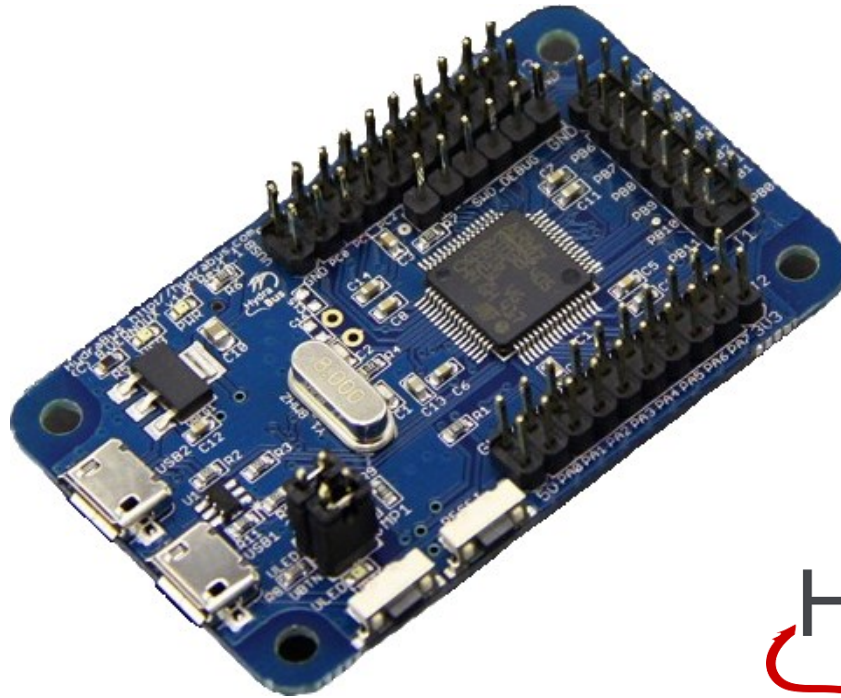


HydraBus

An Open Source Platform



HydraBus/HydraFW GitHub

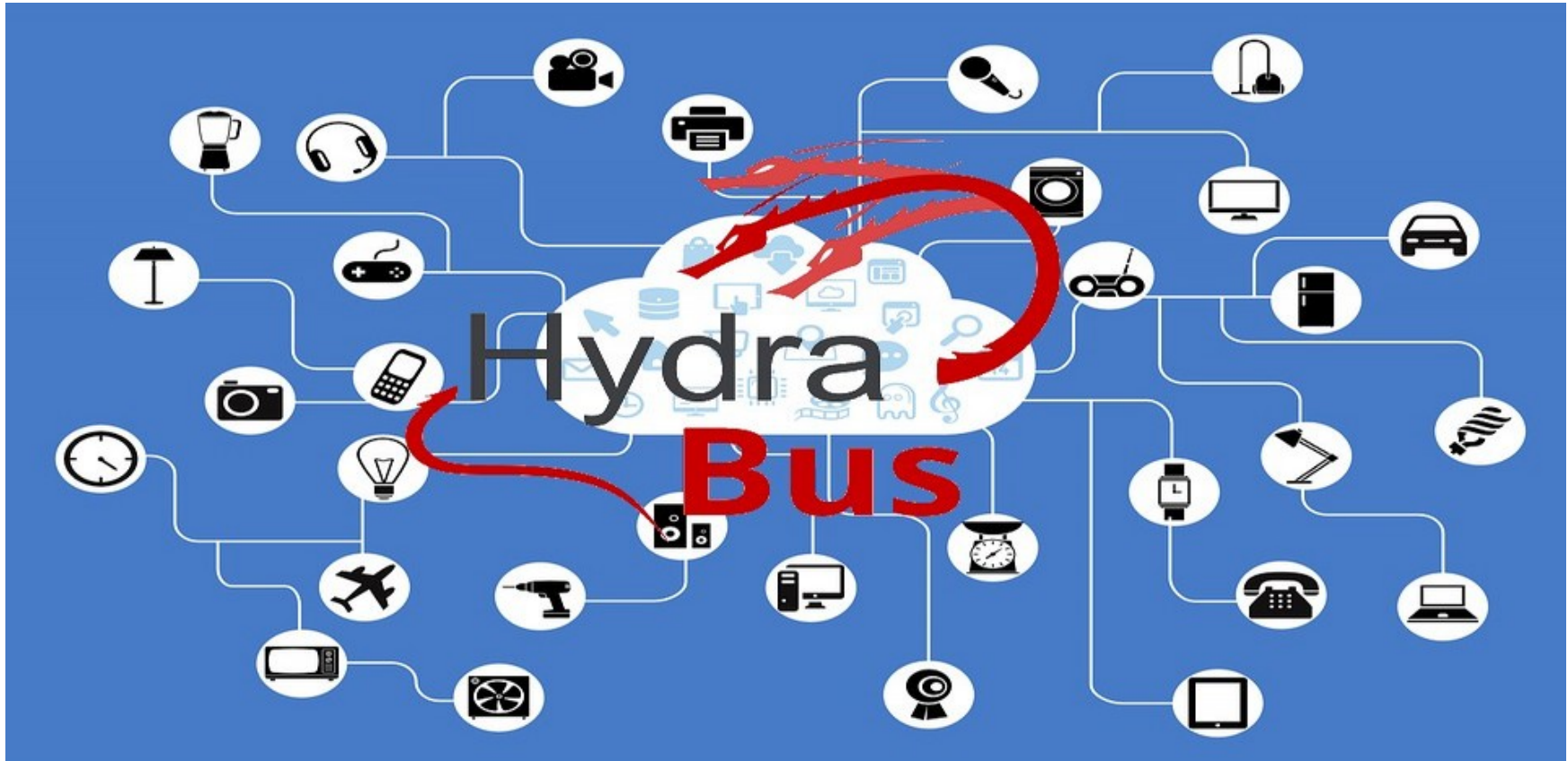
- Hardware / Schematics on GitHub (format Eagle 6.x/7.x)
 - <https://github.com/hydrabus/hydrabus>
 - License CC-BY-NC
- Firmware HydraFW Wiki on GitHub
 - <https://github.com/hydrabus/hydrafw/wiki>
 - Apache License
 - External libraries use their own license

