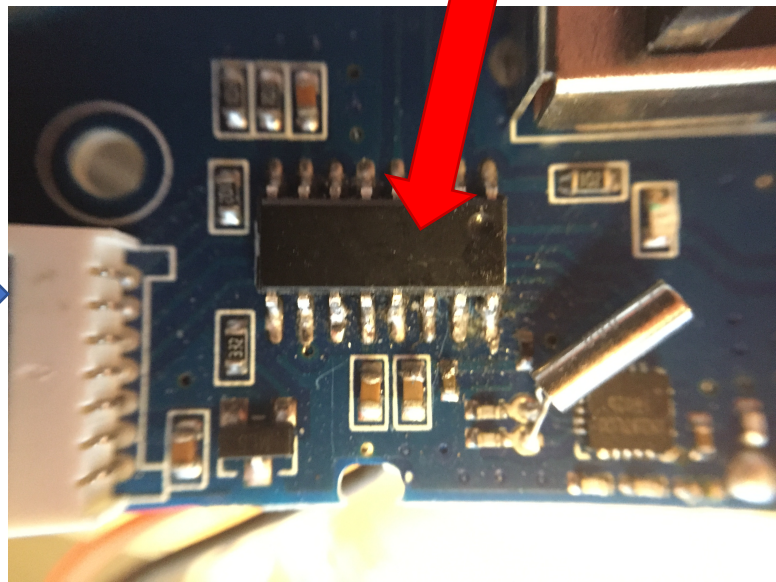




XN297LCU

MCU





XN297L 系列产品说明书

2.4GHz 单片高速无线收发芯片

概 述

XN297L系列芯片是工作在2.400~2.483GHz世界通用ISM频段的单片无线收发芯片。该芯片集成射频收发机、频率发生器、晶体振荡器、调制解调器等功能模块，并且支持一对多组网和带ACK的通信模式。发射输出功率、工作频道以及通信数据率均可配置。芯片已将多颗外围贴片阻容感器件集成到芯片内部。

主 要 特 性

1、功耗较低

发射模式（2dBm）工作电流19mA；接收模式工作电流15mA；休眠电流2uA。

2、节省外围器件

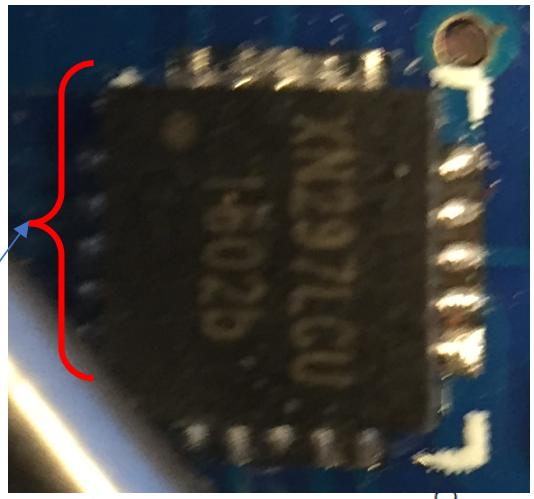
支持外围5个元器件，包括1颗晶振和4个贴片电容；

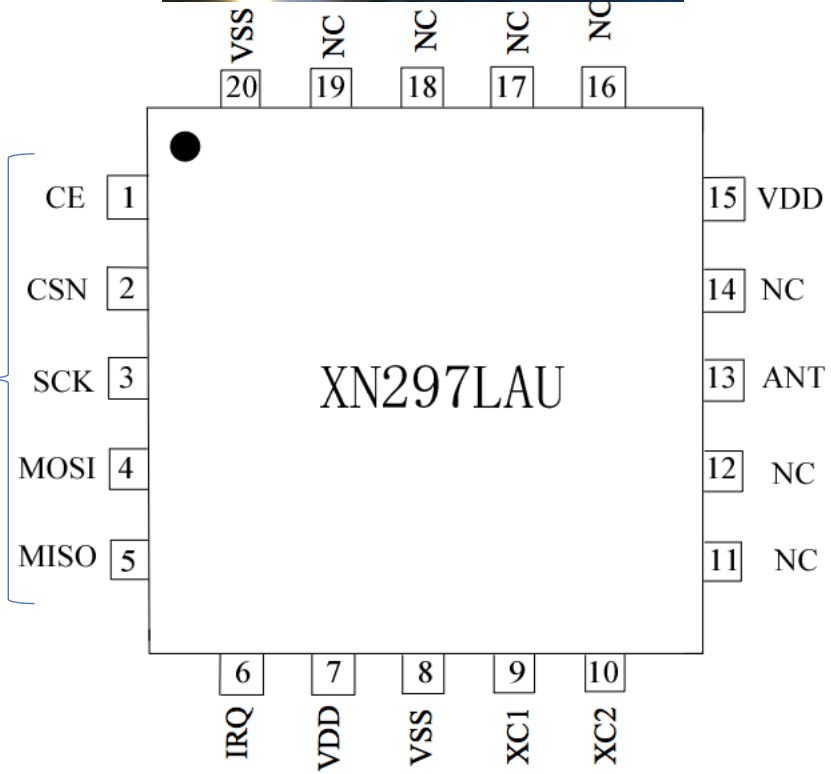
支持双层或单层印制板设计，可以使用印制板微带天线；

芯片自带部分链路层的通信协议；配置少量的参数寄存器，使用方便。

3、性能优异

250K / 1M / 2M bps模式的接收灵敏度为-91 / -87 / -83dBm；发射输出功率最大可达13dBm；抗干扰性好，接收滤波器的邻道抑制度高，接收机选择性较好。

