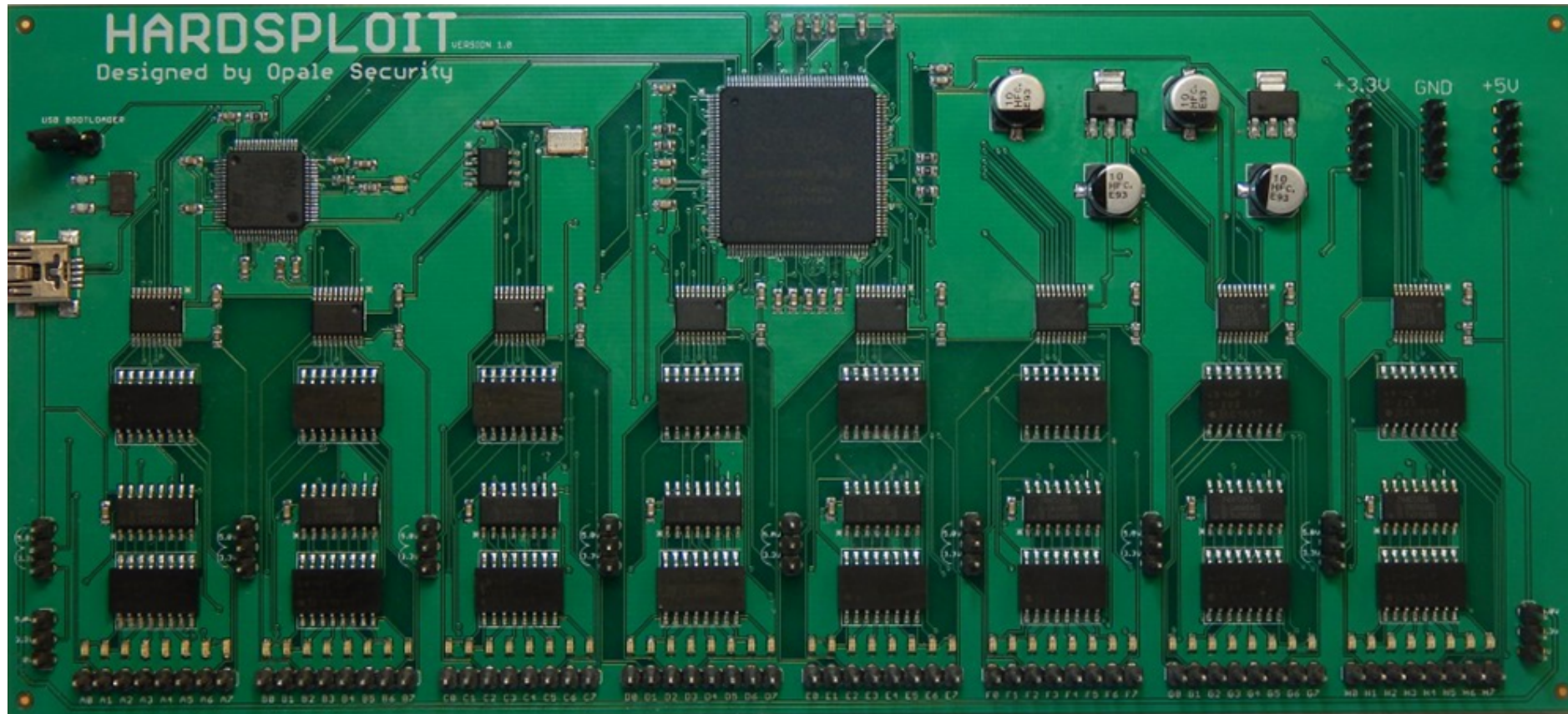


HARDSPLOIT

Framework for Hardware Security Audit

a bridge between hardware & a software pentester



Who we are ?



- Julien MOINARD
 - Electronic engineer @opale-security (French company)
 - Security consultant, Hardware & Software pentester
 - Team project leader of Hardsploit
 - DIY enthusiast
 - Blackhat, NullCon, HIP, CansecWest, 32C3 speaker & trainer
- Yann ALLAIN
 - CEO
 - Blackhat, HackInThebox, HIP, speaker & trainer
 - Cybersecurity veteran (+ 20 years) / (old) electronic engineer
 - Former CSO of ACCOR (software domain)

Opale Security in 1 slide



OPALE
security

A PRAGMATIC APPROACH FOR YOUR IT & IoT SECURITY

TRAININGS

CONSULTING

PRODUCTS

WWW.OPALE-SECURITY.COM

CONTACT@OPALE-SECURITY.COM +33 (0)9 53 22 99 64

Internet of Things & Privacy concern ?

- **Any IoT object could reveal information about individuals**
- **Wearable Technology:** clothes, watches, contact lenses with sensors, microphones with cameras embedded and so on
- **Quantified Self:** pedometers, sleep monitors, and so on
- **Home Automation:** connected households using smart fridges, smart lighting and smart security systems, and so on
- ...



It is estimated that 50 billion devices will be connected by 2020



Source: Cisco
Internet of Things (IoT) Opportunities, 2015

This proliferation poses new privacy and security risks that must be assessed



Internet of Things & Privacy concern ?

- **Last news : (you can update this slide every week ☹)**



Technology

VTech hack: Parents complain of Christmas disappointment

By Kevin Rawlinson
BBC News

5 January 2016 | Technology

VTech was hacked in November, exposing millions of accounts.

In response, the firm took some essential services offline, meaning products could not be registered on Christmas Day.



Turning a Webcam Into a Backdoor

Posted by [Vectra Threat Labs](#) on Jan 12, 2016 5:00:00 AM

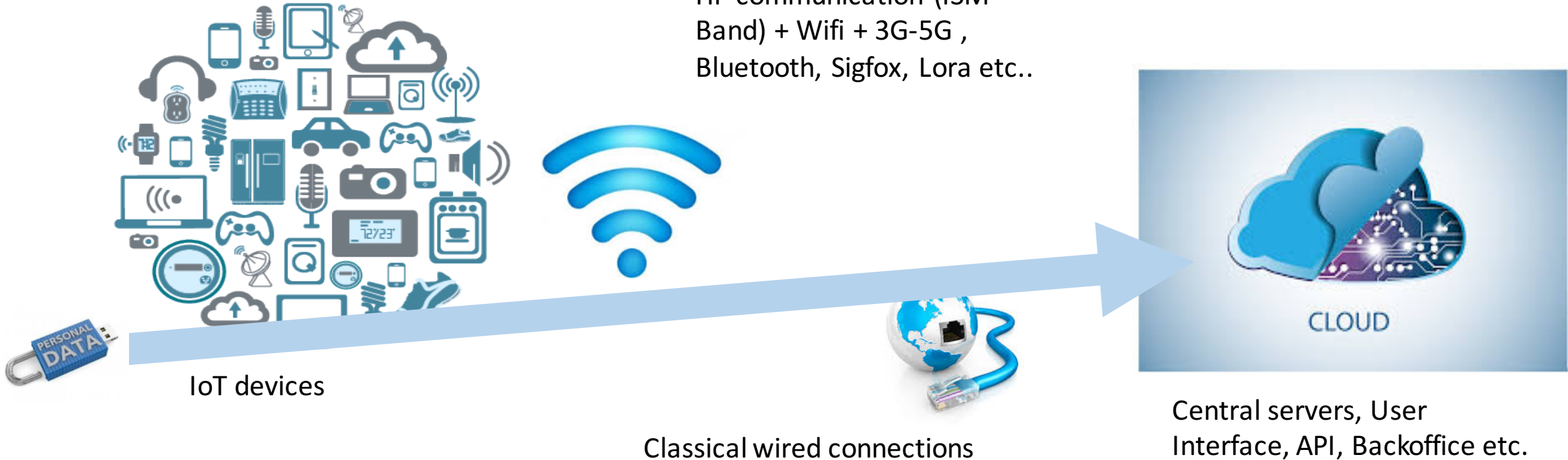
Firmware can be read without any problem (SPI memory)



IoT Eco-system (20000 feet view)

- **Privacy Risk level : Where?**

HF communication (ISM Band) + Wifi + 3G-5G ,
Bluetooth, Sigfox, Lora etc..



Security speaking, hardware is the new software ?



Direct access

SOFTWARE

To secure it:

- Security products (Firewall, Antivirus, IDS,...)
- Security services (Pentest, Audit, ...)
- Tools (Uncountable number of them)

« Bridge » access

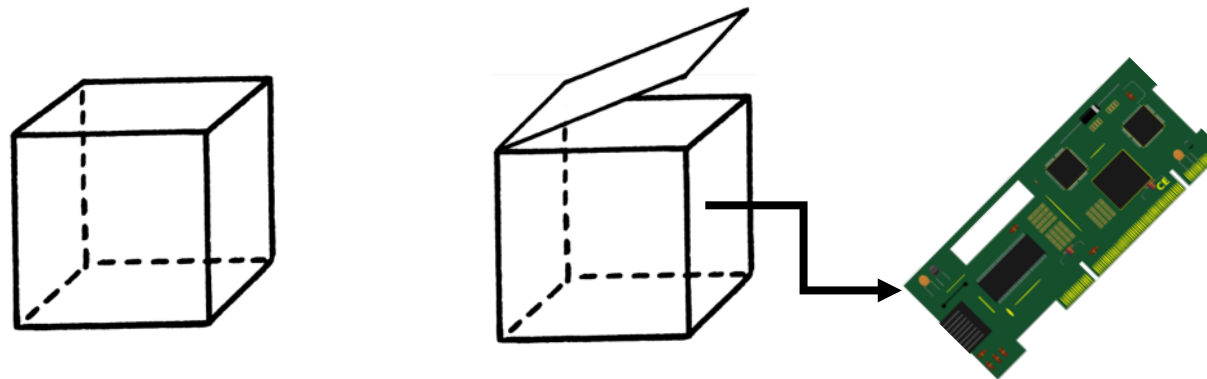
HARDWARE

To secure it:

- Few or unimplemented solutions (Encryption with key in a secure area, anti-replay mechanisms, readout protection, ...)