HydraFW

- HydraFW is the embedded software/firmware for HydraBus hardware (support also hw extensions like HydraNFC/HydraFlash...)
- It is compatible with Bus Pirate commands:
http://dangerousprototypes.com/docs/Bus_Pirate_menu_options_guide#Bus_interaction_commands

FW=FirmWare

HydraBus & The IoT



Communication with external world / IoT

- Serial Port (USART/UART)
- I2C Bus: Slow Bus, sensors, memories...
- CAN/LIN Bus: Slow Bus, sensors (mainly automotive)
- SPI Bus: Fast Bus Wifi / BlueTooth / NFC...
- SD/SDIO (microSD, SDIO Bluetooth/Wifi...)
- USB Bus
- ADC & DAC (Analog \Leftrightarrow Digital)
- GPIO (Input/Output)
- Parallel Bus (Nand Flash)

What to do with an HydraBus ?

- The HydraBus is 40x faster than a BusPirate or an Arduino Uno, which is very convenient in order to communicate with fast signals (Serial/Parallel...)
- MCU HydraBus: STM32F415@168MHz Cortex M4F 32bits, 44/IO (84MHz max), 1MB flash, 192KB SRAM, power consumption < 100mA (less than 2mA with low power mode)
- Use cases
 - "Speak" with electronic device/chipset
 - Sensors like Wifi module(ESP32), NFC, Nand Flash, EEPROM...
 - Arduino (SPI, UART ...)
 - "Spy" (MITM) electronic device (SPI/UART/CAN Bus...)
 - Spy Car(CAN), IoT gadgets...
 - "Analyze" signals (analog or digital) with the help of SUMP protocol and open source software like sigrok / PulseView
 - "Reverse engineering" electronic device (IoT ...)
 - Router(WRT54G JTAG, UART), Car, RFID(NFC...), Smart Lighting...

HydraFW main console commands

- Commands OS (chibios):
show system/memory/threads
- Commands sdcard (sd):
mount/umount, erase, cd <dir>, pwd, ls [opt dir], test_perf, cat <filename>, hd <filename>, rm <filename>, mkdir <filename>, script <filename>
- Commands: ADC/DAC, PWM, GPIO
- Bus Modes: SPI, I2C, UART, JTAG, 1-2&3 wire, CAN (HydraCAN), Flash (HydraFlash), NFC (HydraNFC)

HydraFW

Bus Mode protocol Interaction

- Protocol Interaction (similar commands for any protocol support)
 - [Start (for SPI, I2C means Enable Chip Select)
 -] Stop (for SPI, I2C means Disable Chip Select)
 - : Repeat (e.g. r:10)
 - & DELAY us (support optional repeat :)
 - % DELAY ms (support optional repeat :)
 - **123 0x12 0b110 "hello"** Write 8bits val/string (support optional repeat :)
 - **r** Read or **hd** HexDump (support optional repeat :)
 - During a blocking read or write which wait for data(for example Slave mode) you can abort the wait by pressing HydraBus **UBTN**, else you can also wait timeout which is about 10s.
 - Example: HexDump of an SPI EEPROM: [**0b11 0 hd:32**]

Use cases

What is IoT ?

- Device somehow connected to a smartphone or to the Internet
- Which can be an embedded GNU/Linux system or a proprietary firmware
- Which is a SoC or a microcontroller with peripherals or sensors
- ...
- Which basically is a bunch of chips communicating with each other

Bug hunting IoT

- Primary target : Device firmware
 - Main source of vulnerabilities found there
- Not always accessible from the vendor website
 - Firmware update can be encrypted
- Hidden interfaces can be available
 - Serial console or debugging interfaces

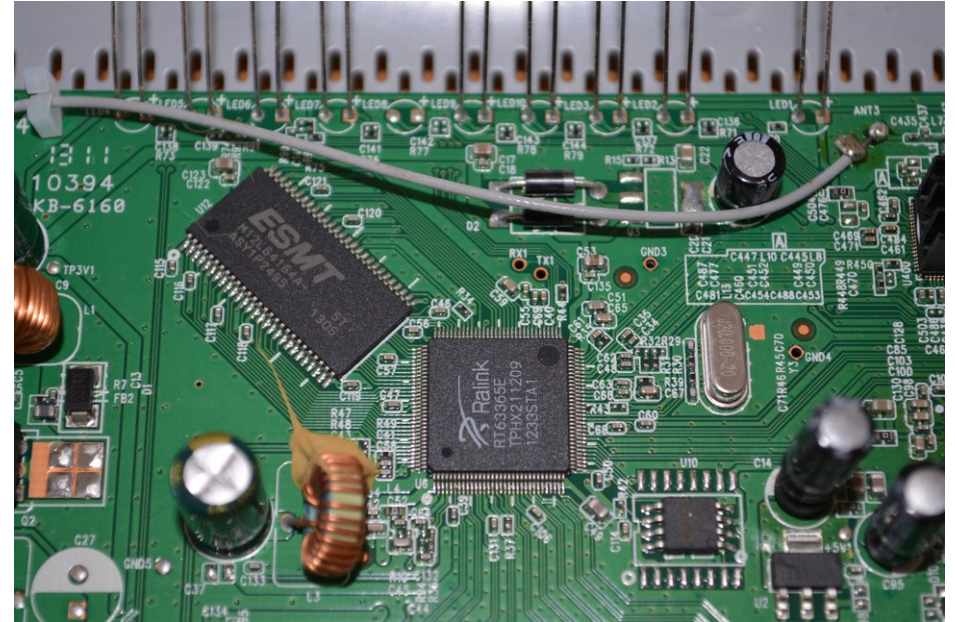
Case 1

- Home router
- Crack open the case
- Undoubtely the trickiest part in the process



Components

- List main components
 - Read their serial number, search for datasheets



\$CHIP1

- Serial number : RT63365e
- Search for serial number online
 - « ADSL2+ processor for residential gateways »
 - MIPS architecture
 - No flash memory
 - Firmware must be stored elsewhere
- Let's skip this for the moment