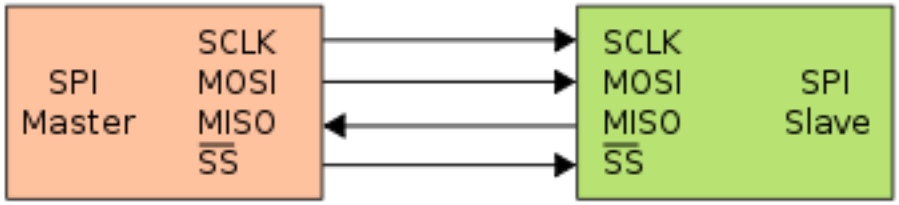




SPI pins

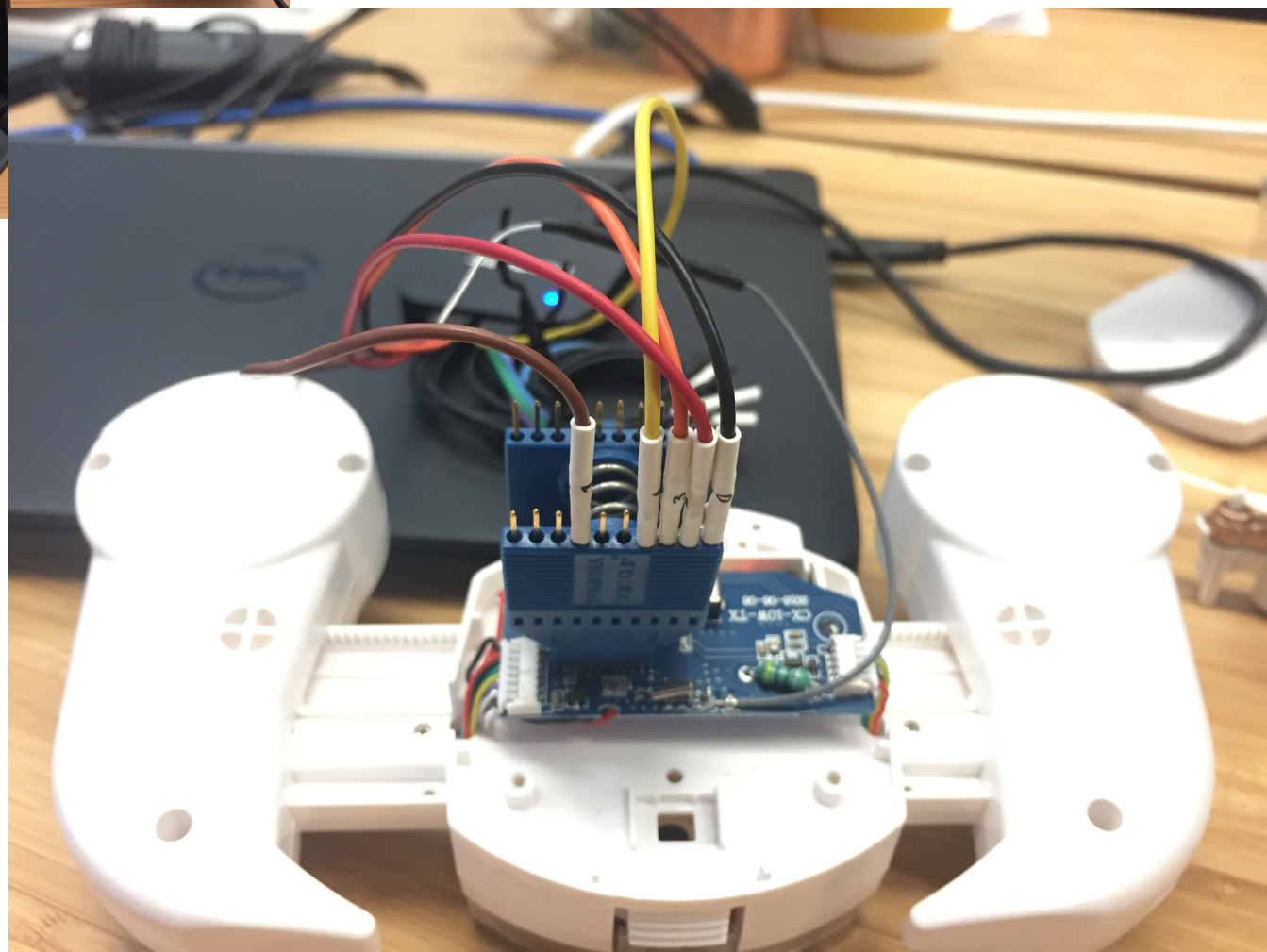
图5.1 XN297LAU芯片引脚功能图

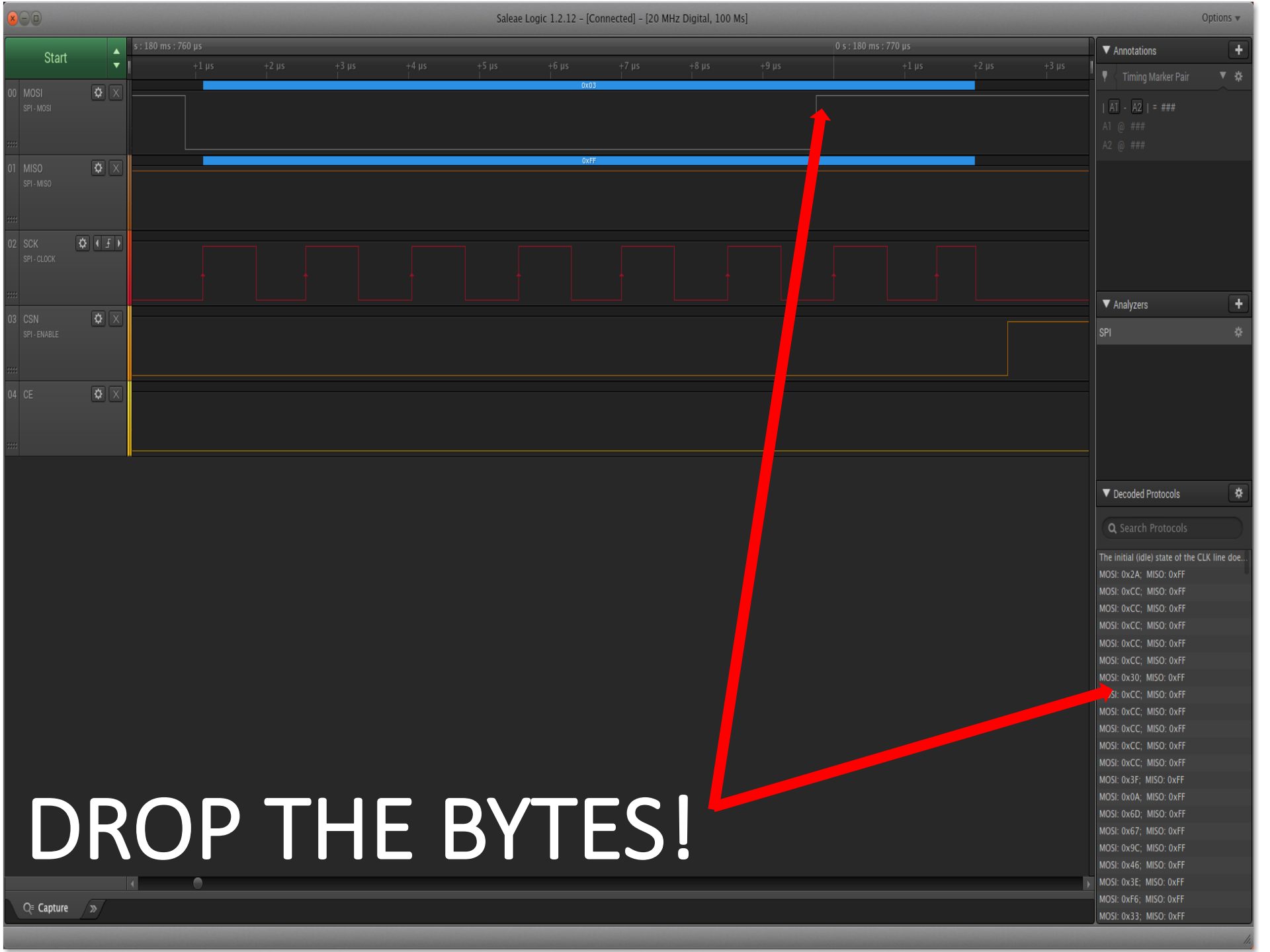
What’s SPI?