Hardsploit & hardware hacking basic procedure

- 1/ Open it
- **2/ Fingerprint all the component if you can else automatic brute forcing**
- 3/ Use those that may contain data (Online / Offline analysis ?)
- **4/ Perform read | write operation on them**
- 5/ Reverse engineering, find vulnerabilities and exploit them



Global Purpose



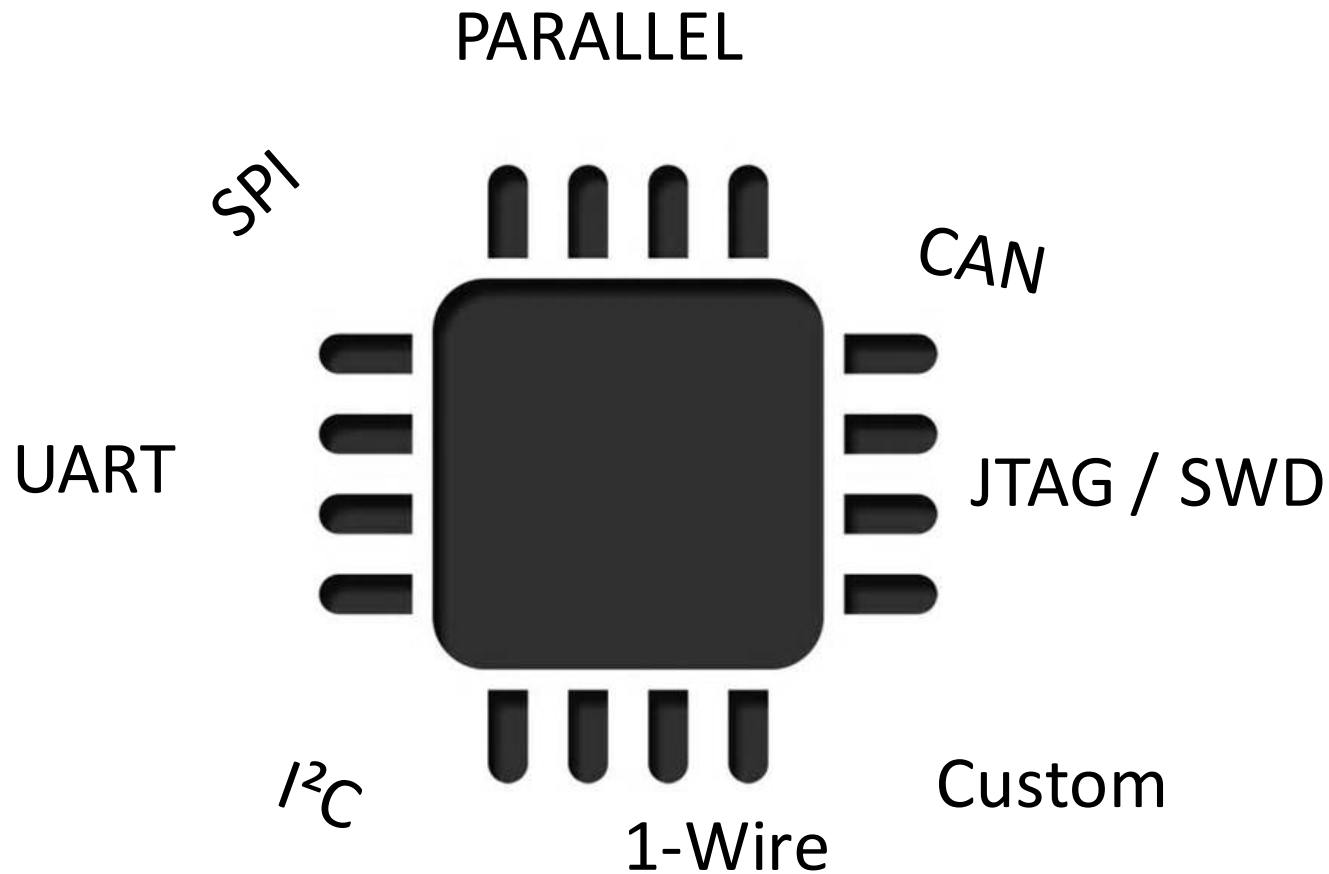
Why ?

- Because chips contain interesting / private data
 - Passwords
 - File systems
 - Firmware
 - ...

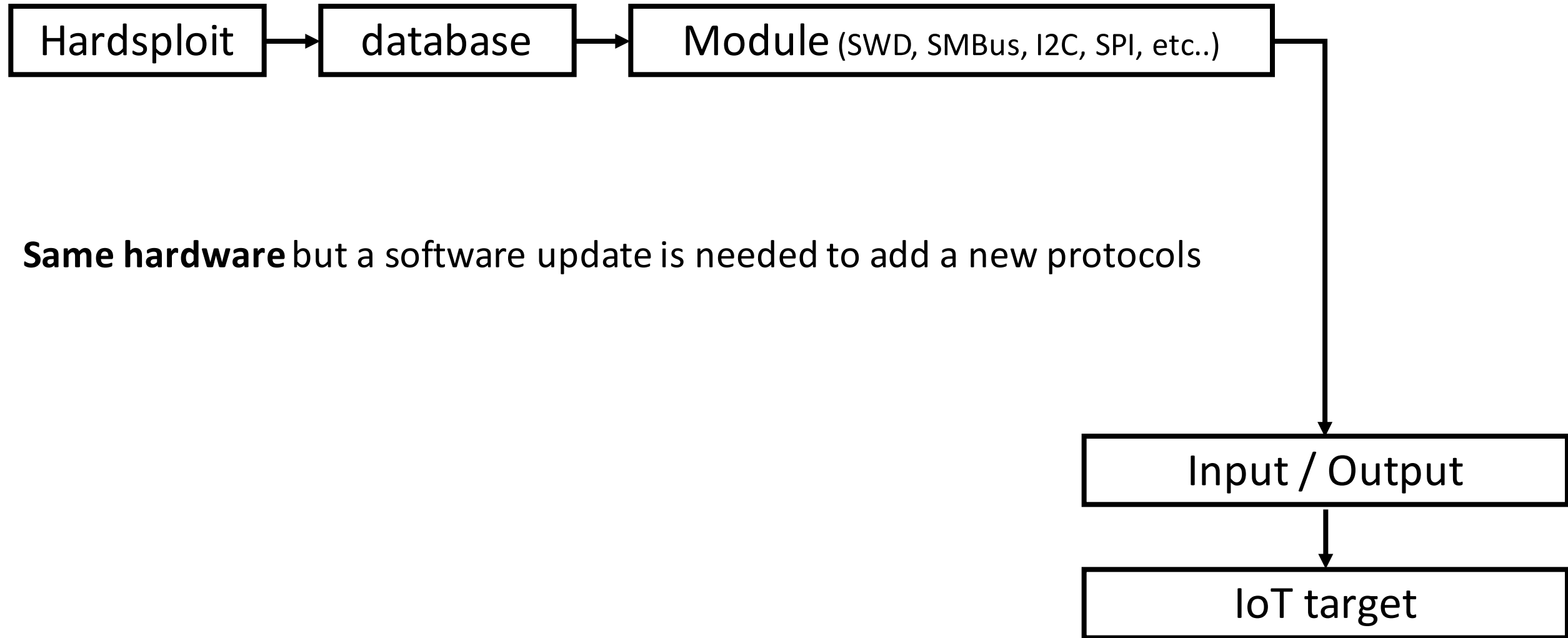
```
0000000 0000 0001 0001 1010 0010 0001 0004 0128
0000010 0000 0016 0000 0028 0000 0010 0000 0020
0000020 0000 0001 0004 0000 0000 0000 0000 0000
0000030 0000 0000 0000 0010 0000 0000 0000 0204
0000040 0004 8384 0084 c7c8 00c8 4748 0048 e8e9
0000050 00e9 6a69 0069 a8a9 00a9 2828 0028 fdfc
0000060 00fc 1819 0019 9898 0098 d9d8 00d8 5857
0000070 0057 7b7a 007a bab9 00b9 3a3c 003c 8888
0000080 8888 8888 8888 8888 288e be88 8888 8888
0000090 3b83 5788 8888 8888 7667 778e 8828 8888
00000a0 d61f 7abd 8818 8888 467c 585f 8814 8188
00000b0 8b06 e8f7 88aa 8388 8b3b 88f3 88bd e988
00000c0 8a18 880c e841 c988 b328 6871 688e 958b
00000d0 a948 5862 5884 7e81 3788 1ab4 5a84 3eec
00000e0 3d86 dcb8 5cbb 8888 8888 8888 8888 8888
00000f0 8888 8888 8888 8888 8888 8888 8888 0000
0000100 0000 0000 0000 0000 0000 0000 0000 0000
*
0000130 0000 0000 0000 0000 0000 0000 0000
000013e
```

How ?

- A hardware pentester need to know electronic buses and he need to be able to interact with them



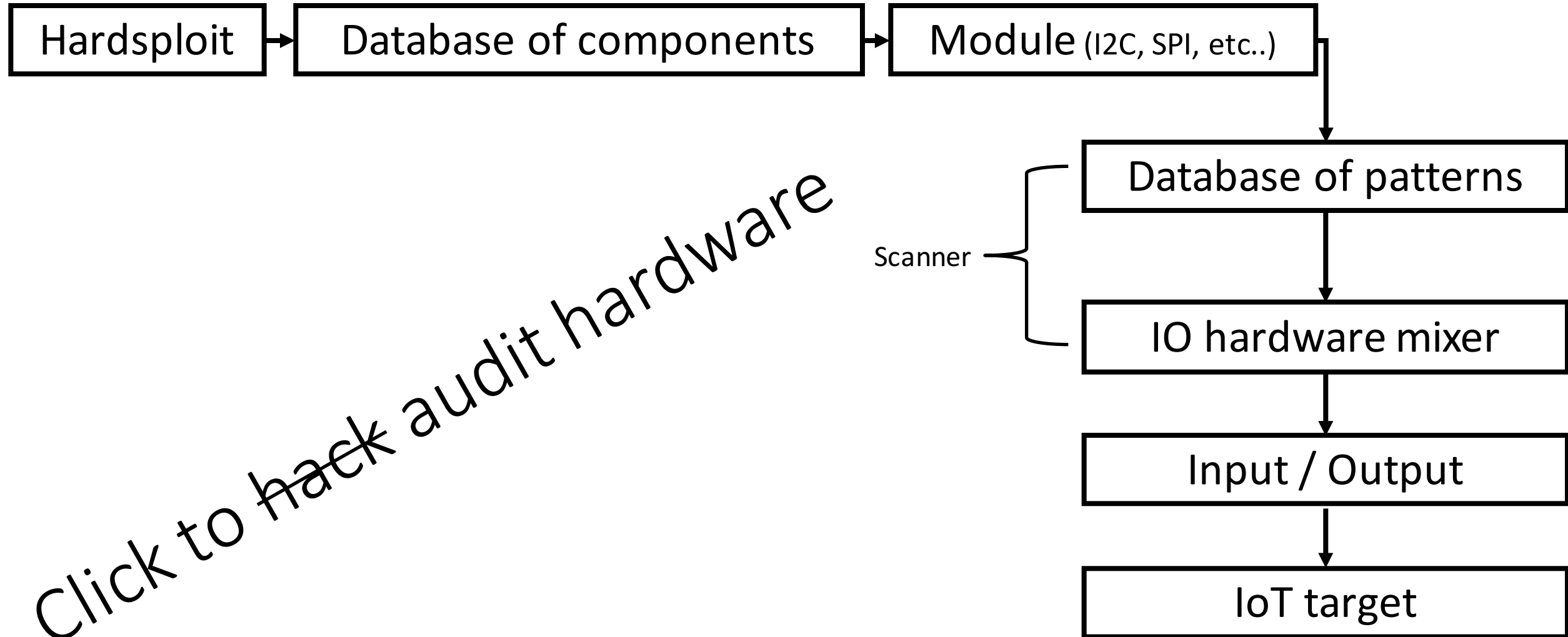
Hardsploit framework



Same hardware but a software update is needed to add a new protocols

Hardsploit bus indention & scanner

(in progress, not published yet)