\$CHIP2

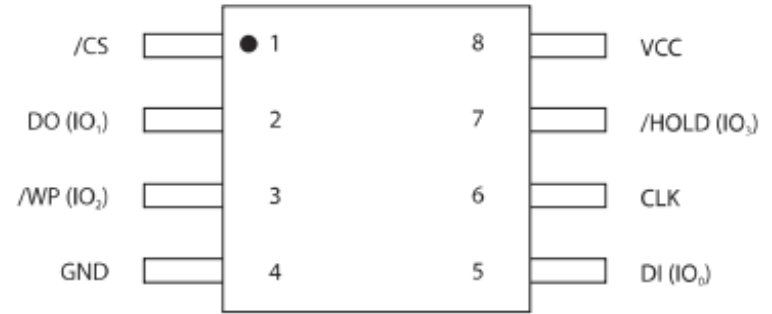
- Serial number : 25q16bvsiq
- Search for serial number online
 - SPI EEPROM
- Electrically-Erasable Programmable Read Only Memory
 - Memory array
 - Data is stored even if the chip is not powered
 - Used to store data
- Probable firmware location !

SPI

- Serial Peripheral Interface
- Bus topology
- Four wires
 - SCLK (Clock)
 - MISO (Master In/Slave Out)
 - MOSI (Master Out/Slave In)
 - SS (Slave Select)

Connect EEPROM to Hydrabus

- From datasheet, get the chip pinout
- From Hydrabus CLI, get the SPI pins
 - *show pins*
- Wire everything together
 - Either wires, hooks or clip



Send EEPROM commands

- Read datasheet, and send correct read command.
- Display hex dump of content

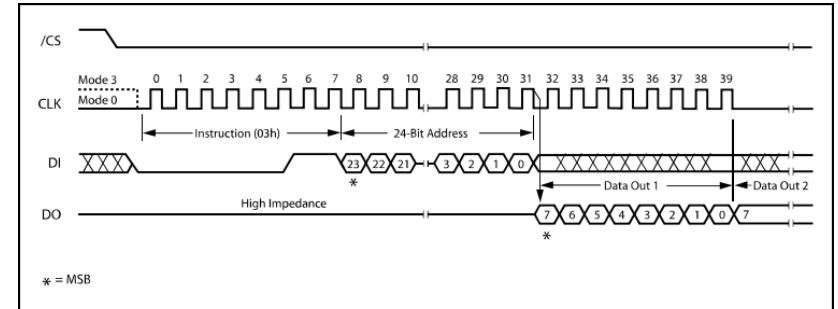


Figure 8. Read Data Instruction Sequence Diagram

```
> spi
Device: SPI1
GPIO resistor: floating
Mode: master
Frequency: 320khz (650khz, 1.31mhz, 2.62mhz, 5.25mhz, 10.50mhz, 21mhz, 42mhz)
Polarity: 0
Phase: 0
Bit order: MSB first
spi1> [ 0x03 0x00:3 hd:16 ]
/CS ENABLED
WRITE: 0x03 0x00 0x00 0x00
00 00 08 25 00 00 10 25 00 00 18 25 00 00 20 25 | ...%...%...%.. %
/CS DISABLED
spi1>
```

Dump the whole image

- Reading bytes is fine to prove that everything is working
- Now, create a script that dumps the whole EEPROM in a file

```
import serial
import struct
ser = serial.Serial('/dev/hydrabus', 115200)

for i in xrange(20):
    ser.write("\x00")
if "BBIO1" not in ser.read(5):
    print "Could not get into bbIO mode"
    Quit()

ser.write('\x01')
if "SPI1" not in ser.read(4):
    print "Cannot set SPI mode"
    quit()

addr = 0
buff=''
print "Reading data"
while (addr < 4096*size):
    ser.write('\x04\x00\x04\x10\x00')
    ser.write('\x03')
    ser.write(struct.pack('>L', addr)[1:])
    ser.read(1)
    buff += ser.read(4096)
    addr+=4096

print ""
end = time.time()

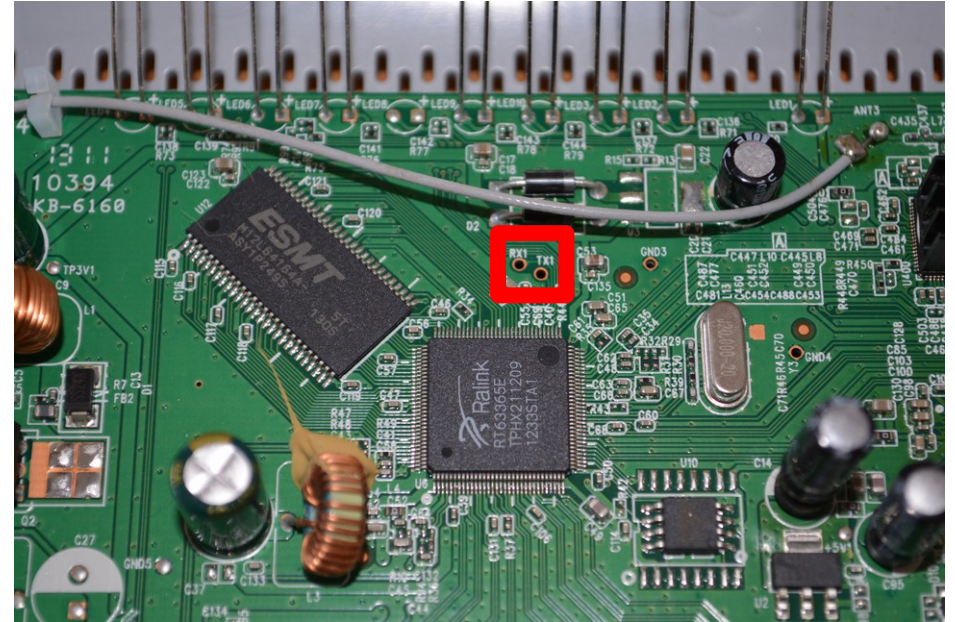
out = open('/tmp/image.bin', 'w')
out.write(buff)
out.close()
```