- Serial Interface Bus
- Synchronous communication
- Used in embedded devices


DT-9205B+ DIGITAL
AUTO POWER OFF




Saleae Logic 1.2.12 - [Connected] - [20 MHz Digital, 100 Ms]
Options
Start
s : 180 ms : 760 µs
0 s : 180 ms : 770 µs
+1 µs
+2 µs
+3 µs
+4 µs
+5 µs
+6 µs
+7 µs
+8 µs
+9 µs
00 MOSI
SPI - MOSI
0x03
01 MISO
SPI - MISO
0xFF
02 SCK
SPI - CLOCK
03 CSN
SPI - ENABLE
04 CE
Annotations
Timing Marker Pair
A1 - A2 = ###
A1 @ ###
A2 @ ###
Analyzers
SPI
Decoded Protocols
Search Protocols
The initial (idle) state of the CLK line doe...
MOSI: 0x2A; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0x30; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0xCC; MISO: 0xFF
MOSI: 0x3F; MISO: 0xFF
MOSI: 0x0A; MISO: 0xFF
MOSI: 0x6D; MISO: 0xFF
MOSI: 0x67; MISO: 0xFF
MOSI: 0x9C; MISO: 0xFF
MOSI: 0x46; MISO: 0xFF
MOSI: 0x3E; MISO: 0xFF
MOSI: 0xF6; MISO: 0xFF
MOSI: 0x33; MISO: 0xFF
DROP THE BYTES!
Capture

T0: TX Power ON

Interesting bytes on MOSI

Packet ID

```
Time [s],Packet ID,MOSI,MISO
0.000000000000000,0,0x2A,0xFF
0.000013360000000,0,0xCC,0xFF
0.000027680000000,0,0xCC,0xFF
0.000042040000000,0,0xCC,0xFF
0.000056360000000,0,0xCC,0xFF
0.000070680000000,0,0xCC,0xFF
0.000087240000000,1,0x30,0xFF
0.000100560000000,1,0xCC,0xFF
0.000114920000000,1,0xCC,0xFF
0.000129240000000,1,0xCC,0xFF
0.000143560000000,1,0xCC,0xFF
0.000157920000000,1,0xCC,0xFF
0.000176920000000,2,0x3F,0xFF
0.000190280000000,2,0x0A,0xFF
0.000204600000000,2,0x6D,0xFF
0.000218960000000,2,0x67,0xFF
0.000233280000000,2,0x9C,0xFF
0.000247600000000,2,0x46,0xFF
0.000265640000000,3,0x3E,0xFF
0.000279000000000,3,0xF6,0xFF
0.000293320000000,3,0x33,0xFF
0.000307640000000,3,0x5D,0xFF
0.000327200000000,4,0x3A,0xFF
0.000340520000000,4,0x45,0xFF
0.000354840000000,4,0x21,0xFF
```


LOOK! BYTES...
BYTES EVERYWHERE
memegenerator.net

XX: YY:ZZ:....:AA:....

Command name (1 byte)

Payload (0-64bytes)

R_REGISTER	000A AAAA	1 to 5 低字节在前	读状态寄存器 AAAAA=5bit 寄存器地址

Register address (5 LSBits)

W_TX_PAYLOAD	1010 0000	1 to 32/64 低字节在前	写发射数据 ,写操作通常由 0 字节开始。

Payload (up to 64 bytes)

```
Analyzing...
controller [0.000087s]: R_REGISTER [RX_ADDR_P0]: cc:cc:cc:cc:cc
controller [0.000177s]: R_REGISTER [TX_ADDR]: cc:cc:cc:cc:cc
controller [0.000266s]: R_REGISTER [BB_CAL]: 0a:6d:67:9c:46
controller [0.000327s]: R_REGISTER [RF_CAL]: f6:33:5d
controller [0.000429s]: R_REGISTER [N/A]: 45:21:ef:2c:5a:50
controller [0.000460s]: R_REGISTER [DEMOD_CAL]: 01
controller [0.000520s]: R_REGISTER [N/A]: 0b:df:02
controller [0.000550s]: FLUSH_TX
controller [0.000580s]: FLUSH_RX
controller [0.000611s]: R_REGISTER [STATUS]: Data avail in RX FIFO, Packet sent on TX (or ACK
controller [0.000641s]: R_REGISTER [EN_AA]: Auto ACK enabled on data pipe(s)
controller [0.000671s]: R_REGISTER [EN_RXADDR]: RX Addr Pipe(s) enabled: 0,
controller [0.000701s]: R_REGISTER [SETUP_AW]: Address width: 5 bytes
controller [0.000731s]: R_REGISTER [RF_CH]: channel 02
controller [0.000761s]: R_REGISTER [SETUP_RETR]: delay = 0s, retries = 0
controller [0.000791s]: R_REGISTER [RX_PW_P0]: RX PIPE0 payload is 11 bytes
controller [0.000822s]: R_REGISTER [RF_SETUP]: data rate: 2 Mbps, rf power: -0 dBm
controller [0.000852s]: ACTIVATE
controller [0.000882s]: R_REGISTER [DYNPD]: DPL_P1 = 0, DPL_P0 = 0, DPL_P3 = 0, DPL_P2 = 0, D
controller [0.000912s]: R_REGISTER [FEATURE]
controller [0.101291s]: R_REGISTER [DEMOD_CAL]: 01
controller [0.101328s]: W_REGISTER [TX_ADDR]: 00
controller [0.204712s]: W_REGISTER [CONFIG]:  CRC enabled (2 bytes) POWER UP Transmitter
controller [0.204740s]: R_REGISTER [CONFIG]:  CRC enabled (2 bytes) POWER UP Transmitter
controller [0.204755s]: FLUSH_TX
controller [0.204771s]: FLUSH_RX
controller [0.204800s]: R_REGISTER [STATUS]: Data avail in RX FIFO, Packet sent on TX (or ACK
controller [0.204831s]: R_REGISTER [RF_CH]: channel 02
controller [0.207689s]: W_TX_PAYLOAD: aa:d7:4a:98:64:e8:03:dc:05:00:00
controller [0.207717s]: R_REGISTER [CONFIG]:  CRC enabled (2 bytes) POWER UP Transmitter
controller [0.207732s]: FLUSH_TX
controller [0.207748s]: FLUSH_RX
controller [0.207777s]: R_REGISTER [STATUS]: Data avail in RX FIFO, Packet sent on TX (or ACK
controller [0.207808s]: R_REGISTER [RF_CH]: channel 02
controller [0.210666s]: W_TX_PAYLOAD: aa:d7:4a:98:64:dc:05:dc:05:00:00
controller [0.210694s]: W_REGISTER [CONFIG]:  CRC enabled (2 bytes) POWER UP Transmitter
```

What do we know now?