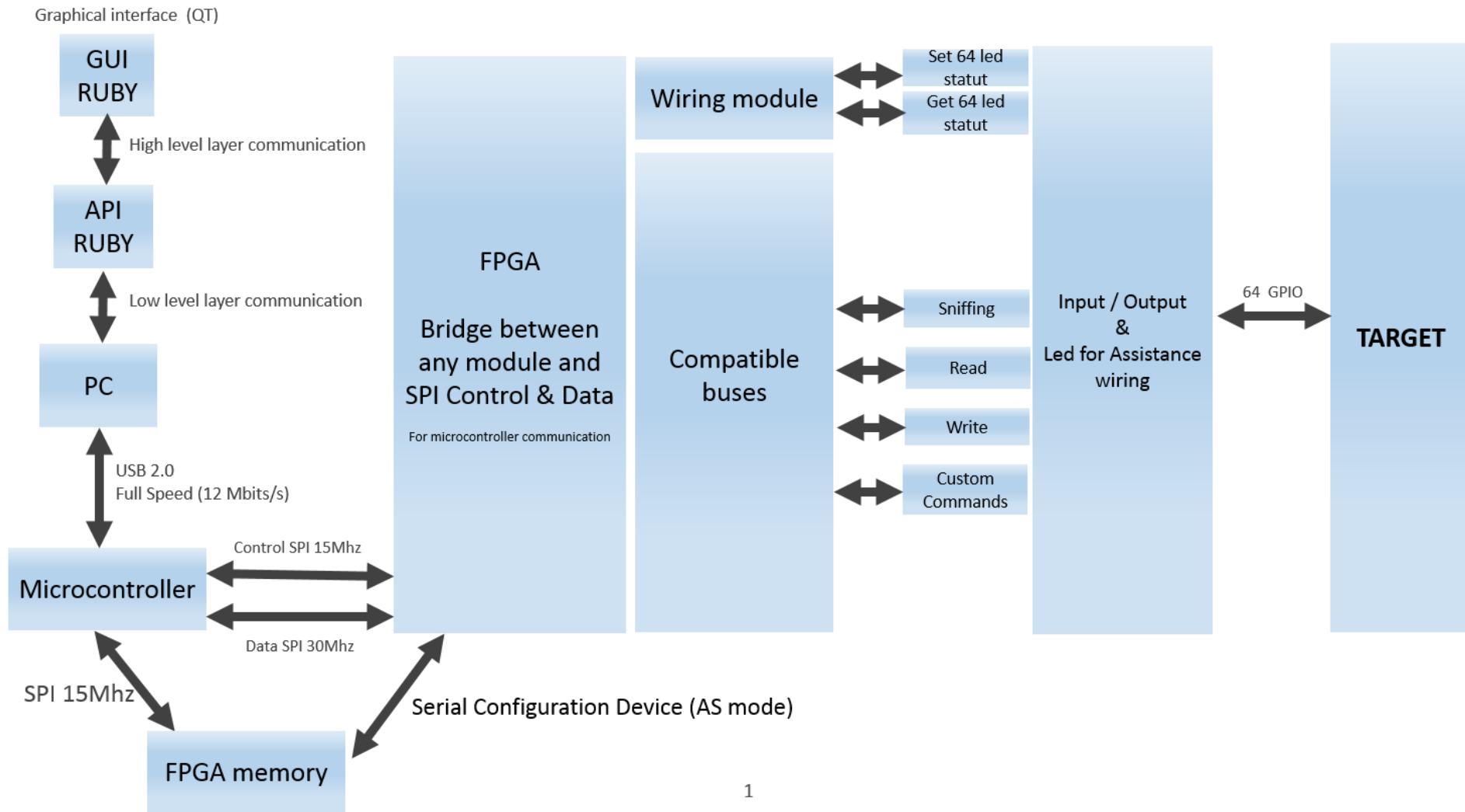


Click to hack audit hardware

Tool of trade

FUNCTIONALITIES	BUSPIRATE	JTAGULATOR	GOODFET	HARDSPLOIT
UART	○	Bus identification	✗	○
SPI	○	✗	○	○
PARALLEL	✗	✗	✗	○
I2C	○	✗	✗	○
JTAG / SWD	○	Bus identification	○	○
MODULARITY	Microcontroller	Microcontroller	Microcontroller	uC / FPGA
EASE OF USE	Cmd line + datasheet	Command line	Command line	Official GUI / API / DB
I/O NUMBER	< 10	24	< 14	64 (plus power)
WIRING	TEXT (but MOSI = SDA ☺)	TEXT / AUTOMATIC identification	TEXT	LED / TEXT / AUTOMATIC identification

Hardsploit: Communication



Prototype making

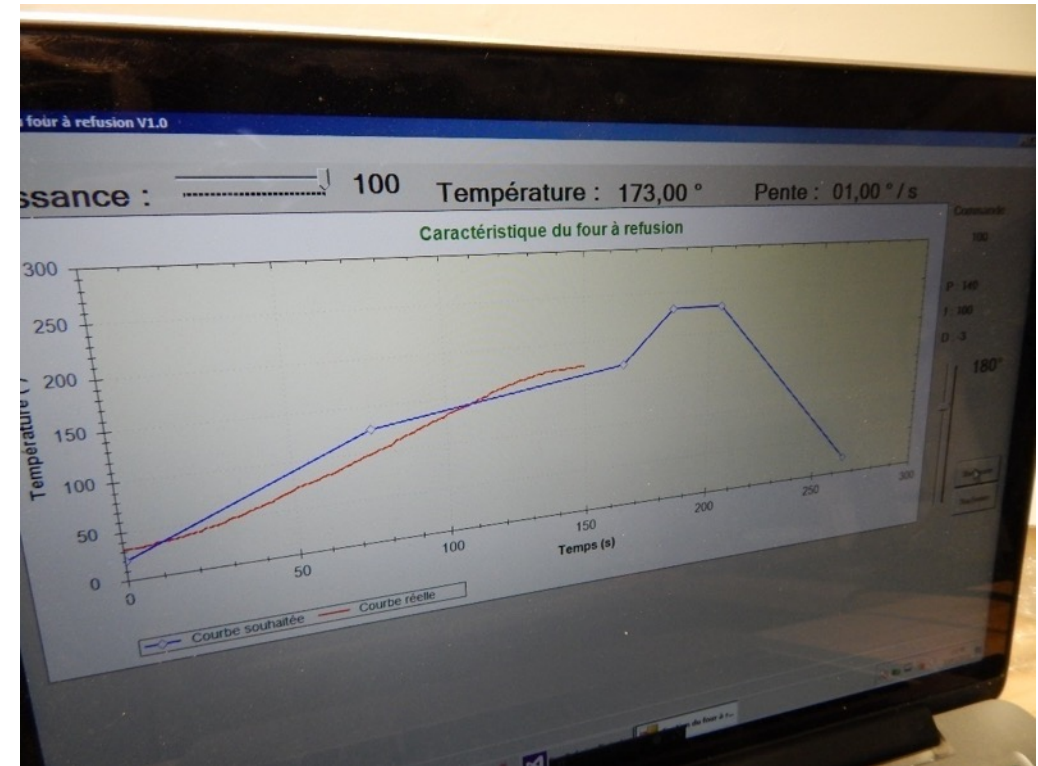
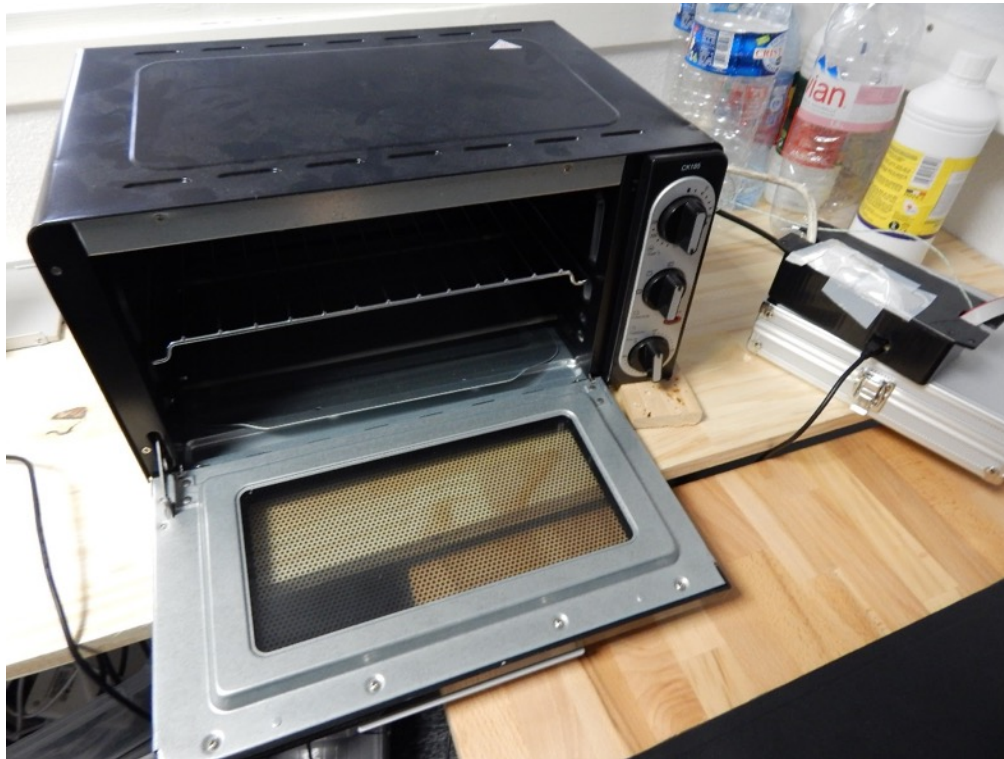
- Applying soldering paste (low budget style)



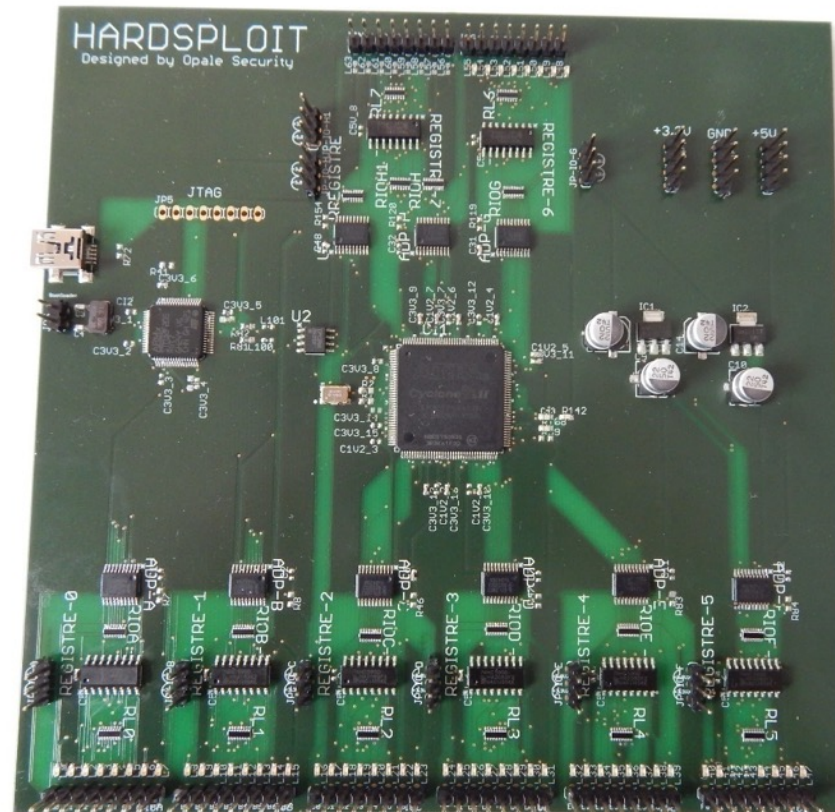
16/03/2016

Prototype making

- Manual reflow oven (DIY style)