Result

```
$ strings image.bin
[...]
ATHE
    print help
ATBA
x      change baudrate. 1:38.4k, 2:19.2k, 3:9.6k 4:57.6k
5:115.2k
ATEN
x, (y) set BootExtension Debug Flag (y=password)
ATSE
    show the seed of password generator
ATTI
(h,m,s) change system time to hour:min:sec or show current time
ATDA
(y,m,d) change system date to year/month/day or show current
date
ATDS
    dump RAS stack
ATDT
    dump Boot Module Common Area
ATDU
x,y    dump memory contents from address x for length y
[...]
```

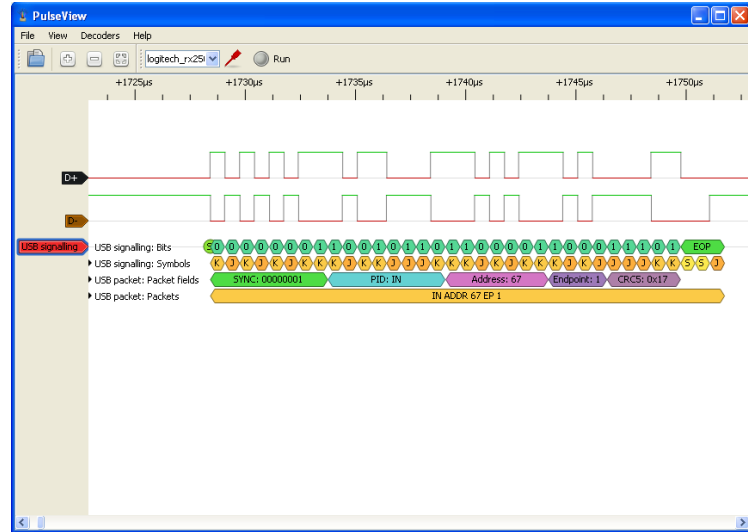
Signal analysis

- Some unused headers are visible, but we don't know what they are used for
 - Labelled RX1 / TX1
- Use the logic analyzer function



Logic analyzer

- Analyses digital signals
 - Only logic states
- Usually coupled with a PC software
 - Decodes logic signals to values
- Captures n samples
 - Capture start can be triggered



Signal analysis

- Connect those headers to Hydrabus
- Open Pulseview, setup the capture
- Search through the available decoders to find a match
 - Requires experience / tests to recognize the protocols

UART Bridge

- Hydrabus can act as a USB / UART bridge
 - In UART mode, use the bridge command
- Serial console is now available on the router

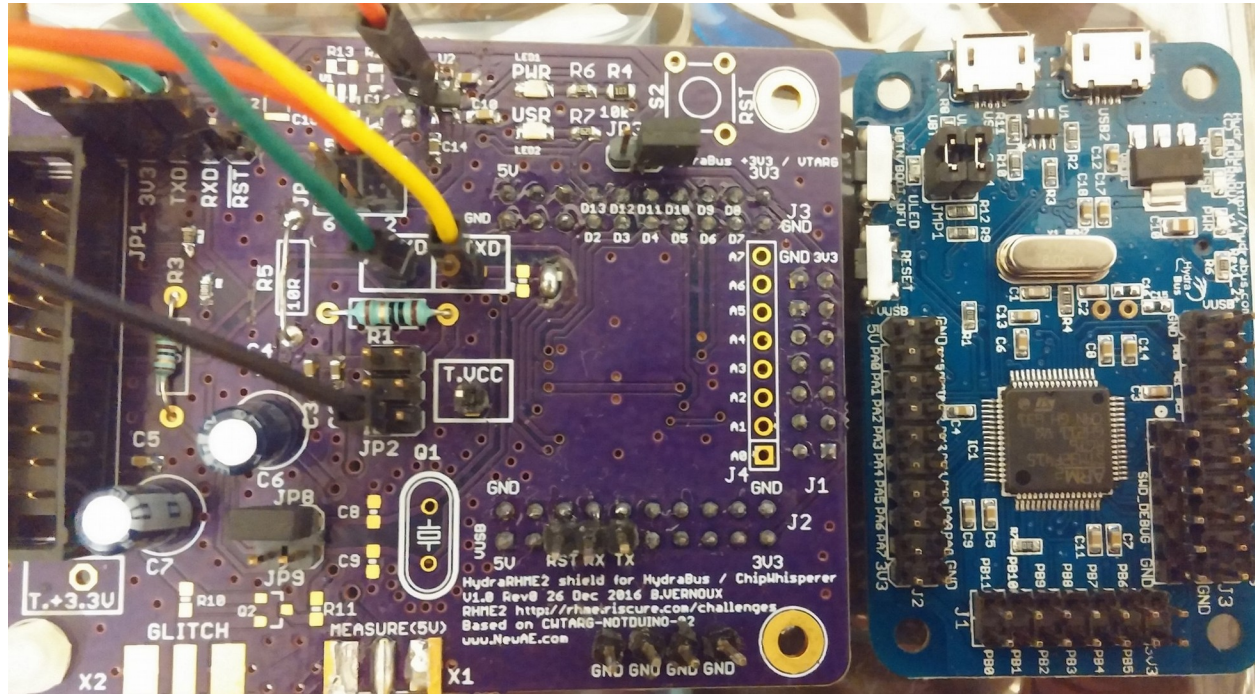
Result

```
Bootbase Version: VTC_SPI1.22 | 2012/4/12 16:30:00
RAM: Size = 8192 Kbytes
Found SPI Flash 2MiB Winbond W25Q16 at 0xbfc00000
SPI Flash Quad Enable
Turn off Quad Mode

RAS Version: 3.0.0 Build 120524 Rel.05221
System ID: $2.12.58.23(G04.BZ.4)3.20.17.0| 2012/05/18 20120518_V003 | 2012/05/18

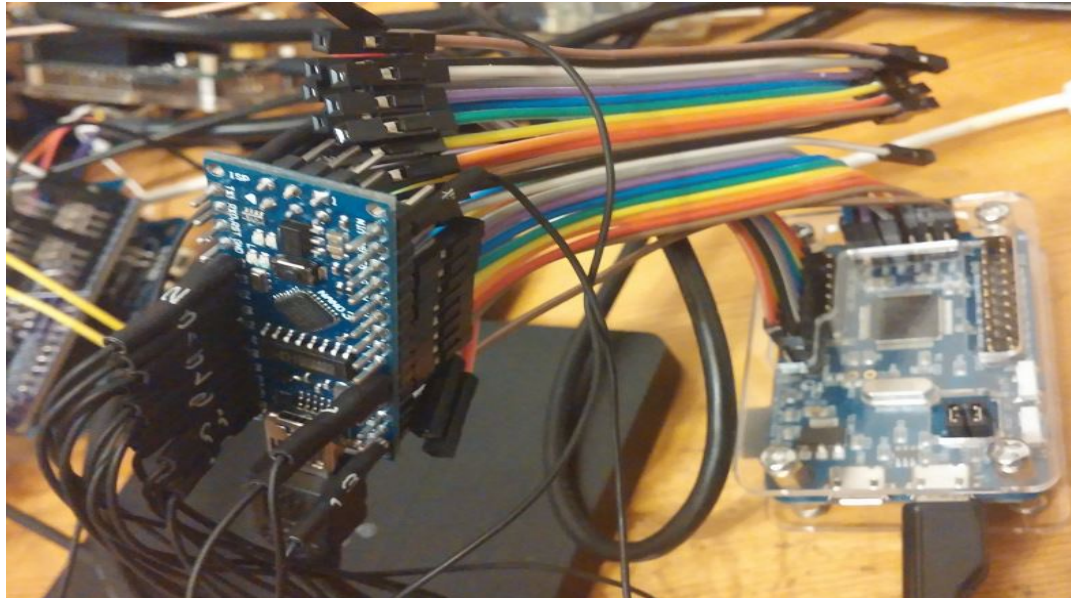
Press any key to enter debug mode within 3 seconds.
.....
Enter Debug Mode
ATHE
===== Debug Command Listing =====
AT just answer OK
ATHE print help
ATBax change baudrate. 1:38.4k, 2:19.2k, 3:9.6k 4:57.6k 5:115.2k
ATENx,(y) set BootExtension Debug Flag (y=password)
ATSE show the seed of password generator
ATTI(h,m,s) change system time to hour:min:sec or show current time
ATDA(y,m,d) change system date to year/month/day or show current date
ATDS dump RAS stack
ATDT dump Boot Module Common Area
ATDUX,y dump memory contents from address x for length y
ATRBx display the 8-bit value of address x
ATRWx display the 16-bit value of address x
ATRLx display the 32-bit value of address x
ATGO(x) run program at addr x or boot router
ATGR boot router
ATGT run Hardware Test Program
ARTWw,x,y,(z) RAM test level w, from address x to y (z iterations)
ATSH dump manufacturer related data in ROM
ATDOx,y download from address x for length y to PC via XMODEM
ATTD download router configuration to PC via XMODEM
ATUR upload router firmware to flash ROM
```