- Data rate: 2Mbps
- Channels: 2, 71, 73, 75, 77 (2 = binding channel, other 4 = ctrl channels)
- Frequency hopping pace: every 3ms
- Frequency hopping behavior: cyclic
- CRC is enabled and it’s 2 bytes long (probably CRC16)
- Packet length is 11 bytes

Over the air

Let’s use some RF-Fu

nuand

Home | Blog | Forums | Get in

bladeRF x115

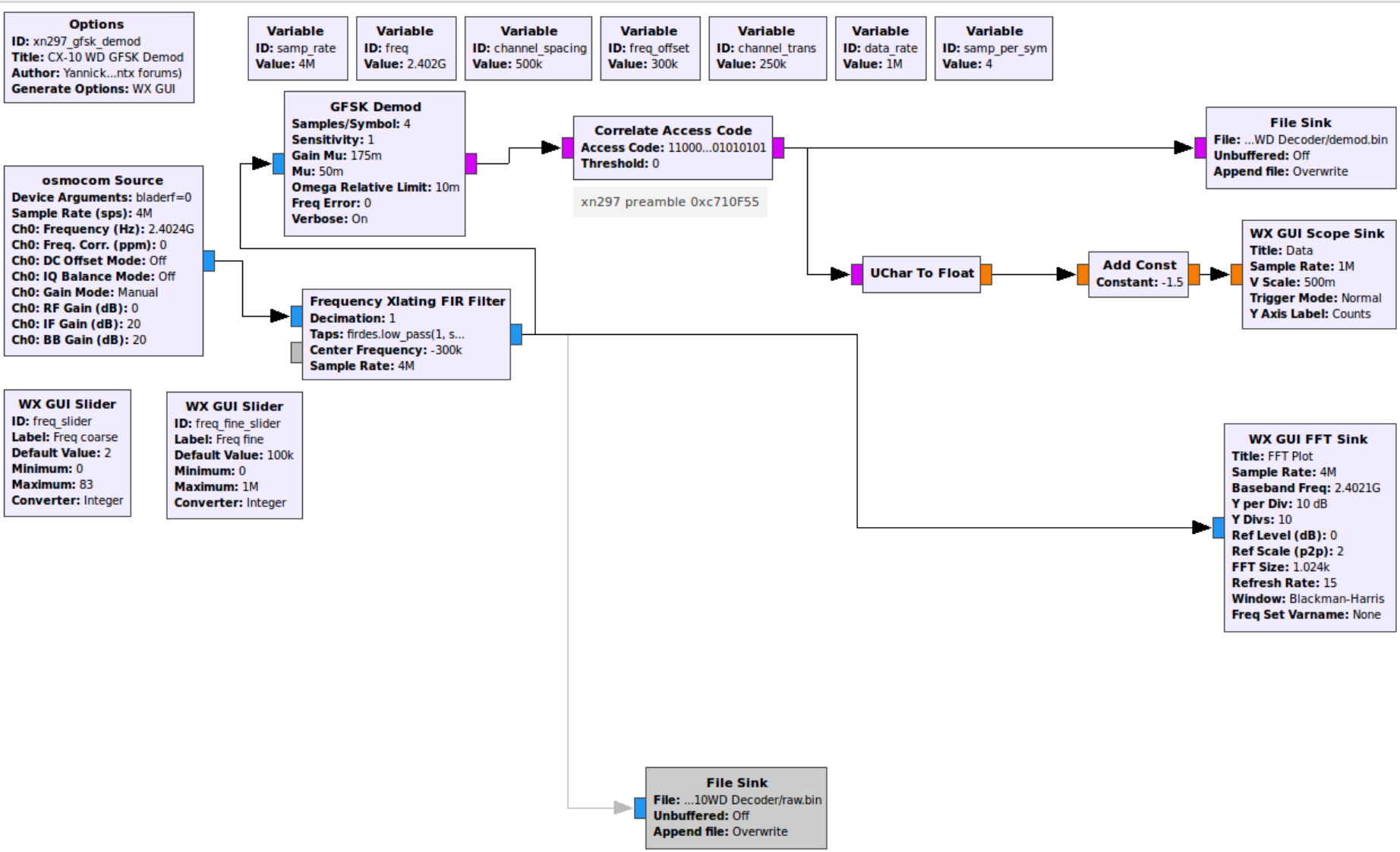
$650.00

The bladeRF x115 comes with a larger 115KLE Cyclone IV FPGA that provides additional room for hardware accelerators and signal processing chains including FFTs, Turbo Decoders, transmit modulators/filters, and receive acquisition correlators for burst modems.

In Stock - Ships by February 13. Order now to reserve spot.

1 Add to cart

GNURadio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

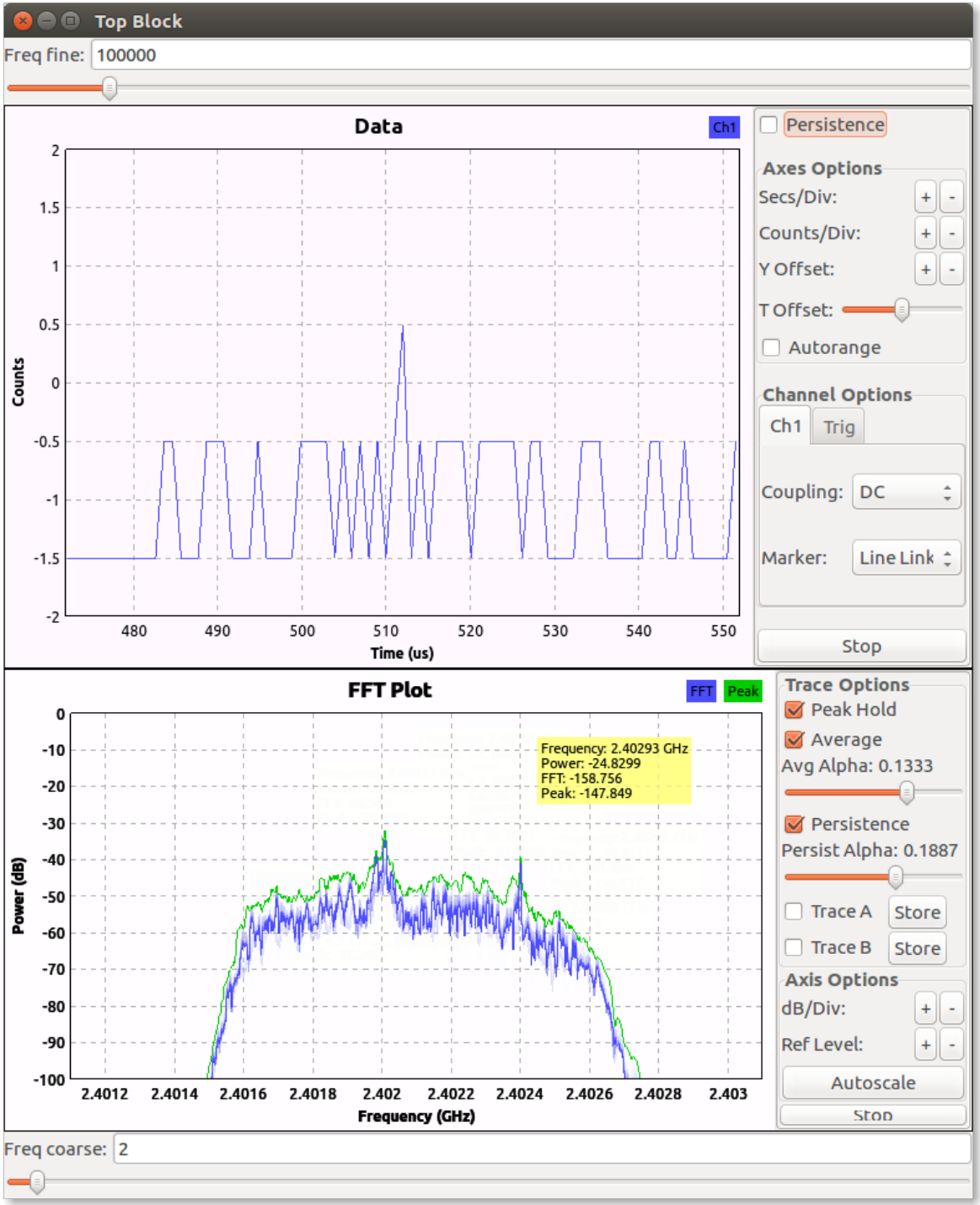


Original from: [deviationTX forums]