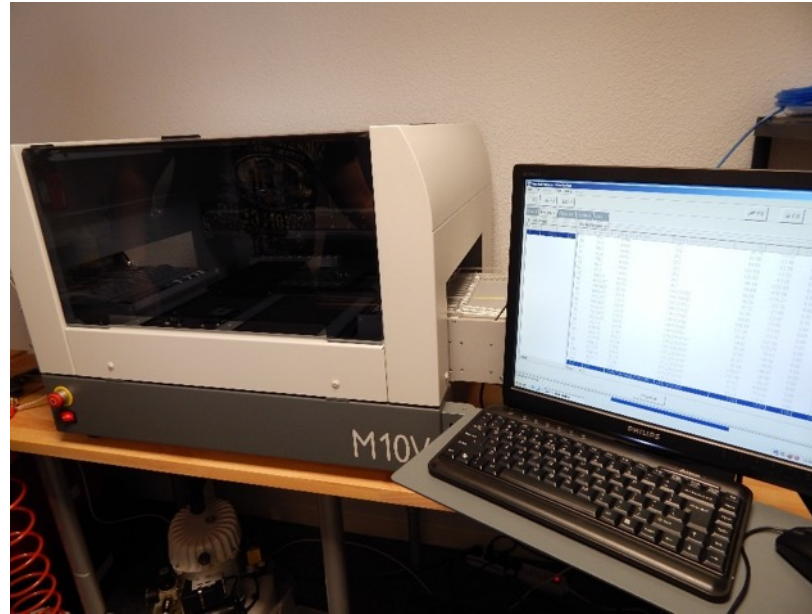
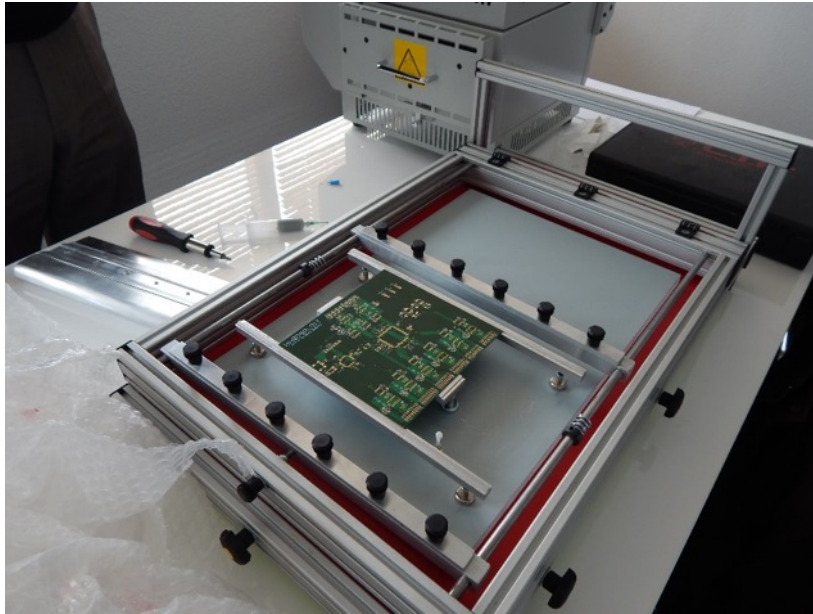


Prototype V0.1 aka The Green Goblin 😊



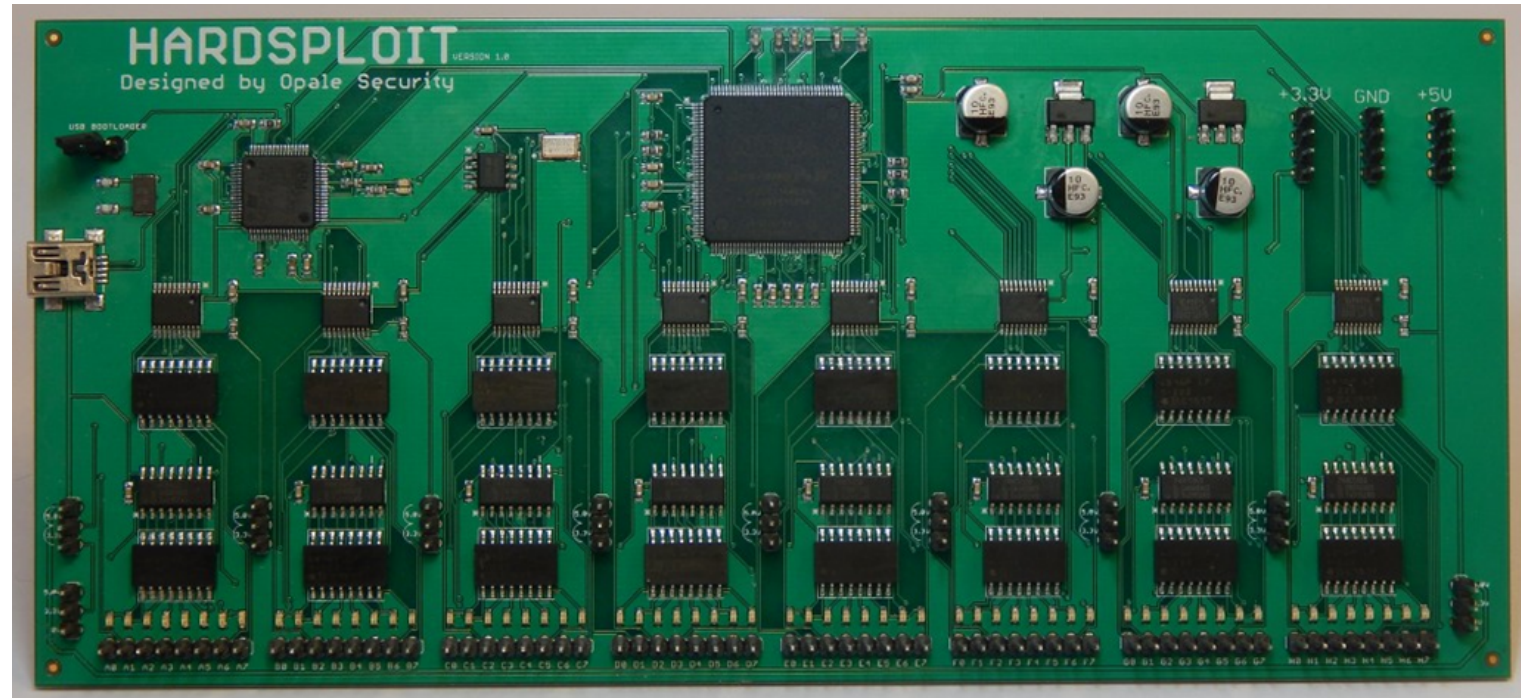
Prototype making (with a budget)

- The rebirth

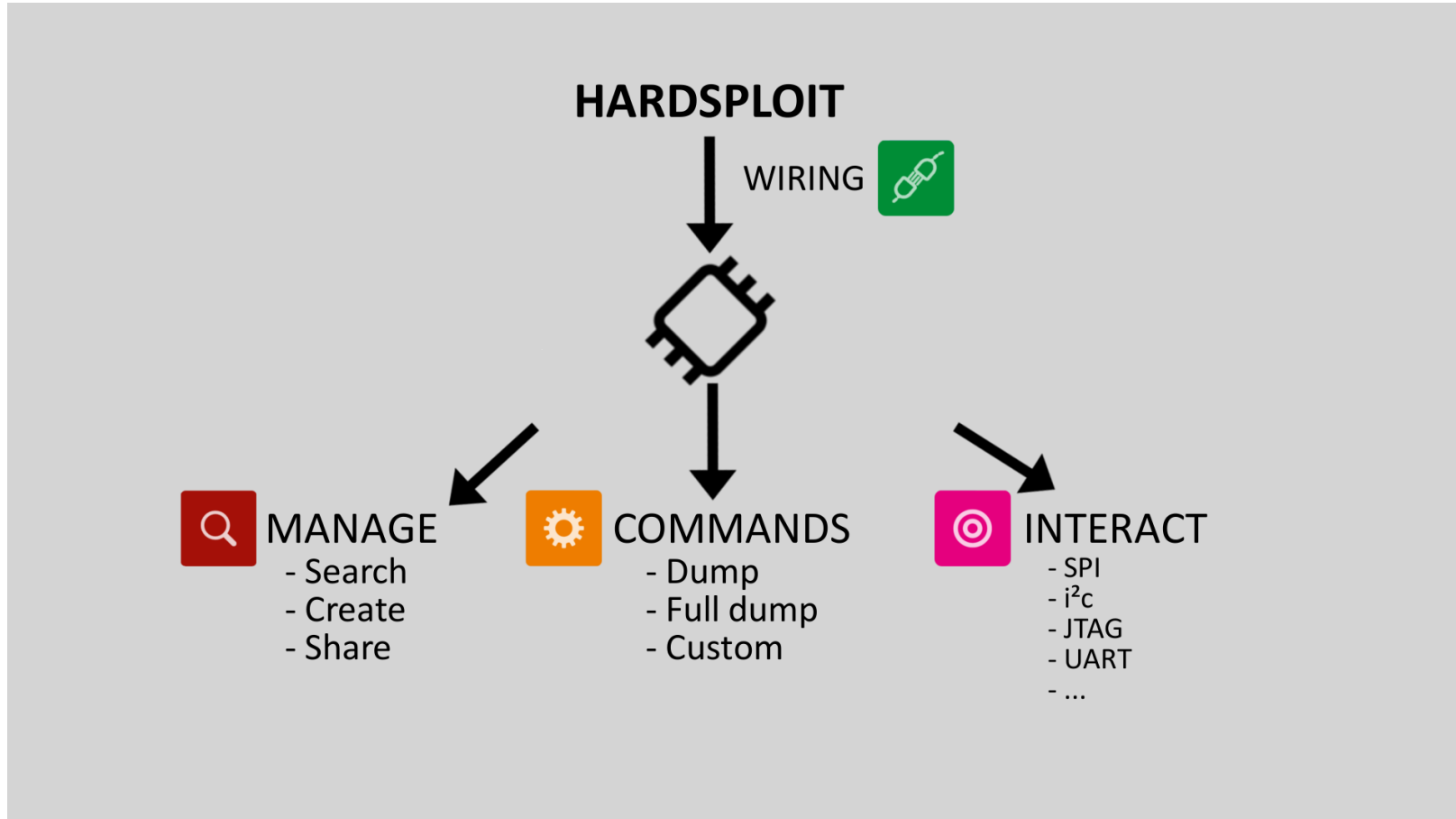


The board – Final version

- 64 I/O channels
- ESD Protection
- Target voltage: 3.3 & 5V
- Use a Cyclone II FPGA
- USB 2.0
- 20cm x 9cm



Hardsploit organization



Chip management

- Search
- Create
- Modify
- Interact

Hardsploit - Chip management

Menu SWD About

Current chip: Manufacturer... Type...