RHME2 Challenge



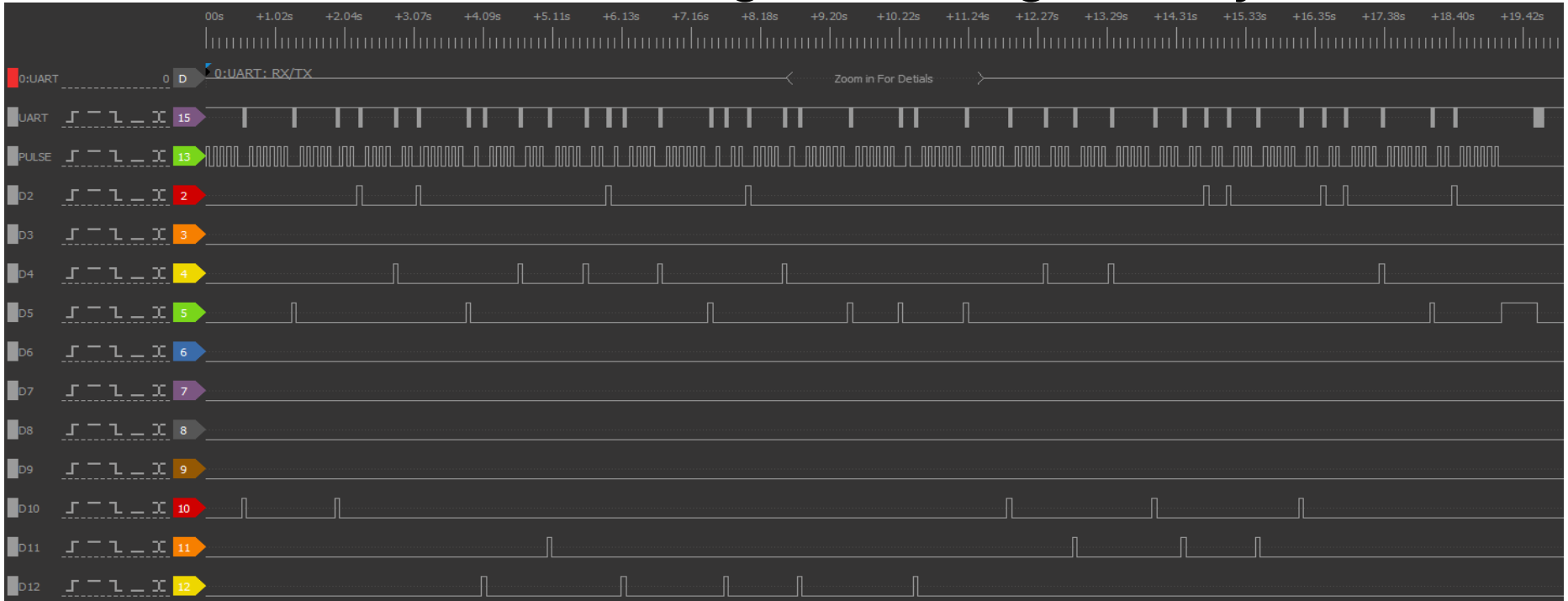
RHME2 - Whac The Mole

- Whac The Mole Challenge
 - <https://github.com/hydrabus/rhme-2016/blob/master/Other/WhackTheMole.md>