(top_block.py:14022): Gtk-WARNING **: gtk_widget_size_allocate(): attempt to allocate widget with width -5 and height 17

(top_block.py:14022): Gtk-WARNING **: gtk_widget_size_allocate(): attempt to allocate widget with width -5 and height 17

Id	Value
Imports	
Variables	
channel_sp	.5e6


Top Block
Freq fine: 100000
Data
Ch1
Counts
Time (us)
Persistence
Axes Options
Secs/Div:
Counts/Div:
Y Offset:
T Offset:
Autorange
Channel Options
Ch1
Trig
Coupling: DC
Marker: Line Link
Stop
FFT Plot
FFT
Peak
Power (dB)
Frequency (GHz)
Frequency: 2.40293 GHz
Power: -24.8299
FFT: -158.756
Peak: -147.849
Trace Options
Peak Hold
Average
Avg Alpha: 0.1333
Persistence
Persist Alpha: 0.1887
Trace A
Store
Trace B
Store
Axis Options
dB/Div:
Ref Level:
Autoscale
Stop
Freq coarse: 2

But…

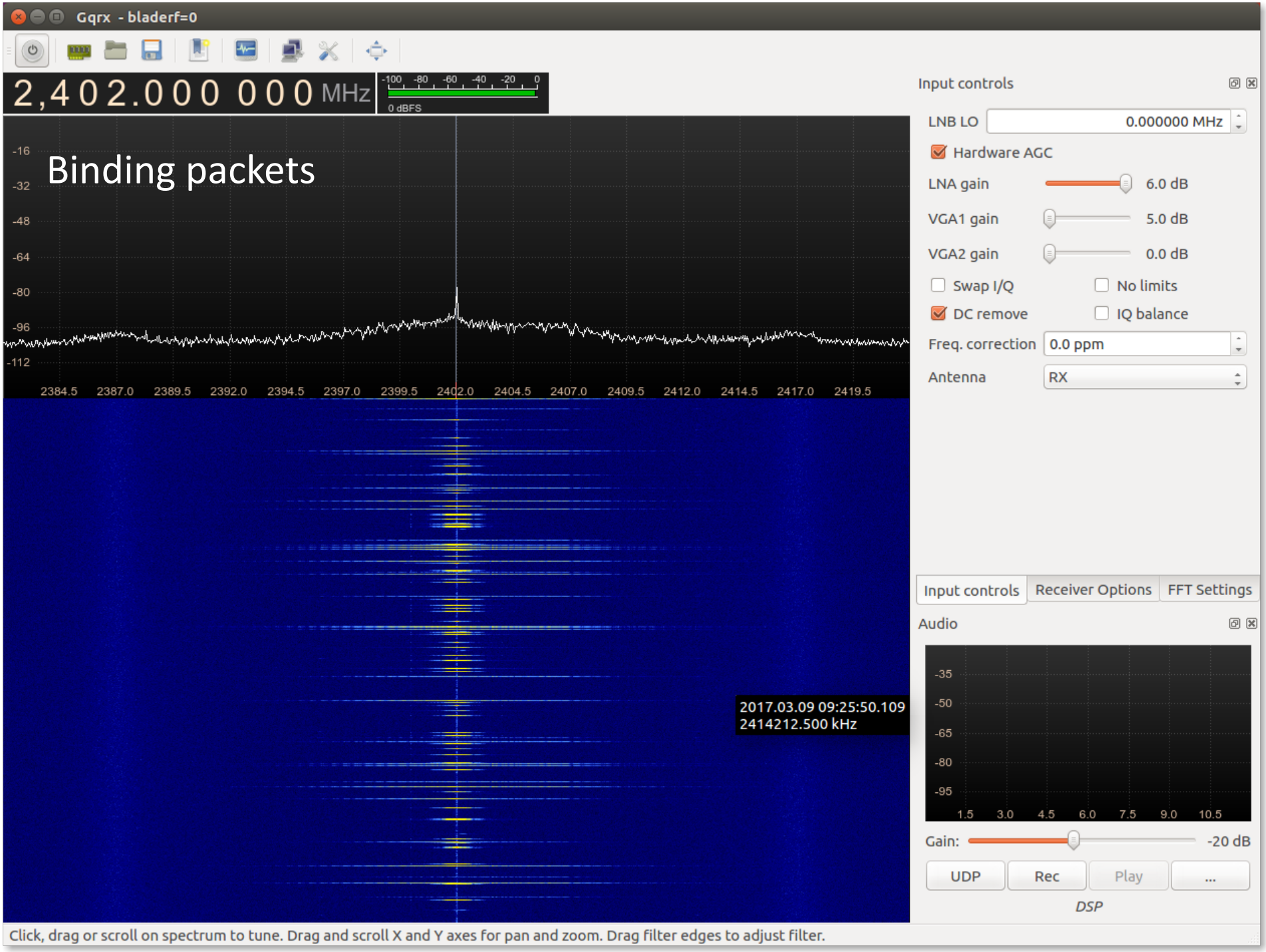
"The channel hopping is generally unpredictable, and Software Defined Radios are slower to retune than the nRF24L radios. This makes it difficult for an SDR based decoder to observe all of the transmitted packets." – *MouseJack, KeySniffer and Beyond by Marc Newlin*