Reference	Type	Manufacturer	BUS
1 P33-65nm	MEMORY	Numonyx	PARALLEL
2 25LC640	MEMORY	MICROCHIP	SPI
3 24LC64	MEMORY	MICROCHIP	I2C
4 M25P40	MEMORY	Micron	SPI
5 SST39VF802C-70-4I-EKE	MEMORY	MICROCHIP	PARALLEL
6 AS6C4008-55TIN	MEMORY	ALLIANCE MEMORY	PARALLEL

Double click a chip reference to load it Create component

Console:

Date / Time	Message
1 21/12 14:51	Hardsploit board detected GUI V2.0 beta API V1.0.6 BOARD : HW:V1.00 SW:V1.0.2
2 21/12 14:51	Hardsploit ready to suck chip souls !

Hardsploit - Chip editor

Name / Reference:

Description:

Voltage: 3,3V 5V

Manufacturer:

Type:

Package:

Not in the list ? Create a new one

Package name:

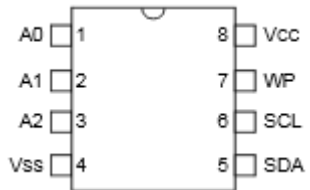
Package pin number:

Package shape: Square Rectangular

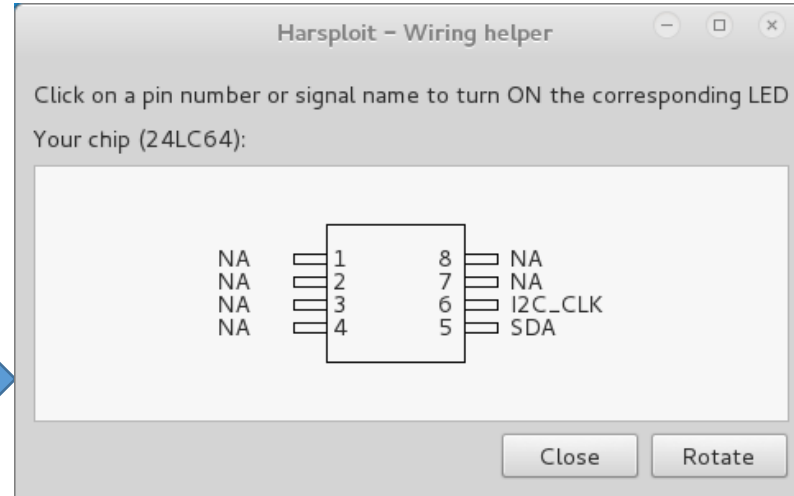
Pin Number	Bus	Signal
1	NA	NA
2	NA	NA
3	NA	NA
4	NA	NA
5	I2C	SDA
6	I2C	I2C_CLK
7	NA	NA

To complete this form, please report to the component datasheet.

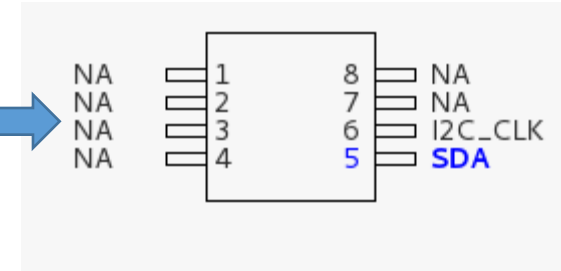
Wiring helper



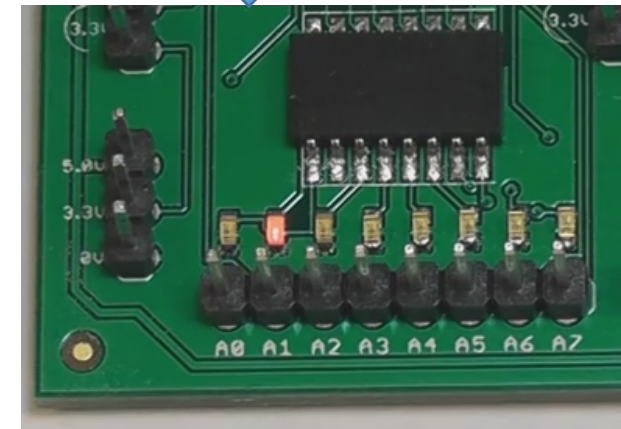
Datasheet representation



Harsploit Wiring module representation



GUI <--> Board interaction



Settings

Hardsploit - I²C settings

24LC64 PARAMETERS

Base address (W):

Base address (R):

Frequency (Khz):

Total size:

Bus scan:

Address	R/W
---------	-----

Hardsploit - Bus settings

25LC640 PARAMETERS

Page size: Total size (8 bits word):

Frequency (Mhz): Mode:

SPI command read:

Hardsploit - Parallel settin...

P33-65nm PARAMETERS

Total size:

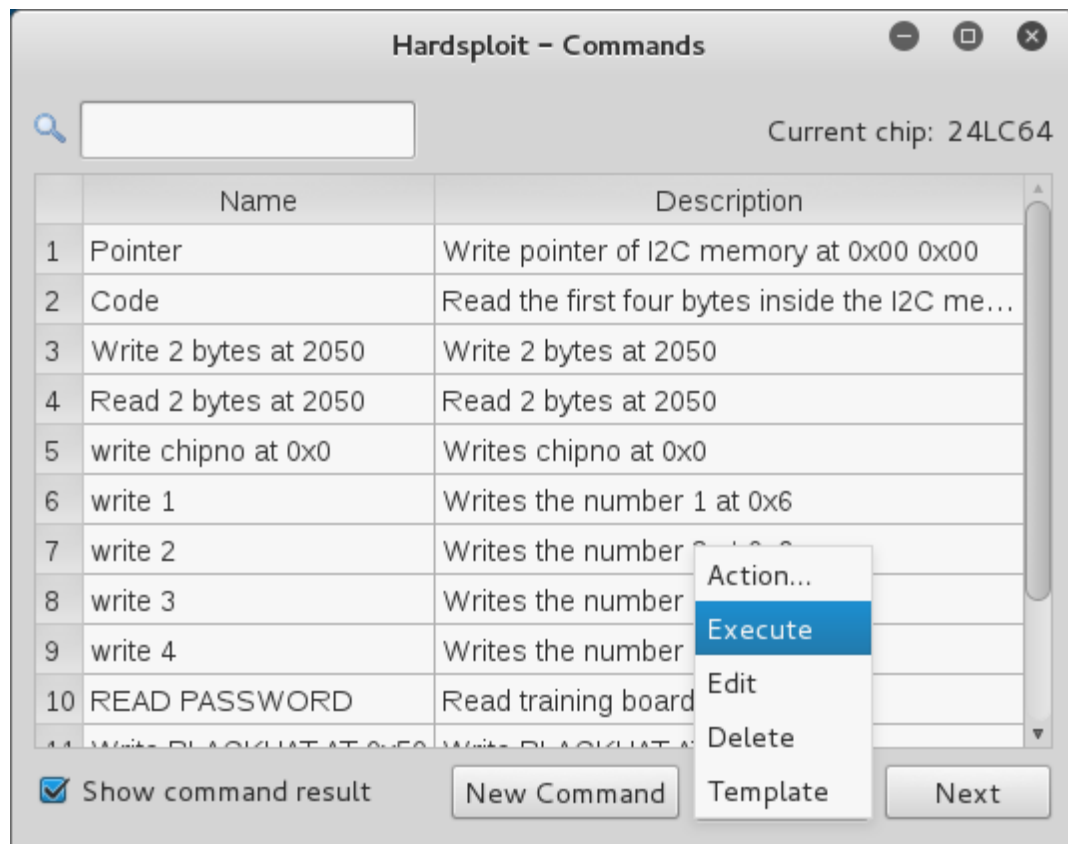
Read latency:

Write latency:

Word size: 8 bits 16 bits

Page size:

Command editor



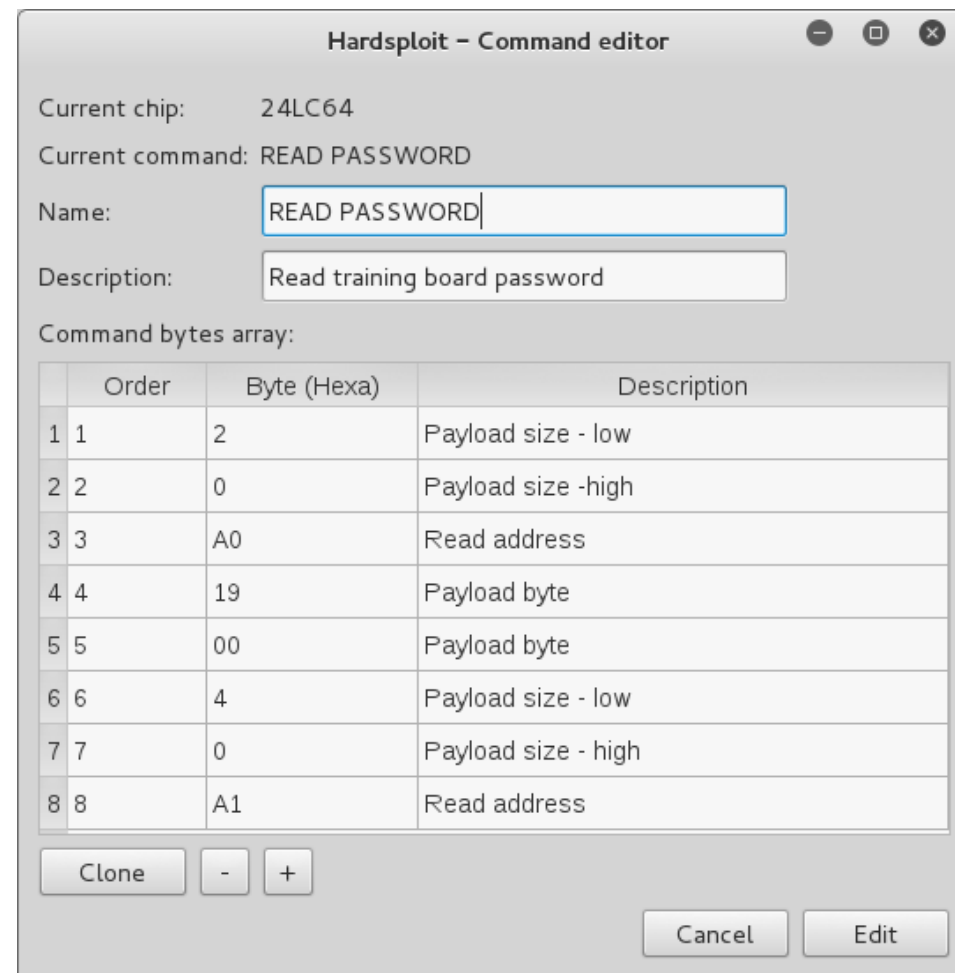
Hardsplit - Commands

Current chip: 24LC64

	Name	Description
1	Pointer	Write pointer of I2C memory at 0x00 0x00
2	Code	Read the first four bytes inside the I2C me...
3	Write 2 bytes at 2050	Write 2 bytes at 2050
4	Read 2 bytes at 2050	Read 2 bytes at 2050
5	write chipno at 0x0	Writes chipno at 0x0
6	write 1	Writes the number 1 at 0x6
7	write 2	Writes the number 2 at 0x7
8	write 3	Writes the number 3 at 0x8
9	write 4	Writes the number 4 at 0x9
10	READ PASSWORD	Read training board password
11	Write BLACKHAT AT 0x50	Write BLACKHAT AT 0x50