RHME2 - Whac The Mole

- Whac The Mole Challenge with Logic Analyzer



RHME2 - Secret Sauce

- Secret Sauce Challenge
 - This challenge ask for a password so the idea was to recover it using a timing attack with the help of HydraBus
 - <https://github.com/hydrabus/rhme-2016/blob/master/Other/SecretSauce.md>

RHME2 - Fiasco (Fault Injection)

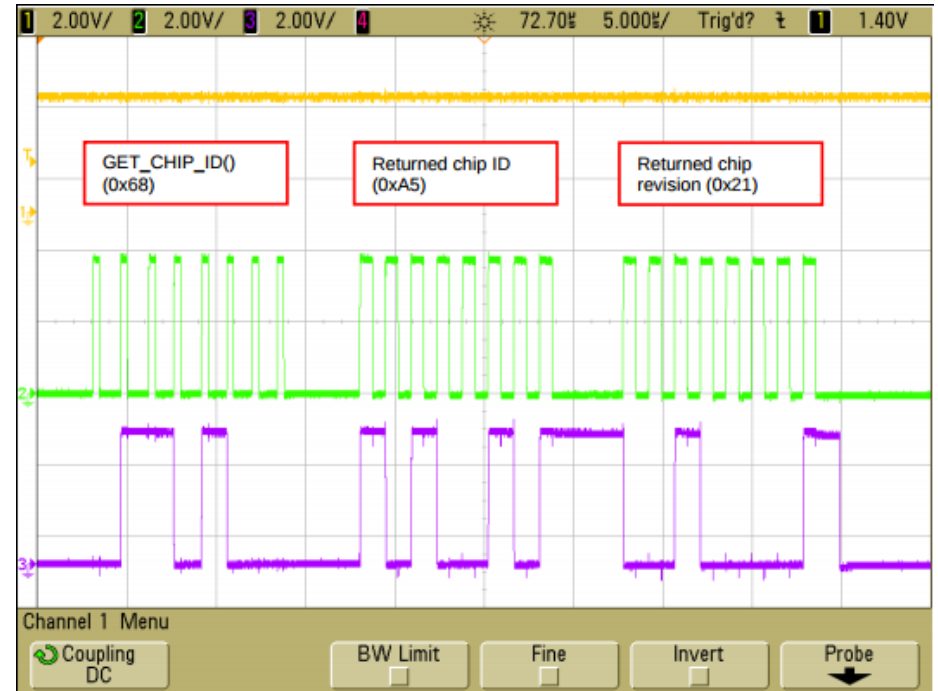
- This challenge asks for a password and the idea is to do a VCC Glitch on the Arduino board in order to skip/jump over the check and display the flag
 - Results with HydraBus + Custom Board with MOSFET
 - Please write your password: gpio glitch trigger PB0 pin PC15 length 100 offsets 191200
Good try, cheater!^M
Chip locked^M
 - Please write your password: gpio glitch trigger PB0 pin PC15 length 100 offsets 191300
Chip unlocked^M
Your flag is: 02ab16ab3729fb2c2ec313e4669d319e
 - <https://github.com/hydrabus/rhme-2016/blob/master/FaultInjection/Fiasco.md>

Case 2

- Unspecified board, sorry
- Uses CC2530 microcontroller
- Debug port available
 - Uses custom debugging protocol
 - No ccDebugger at hand at that time

Protocol details

- Application note found on TI website
- Simple two wire protocol
 - Clock / Data
 - Master drives the clock
 - Data channel is bidirectional



Dumping

- Using 2-wire mode, able to communicate with the chip and dump its flash memory

Result

```
RESET_0:
; FUNCTION CHUNK AT 000006CE SIZE 00000016 BYTES
75 D0 00      mov     PSW, #0          ; Program Status Word Register
75 81 3F      mov     SP, #0x80       ; Stack Pointer
75 0C 00      mov     TF0+1, #0x80    ; Move (Op1 <- Op2)
75 0D 01      mov     TF0+2, #1      ; Move (Op1 <- Op2)
02 06 CE      ljmp    bank0_6CE     ; Long Jump
; End of function RESET_0
```

```
; START OF FUNCTION CHUNK FOR RESET_0
bank0_6CE:
; Clear Operand (0)
E4           clr     A
90 01 80     mov     DPTR, #0x180    ; Move (Op1 <- Op2)
78 11       mov     R0, #0x11   ; Move (Op1 <- Op2)
79 02       mov     R1, #2    ; Move (Op1 <- Op2)
80 02       sjmp   bank0_6DA ; Short jump
```

```
bank0_6DA:
; Decrement Operand and JNZ
D8 FC      djnz   R0, bank0_6D8
```