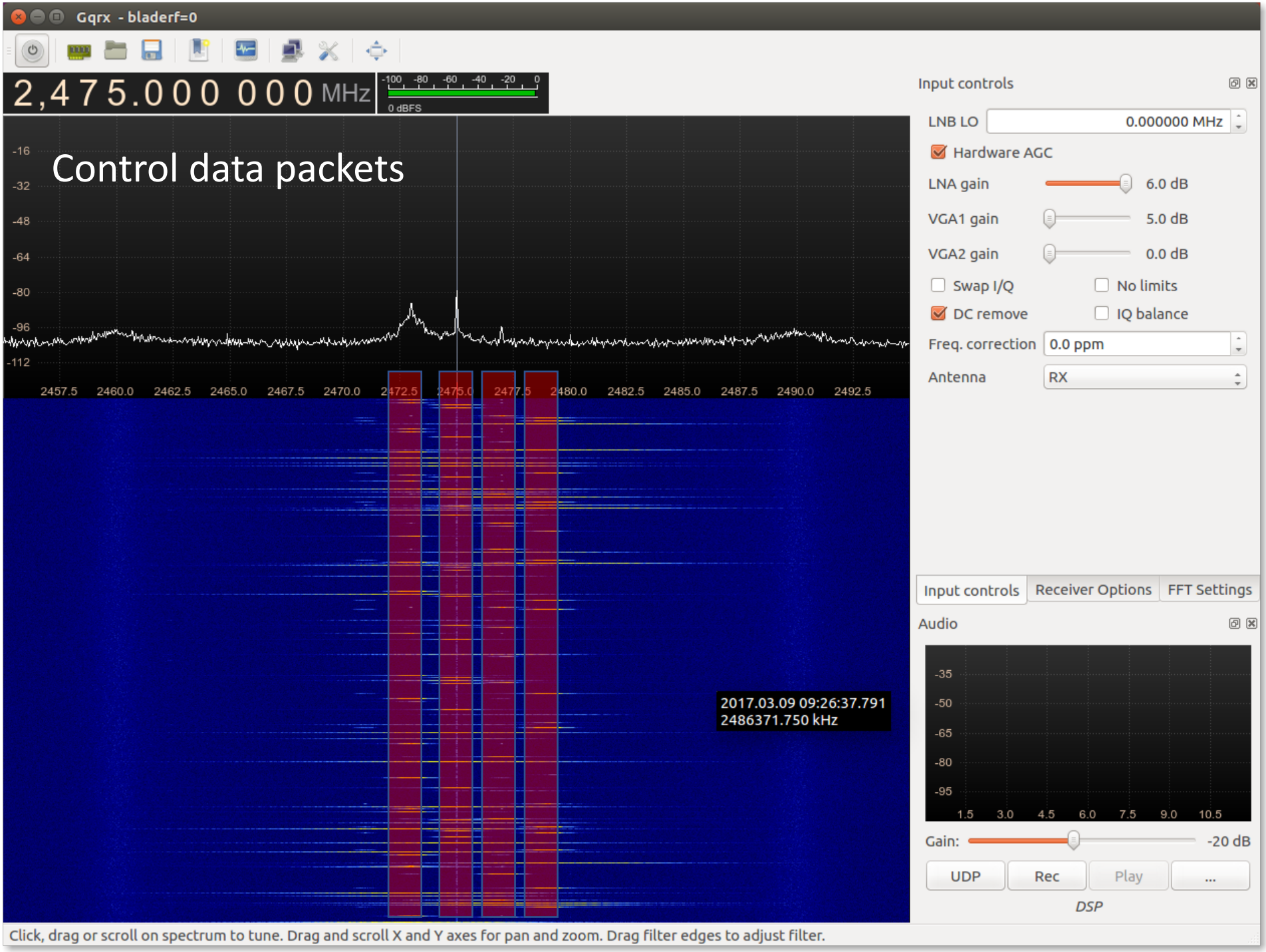
Hardware vs. Software Defined Radio

- Fixed functionality
- Really good at one thing
- Wifi card, wireless mouse dongle, bluetooth dongle, cellular modem, etc

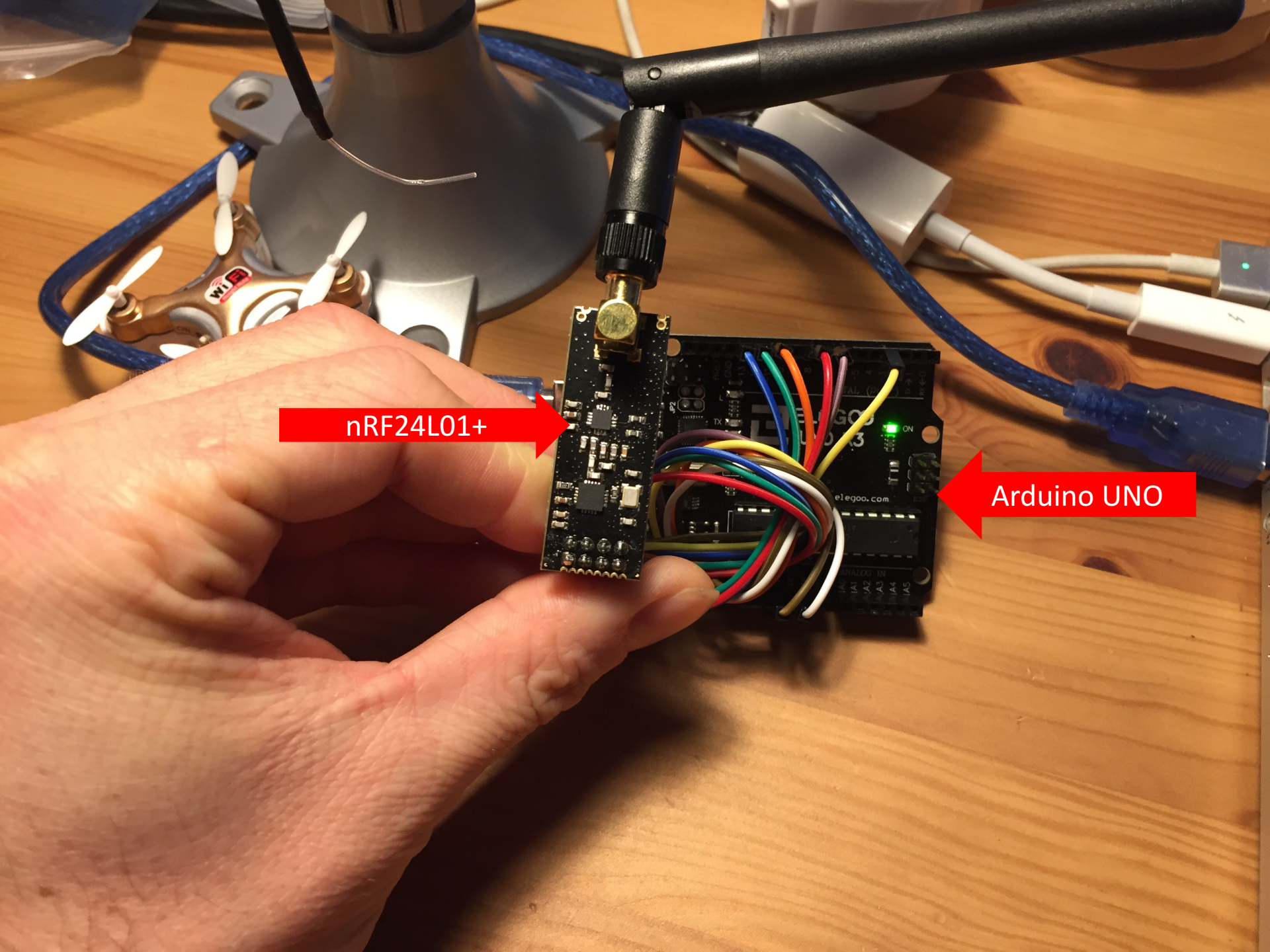
- Reconfigurable on the fly
- Relies on computer or FPGA
- Lots of open source protocol stacks available
- USB and host computer timing limitations