Show command result

New Command Template Next



Hardsplit - Command editor

Current chip: 24LC64

Current command: READ PASSWORD

Name: READ PASSWORD

Description: Read training board password

Command bytes array:

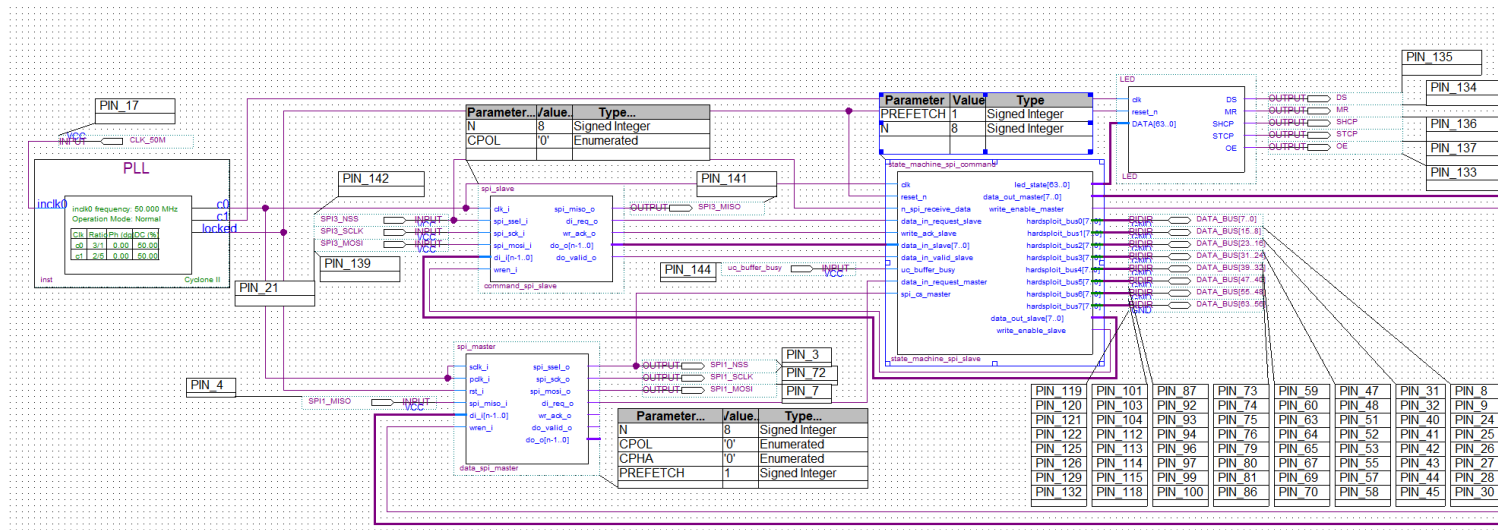
	Order	Byte (Hexa)	Description
1	1	2	Payload size - low
2	2	0	Payload size -high
3	3	A0	Read address
4	4	19	Payload byte
5	5	00	Payload byte
6	6	4	Payload size - low
7	7	0	Payload size - high
8	8	A1	Read address

Clone - +

Cancel Edit

What are available on github (Open) ?

- Microcontroller (c)
- API (ruby)
- GUI (ruby)
- Create your own Hardsploit module : VHDL & API (ruby)



Already available (github)

Parallel non multiplexed memory dump

- 32 bits for address
- 8/16 bits for data

Helping wiring

I2C 100Khz 400Khz and 1 Mhz

- Addresses scan
- Read, write, automatic full and partial dump

SPI mode 0,1,2,3 up to 25 Mhz

- Read, write, automatic full and partial dump

SWD interface (like JTAG but for ARM core)

- Dump and write firmware of most ARM CPU

GPIO interact / bitbanging (API only for the moment)

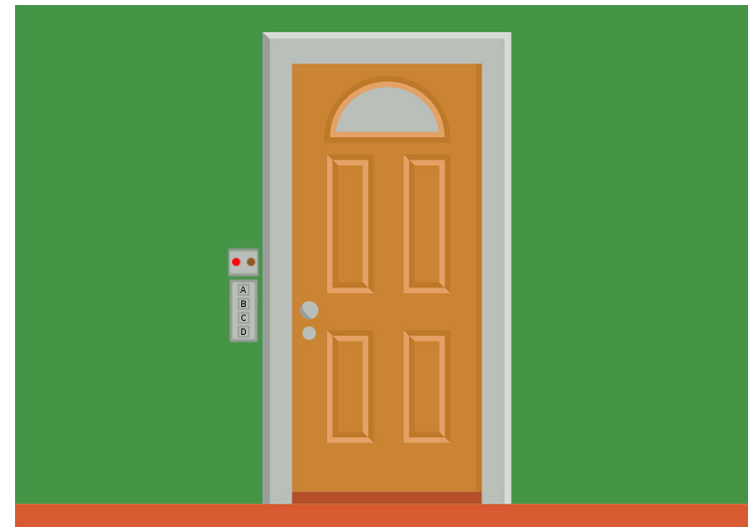
- Low speed < 500Hz read & write operations on 64 bits

More to come (see online roadmap)...

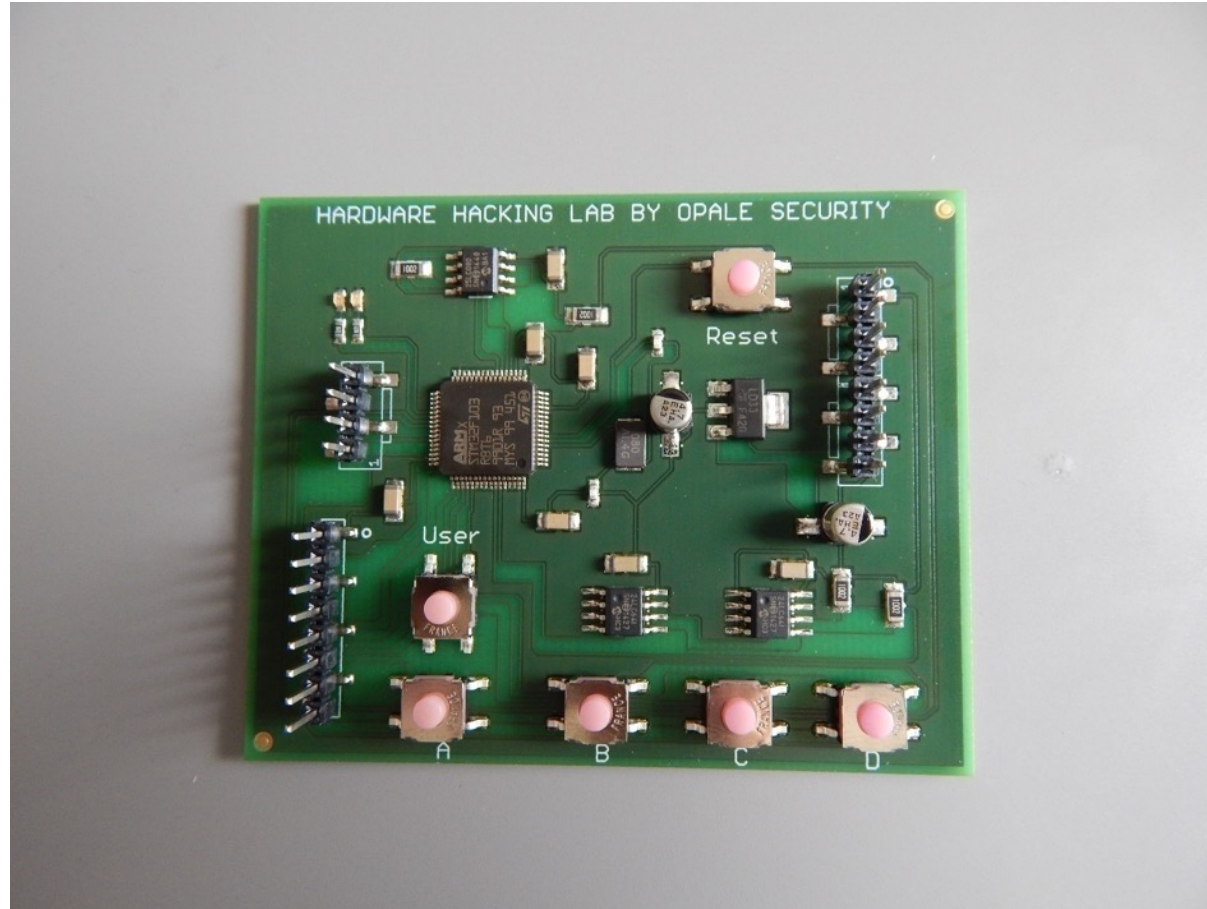
- Automatic bus identification & Scanner (@30%)
- Component & commands sharing platform (@90%)
- TTL UART Module with automatic detection speed (@80%)
- Parallel communication with multiplexed memory
- I2C sniffing (shot of 4000 bytes up to 1 Mhz)
- SPI sniffing (shot of 8000 / 4000 byte half / full up to 25Mhz)
- RF Wireless transmission training platform (Nordic NRF24, 433Mhz, 868Mhz transceivers)
- Metasploit integration (module) ??
- JTAG
- 1 Wire
- CanBUS (with hardware level adapter)
- ...