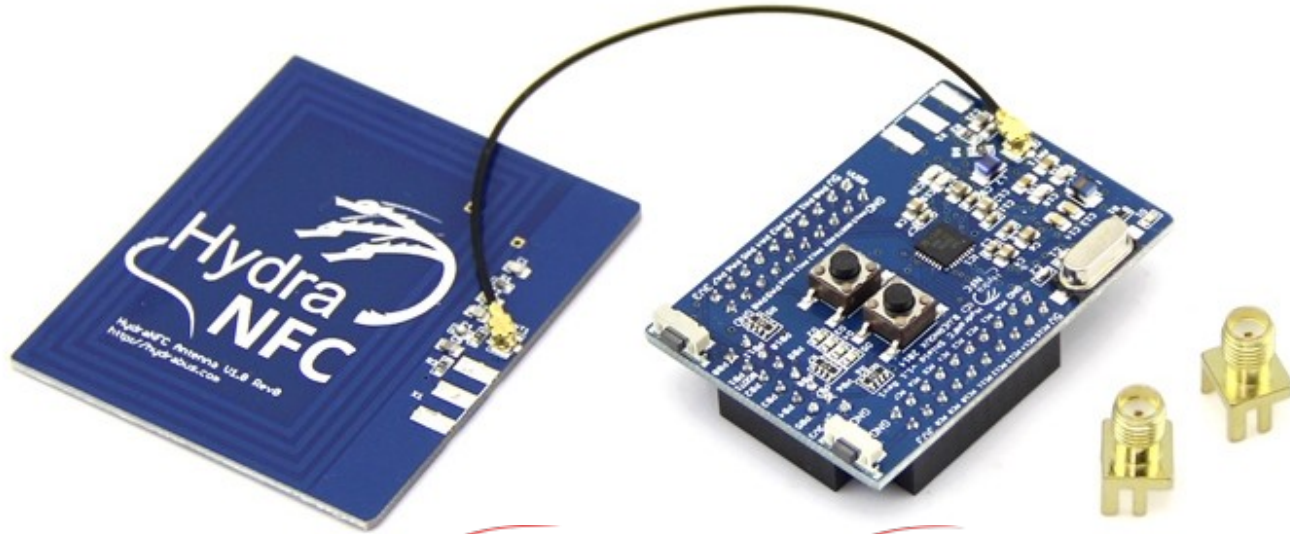
```
D9 FA      djnz   R1, bank0_6D8 ; Decrement Operand and JNZ
```

```
bank0_6D8:
; Move from/to external RAM
F0        movx  @DPTR, A
A3        inc   DPTR ; Increment Operand
```

```
12 04 72   lcall  main      ; Long Subroutine Call
12 07 5B   lcall  endless_loop ; Long Subroutine Call
; END OF FUNCTION CHUNK FOR RESET_0
```

Shields

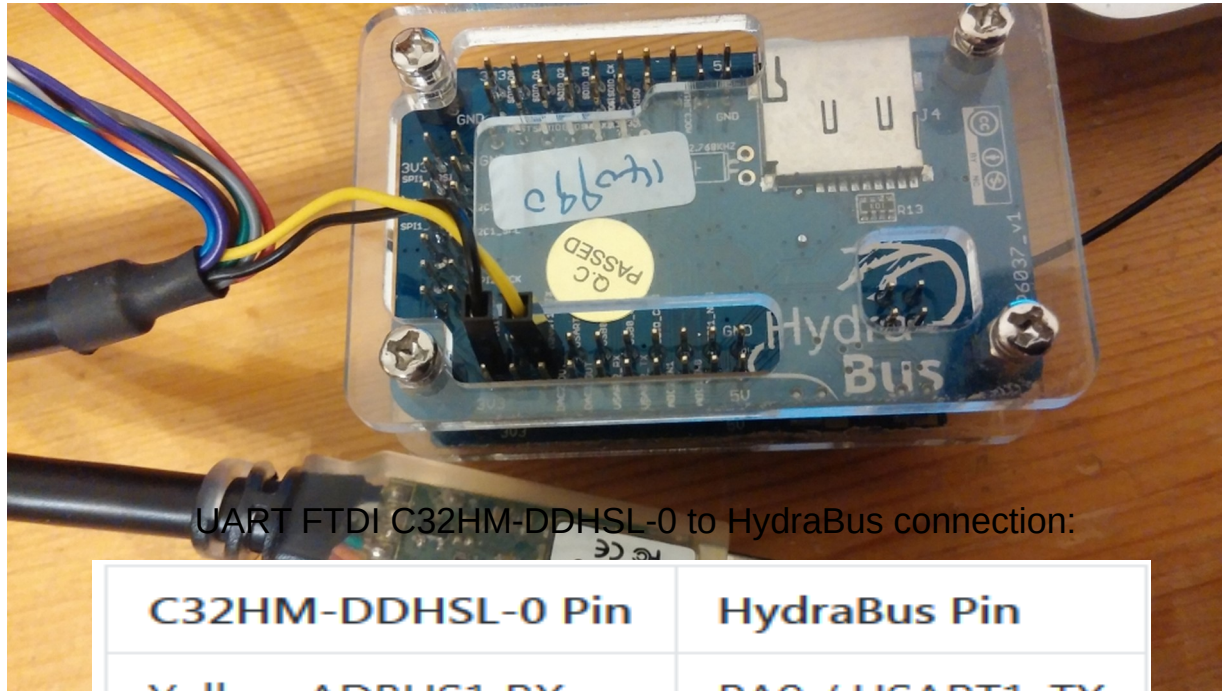
HydraNFC



HydraBus+HydraNFC Firmware

- Actual Firmware features (HydraFW):
 - Read UID NFC Vicinity/ISO15693 and Mifare
 - Read Data on Mifare UL
 - Emulation ISO14443A/Mifare UL/Classic (Alpha)
 - Sniffer ISO14443A with unique hard real-time infinite trace mode (requires FTDI external hw & PC with hydratool sw)
 - Autonomous sniffer ISO14443A (Mifare ...) include data from TAG & READER (data saved in microSD)
 - HydraFW HydraNFC online guide see:
 - <https://github.com/hydrabus/hydralfw/wiki/HydraFW-HYDRANFC-guide>

Sniffer real-time infinite trace mode

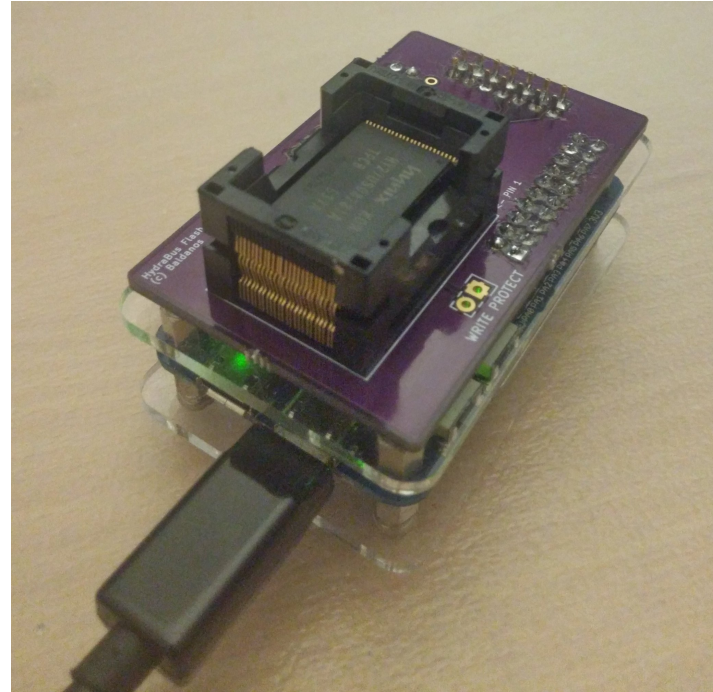


UART FTDI C32HM-DDHSL-0 to HydraBus connection:

C32HM-DDHSL-0 Pin	HydraBus Pin
Yellow ADBUS1 RX	PA9 / USART1_TX
Black GND	GND (near PA9)

HydraFlash

- Designed to dump Flash NAND chips
 - Found in more and more devices
- No hardware support from the MCU
 - Uses GPIO in bitbang mode



HydraFlash

- Uses a fork of DumpFlash to handle commands
 - Some manufacturers use different commands
 - Already manages OOB
- Decent reading and writing speeds
 - ~200KB/s on test chip

```
$ python2 DumpFlash.py -d /dev/hydrabus -i
Into BBIO mode
Switching to flash mode
Setting chip enable
Full ID: AD73AD73AD73
ID Length: 6
Name: NAND 16MiB 3,3V 8-bit
ID: 0x73
Page size: 0x200
OOB size: 0x10
Page count: 0x8000
Size: 0x10
Erase size: 0x4000
Block count: 1024
Options: 0
Address cycle: 3
Bits per Cell: 4
Manufacturer: Hynix
```

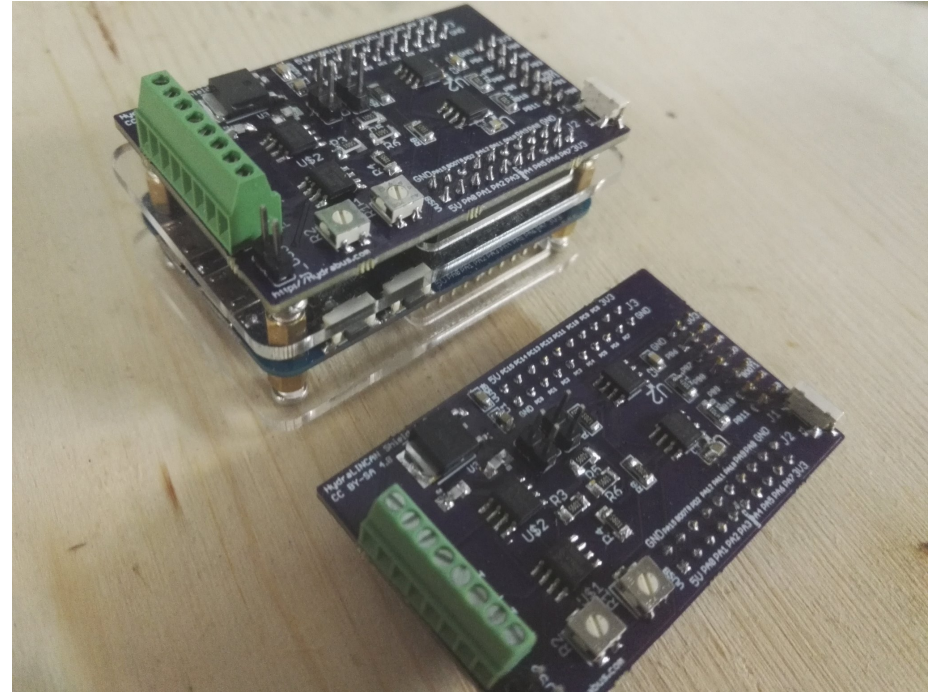
HydraFlash – Fun facts

- Got some flash chips from eBay
- Branded as new
- Unfortunately, no juicy information :(

```
$strings /tmp/dump
[...]
Media is write-protected!
FCU failed on ECC/CRC error!
FCU general error!
FCU%s timed out!
  Burst
  'Copyright (c) 1996-2004 Express Logic Inc.
  * FileX LX4180/Green Hills Version G3.1a.3.1a *
  /home/sandbox/sde/lib/c/time/offtime.c
  /home/sandbox/sde/lib/c/time/tzfile.h
  ### Battery_Check : byPowerOnLevelAfterDummy = %
  Battery_Check : NiMH Battery =====
  Battery_Check : BEFORE LENS MOVE
  Battery_Check : sBattery.byLevel
  < BAT_NO_MOVE_LENS_LEVEL
  [...]
```

HydraLINCAN

- Designed to handle CAN and LIN buses communication
 - Mostly found in automotive
- Made by smillier
- Already tested on my car
 - Still alive ;-)



HydraLINCAN

- CLI and BBIO already implemented and working
- SLCAN implementation is underway
 - Able to use all utilities provided by can-utils

Alternative firmwares

- Micropython
 - Python 3.x for microcontrollers
 - Official support
 - <http://micropython.org>
- BlackMagic
 - JTAG/SWD probe / GDB server
 - Official support
 - <https://github.com/blacksphere/blackmagic>

Project status

- Lots of added features last year
 - Frequency measurement
 - Hexdump mode
 - 1-wire mode
 - AVR programming
 - NAND Flash support
 - Hex escapes
 - ...

Project status – cont.

- Project is getting close to 1.0
 - Will be the first stable version
 - Some modes need to be added to provide full set of features
 - I2c slave mode
 - Data sniffers
 - I2c
 - [1,2,3]-wire

Hydrabus Workshop

- Tomorrow afternoon, open to all
- Many different activities
 - HydraFW hackathon
 - Hydrabus kits offered for first 2 merged PRs made during RMLL
 - Test Hydrabus on practice targets
 - Learn signal analysis
 - Get yours !

Conclusions

- Hydrabus will not replace dedicated tools
- However, nice all-in-one device that can be used for hackers and makers for quick prototyping and development
- Still requires some technical background to be used efficiently

Thank you !

Shameless promotion

- Blackalps
 - Security conference
 - Yverdon, Switzerland
 - November 2017
 - CFP open until 31.07.2017

<http://www.blackalps.ch>