From Marc Newlin @ HITB CommSec 2016


Gqrx - bladerf=0
2,402.000 000 MHz
-100 -80 -60 -40 -20 0
0 dBFS
Binding packets
-16
-32
-48
-64
-80
-96
-112
2384.5 2387.0 2389.5 2392.0 2394.5 2397.0 2399.5 2402.0 2404.5 2407.0 2409.5 2412.0 2414.5 2417.0 2419.5
2017.03.09 09:25:50.109
2414212.500 kHz
Input controls
LNB LO
0.000000 MHz
Hardware AGC
LNA gain
6.0 dB
VGA1 gain
5.0 dB
VGA2 gain
0.0 dB
Swap I/Q
No limits
DC remove
IQ balance
Freq. correction
0.0 ppm
Antenna
RX
Input controls
Receiver Options
FFT Settings
Audio
-35
-50
-65
-80
-95
1.5 3.0 4.5 6.0 7.5 9.0 10.5
Gain:
-20 dB
UDP
Rec
Play
...
DSP
Click, drag or scroll on spectrum to tune. Drag and scroll X and Y axes for pan and zoom. Drag filter edges to adjust filter.


Gqrx - bladerf=0
2,475.000 000 MHz
-100 -80 -60 -40 -20 0
0 dBFS
Control data packets
-16
-32
-48
-64
-80
-96
-112
2457.5 2460.0 2462.5 2465.0 2467.5 2470.0 2472.5 2475.0 2477.5 2480.0 2482.5 2485.0 2487.5 2490.0 2492.5
2017.03.09 09:26:37.791
2486371.750 kHz
Input controls
LNB LO
0.000000 MHz
Hardware AGC
LNA gain
6.0 dB
VGA1 gain
5.0 dB
VGA2 gain
0.0 dB
Swap I/Q
No limits
DC remove
IQ balance
Freq. correction
0.0 ppm
Antenna
RX
Input controls
Receiver Options
FFT Settings
Audio
-35
-50
-65
-80
-95
1.5 3.0 4.5 6.0 7.5 9.0 10.5
Gain:
-20 dB
UDP
Rec
Play
...
DSP
Click, drag or scroll on spectrum to tune. Drag and scroll X and Y axes for pan and zoom. Drag filter edges to adjust filter.

What can we do?


nRF24L01+
Arduino UNO

Promiscuous receiver

- Technique presented in 2011 by Travis Goodspeed
- Capture all bytes sent by nRF24 like chip using illegal register value
- RF ➔ bytes

TRAVIS GOODSPEED'S BLOG

MONDAY, FEBRUARY 7, 2011

Promiscuity is the nRF24L01+'s Duty

by Travis Goodspeed <travis at radiantmachines.com>
extending the work of Thorsten Schröder and Max Moser
of the KeyKeriki v2.0 project.

BLOG ARCHIVE

- ► 2013 (1)
- ► 2012 (3)
- ▼ 2011 (7)
  - ► Dec (1)
  - ► Sep (2)
  - ► May (1)
  - ► Mar (1)
  - ▼ Feb (1)
    - Promiscuity is the nRF24L01+'s Duty

Channel scanner

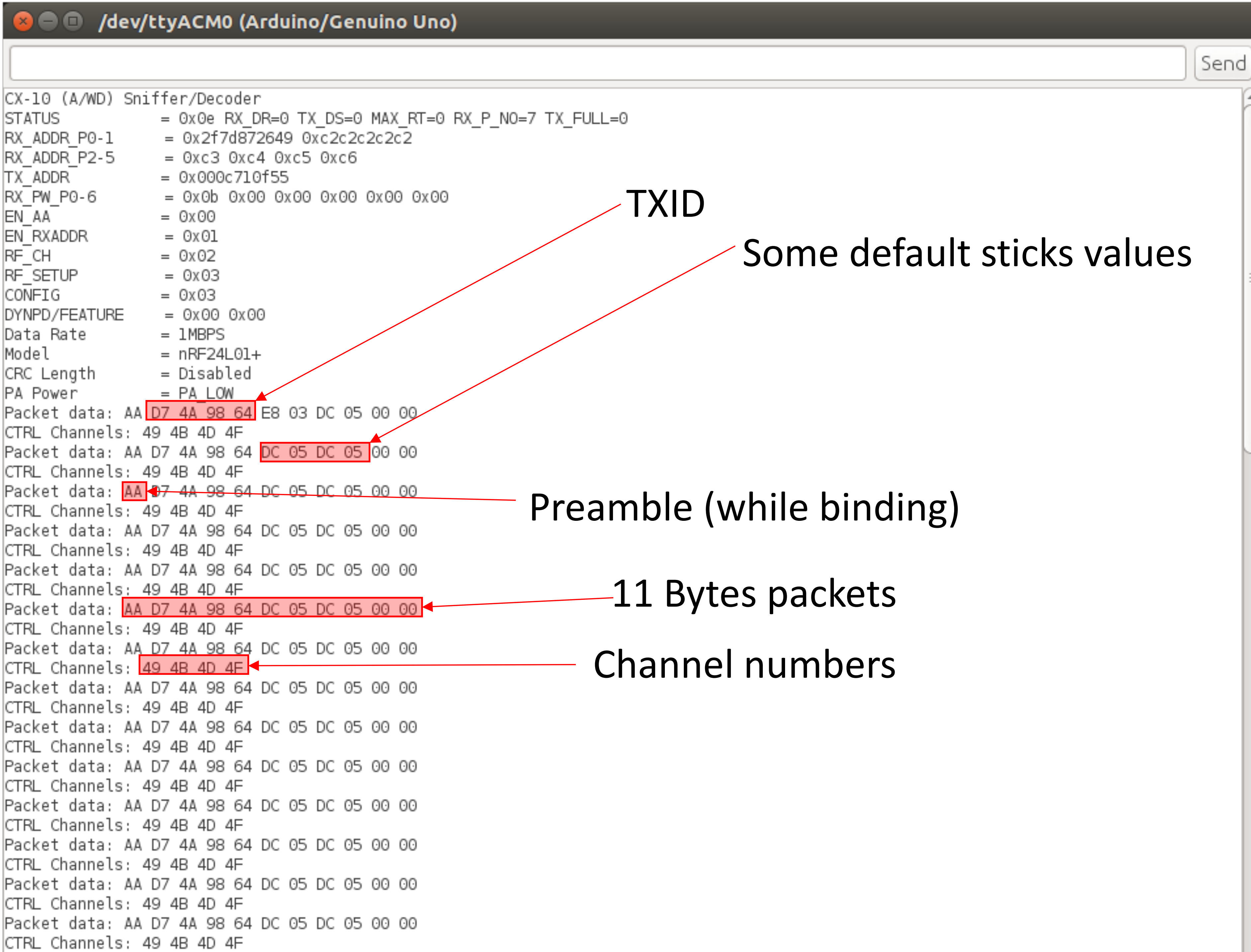
- Cycle through all 83 channels to find some data
- Display the channels where carrier wave is found

- Drawbacks:
  - Data leak on adjacent channels ➔ testing the power of the carrier (RF24.testRPD()) to remove false positives
  - 2.4GHz band is used by Wifi/Bluetooth, … ➔ lot of interferences

VS.