Concrete case

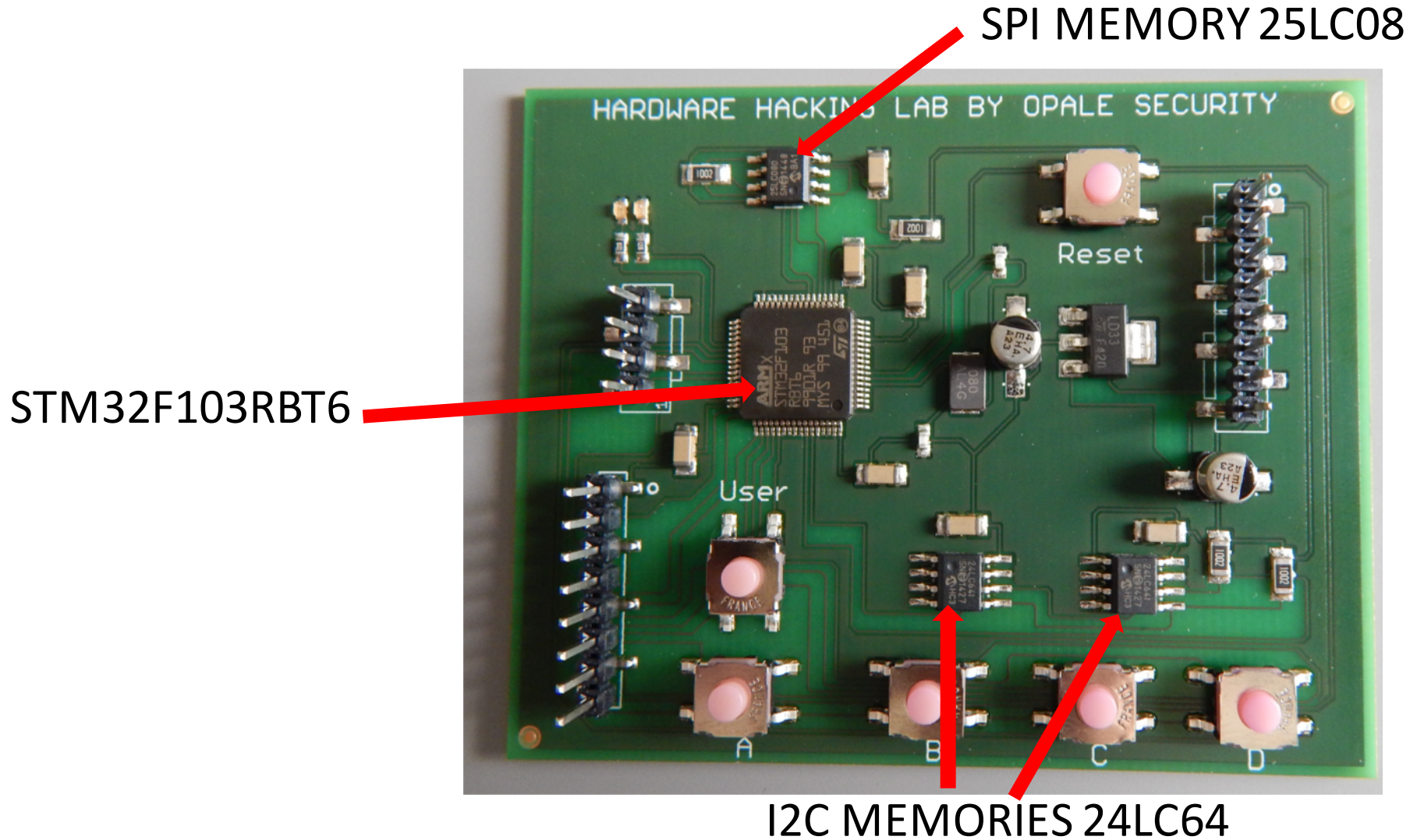
- An electronic lock system
- 4 characters pin code A – B – C – D
 - Good combinaison – Door opens, green L.E.D turn on
 - Wrong combinaison – Door closes, red L.E.D turn on



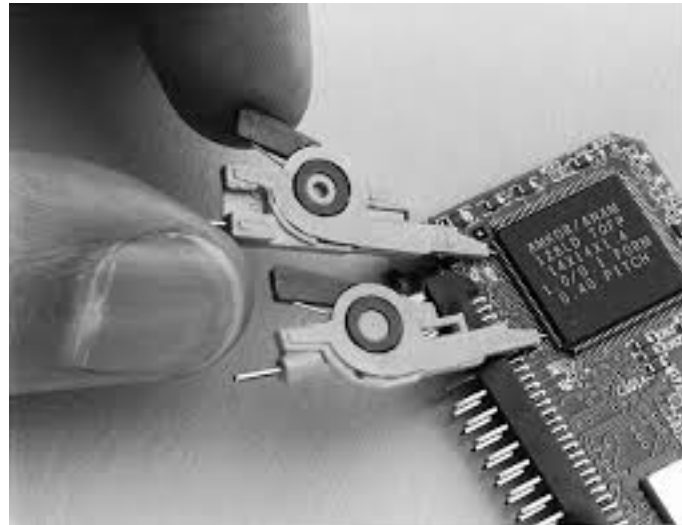
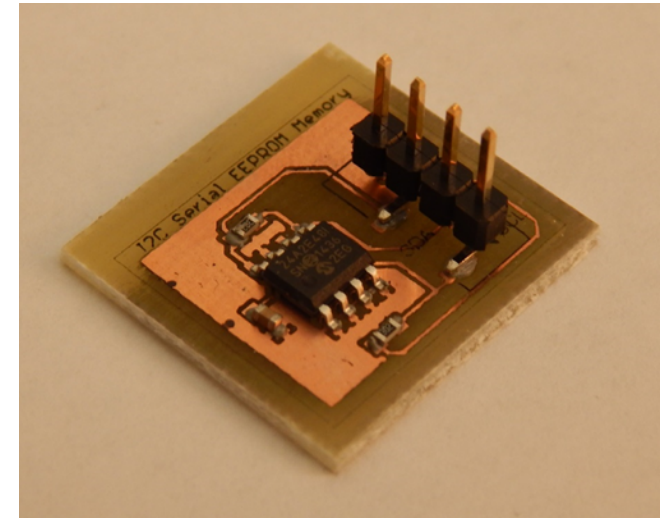
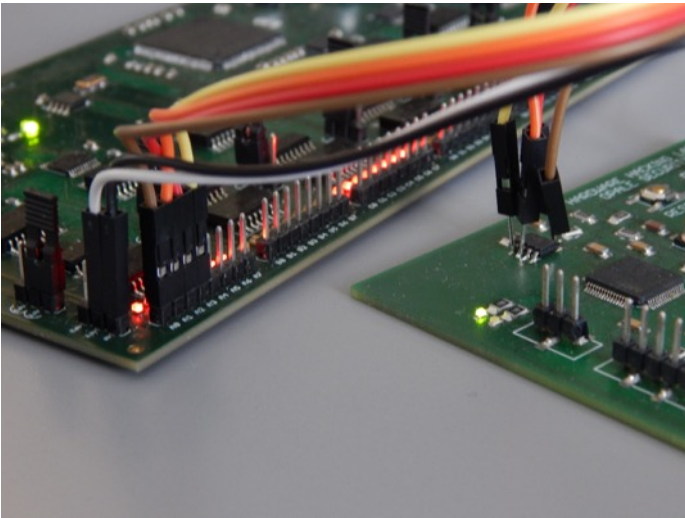
Concrete case: Open it



Concrete case: Fingerprint



Concrete case: Online / Offline analysis ?



Concrete case: hardsploit scenario

1. Open Hardsploit to create the component (if not exist)
2. Connect the component to Hardsploit (wiring helping)
3. Enter and save the component settings (if not exist)
4. Dump the content of the memories (1 click)
5. Change the door password by using commands (few clicks)
6. Try the new password on the lock system (enjoy)

Concrete case:

Read | Write operation, I2C, SPI, SWD ...

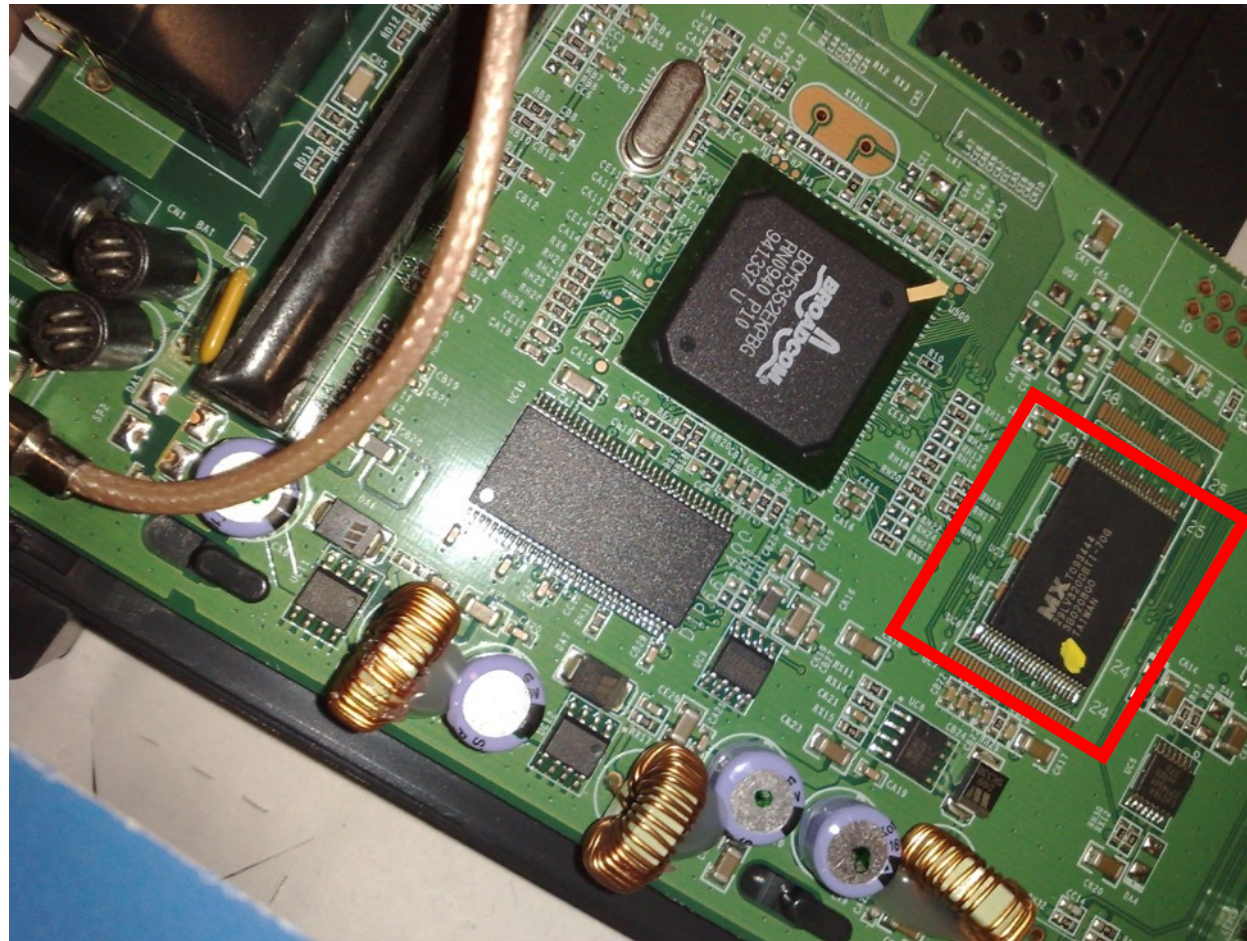


- Time for a live demo ?