Receive and decode data

- Tune the nRF24 using all the previous information we got earlier (SPI + Spectrum analysis)
- Start sniffing the bytes…


/dev/ttyACM0 (Arduino/Genuino Uno)
Send
CX-10 (A/WD) Sniffer/Decoder
STATUS = 0x0e RX_DR=0 TX_DS=0 MAX_RT=0 RX_P_NO=7 TX_FULL=0
RX_ADDR_P0-1 = 0x2f7d872649 0xc2c2c2c2c2
RX_ADDR_P2-5 = 0xc3 0xc4 0xc5 0xc6
TX_ADDR = 0x000c710f55
RX_PW_P0-6 = 0x0b 0x00 0x00 0x00 0x00 0x00
EN_AA = 0x00
EN_RXADDR = 0x01
RF_CH = 0x02
RF_SETUP = 0x03
CONFIG = 0x03
DYNPD/FEATURE = 0x00 0x00
Data Rate = 1MBPS
Model = nRF24L01+
CRC Length = Disabled
PA Power = PA_LOW
Packet data: AA D7 4A 98 64 E8 03 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
Packet data: AA D7 4A 98 64 DC 05 DC 05 00 00
CTRL Channels: 49 4B 4D 4F
TXID
Some default sticks values
Preamble (while binding)
11 Bytes packets
Channel numbers

55:DC:05:DC:05:DC:05:DC:05:00:00

Aileron

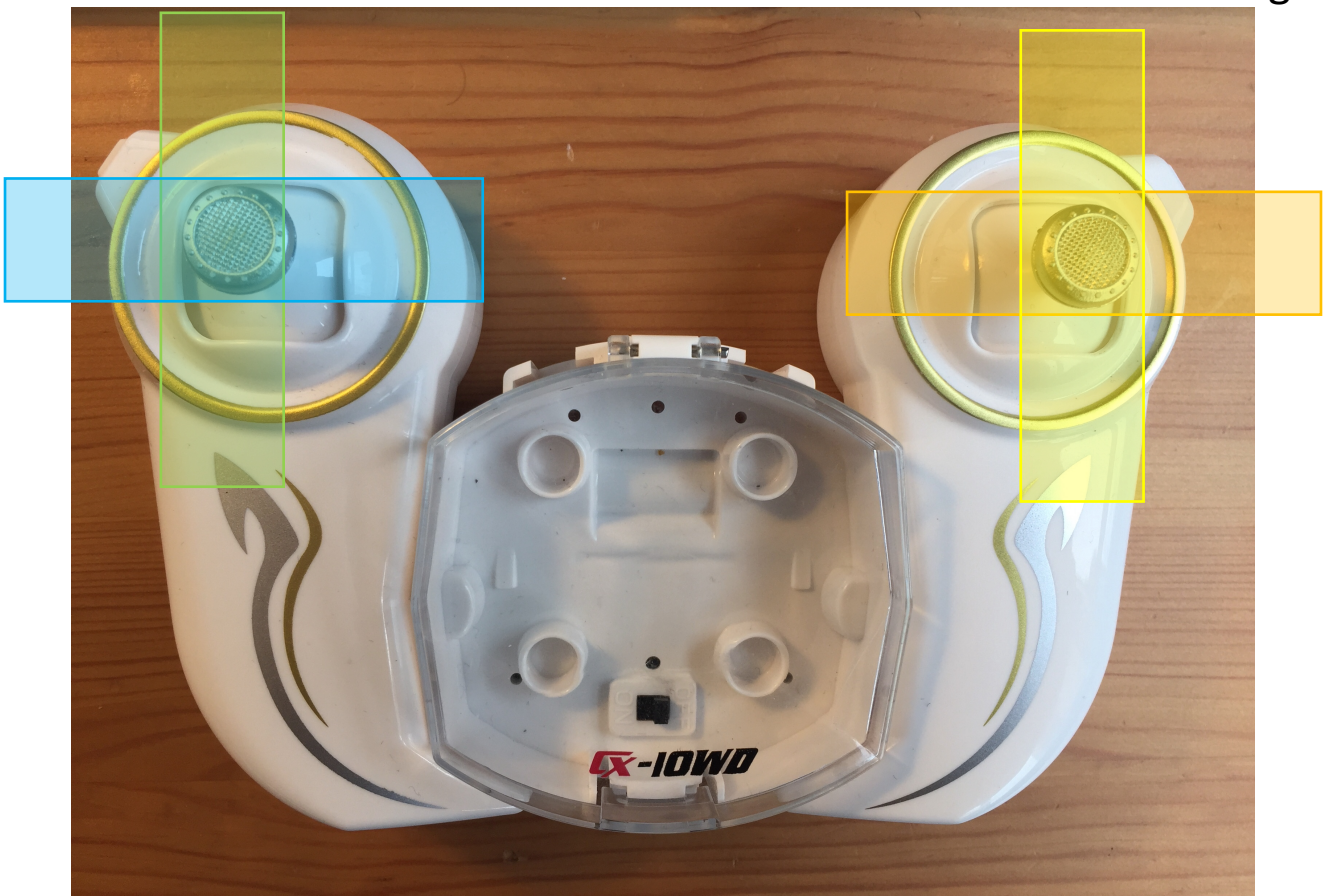
Elevator

Throttle

Rudder

Rate

Take-Off | Landing



Let’s try to hijack!

Hypotheses

- Objective: we want to take over the control of a flying CX-10 WD
- While reversing:
  - No authentication ➔ spoofing TX ID should be a good start
  - No encryption (only data scrambling)
  - Similar protocols already reversed (previous CX-10 models) 😎
- Different papers talk about timing/race condition:
  - Send the commands before the original TX?
  - Talk louder than the original TX?

Other possible attacks

- CX-10WD drone is a WiFi access point:
  - Vulnerable to wifi deauth (Aircrack-ng suite is your friend)
- Jamming (illegal) the control channels:
  - Need RF power amplifier
  - 4 ctrl channels are adjacent: maximum bandwidth needed is 13 MHz


DEMO GODS
PLEASE LET THESE DEMOS WORK
memegenerator.net

Conclusion

Conclusion

- Lot of proven techniques
- You can reproduce them on any IoT/Embedded device
- We targeted toy quads (cheaper)
- More expansive & famous ones also use similar transceiver → same techniques apply (might have to deal with encryption though)

Thanks !

- Shout-out to my new coworker Chi who helped me a lot
- As well as Kevin2600 who found some really good papers/inspirations to get me started in the RF world 🤓


Questions?

Links & References

- OSINT Reverse engineering of the ARFz – *Marc Newlin*
- Mousejack – *Marc Newlin*
- GW008 Drone reverse thread – *@goebish*
- Reverse Engineering a Quadcopter RC Series (4 parts)

ISTUARY

INNOVATION GROUP