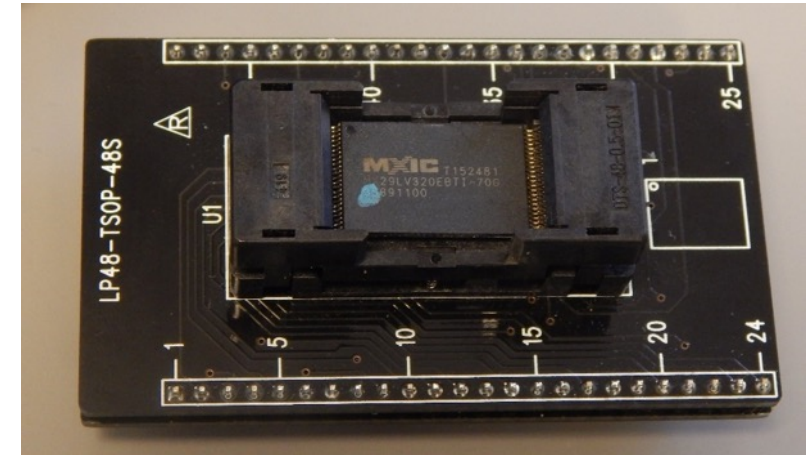
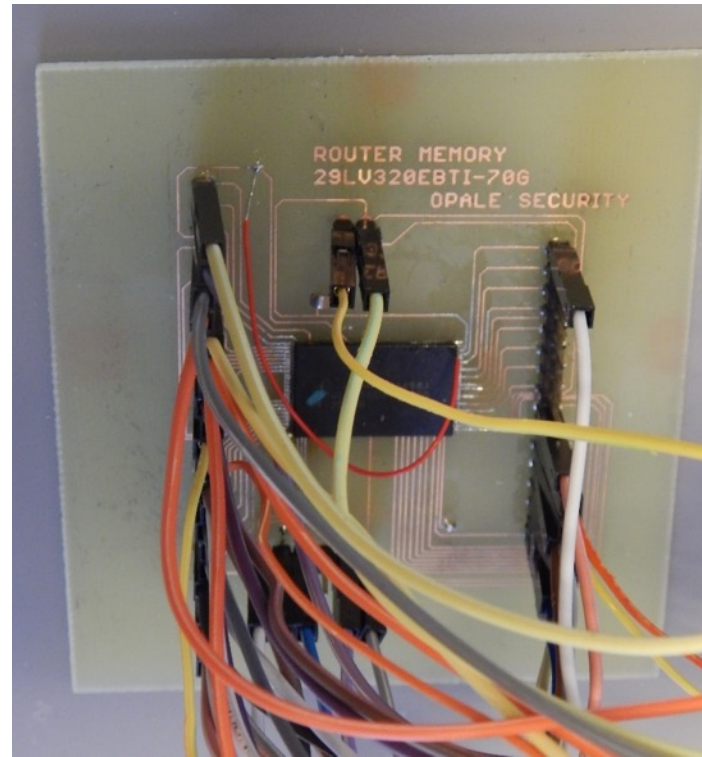
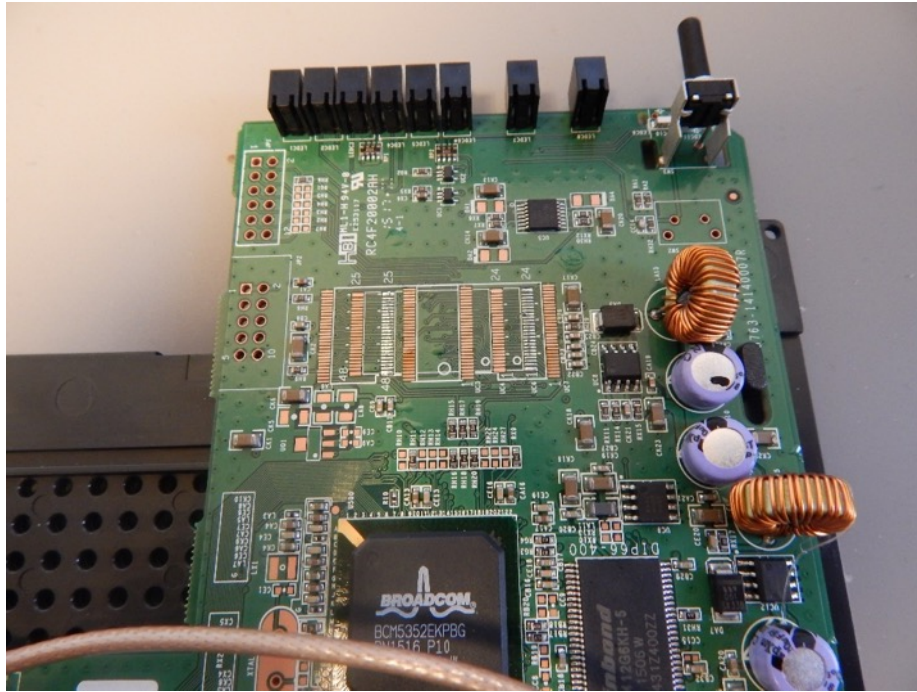
Parallel bus memory



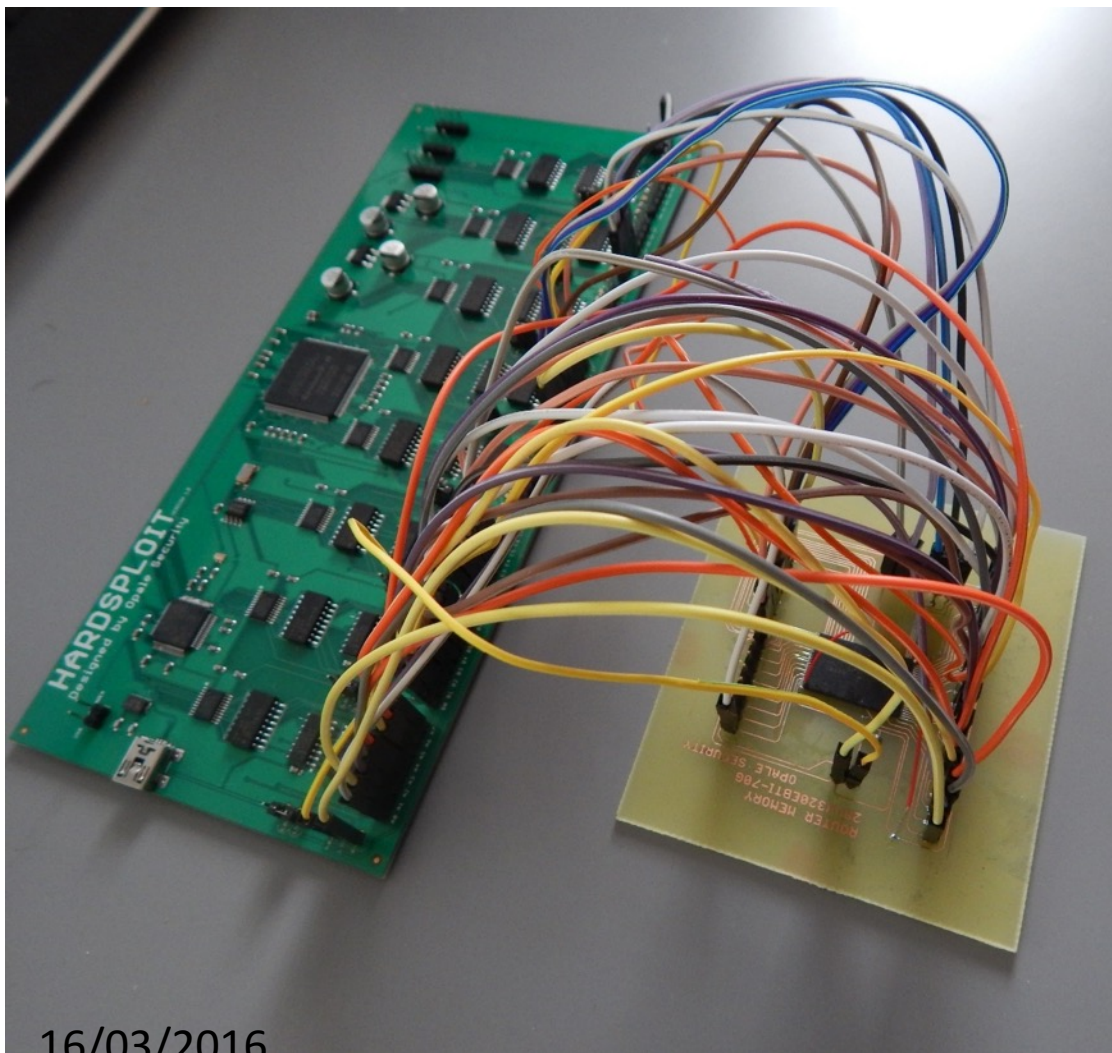
Concrete case: Fingerprint



Concrete case: Offline analysis



Concrete case: Ready to dump the content



```
/root/Bureau/FirmwareRouter.bin - Bless
File Edit View Search Tools Help
FirmwareRouter.bin x
00000000 | 17 08 A4 2C 12 F1 85 19 41 77 85 19 DB 7B 85 19 48 44 D5 | .....Aw...{..HD.
00000013 | E6 88 78 85 19 D8 B2 85 19 76 DD 85 19 E8 FF E9 87 4D 3E | ...x.....v.....M>
00000026 | 85 19 0C 14 85 19 25 28 85 19 CB E9 A4 34 5C 4C 85 19 E0 | .....%(....4\L...
00000039 | 36 85 19 8A DC 85 19 0E 00 C2 DF 35 BE 85 19 9A 88 85 19 | 6.....5.....
0000004c | 3B 28 85 19 31 4C 1B E4 0A 55 85 19 2D 86 85 19 59 30 85 | ;(..lL...U.....Y0.
0000005f | 19 04 00 F9 70 21 CD 85 19 75 30 85 19 AE 0A 85 19 E2 64 | ...p!...u0.....d
00000072 | 85 19 5E DC 85 19 97 CC 85 19 D3 4D FF FF 69 6E CD 52 65 | ..^.....M..in.Re
00000085 | 2C FF FF 26 4B FF FF 3E 32 FF FF E3 D8 03 BF 91 E0 FF FF | ...&K..>2.....
00000098 | 66 0E FF FF A3 29 FF FF 10 00 E1 37 02 1B FF FF 72 7A FF | f.....).....7.....rz.
000000ab | FF E3 D7 FF FF 25 1C 00 00 2B FF FF 0A 41 FF FF 87 69 | .....%.....+.....A...i
000000be | FF FF 21 A0 05 35 0E 2A FF FF 76 48 FF FF D6 95 FF FF C8 | ...!...5.*...vH.....
000000d1 | B8 D6 04 93 07 FF FF 9C 96 FF FF 2F 7E FF FF 76 65 ED 3C | ...../...~...ve.<
000000e4 | B9 7C FF FF 24 72 FF FF 9D 86 FF FF 94 D7 FF FF E4 B6 FF | .|.$.r.....
000000f7 | FF B8 5A FF FF F8 5E 46 4C 61 64 34 95 20 E1 FF FF ED 17 | ..Z...^FLad4. ....
0000010a | FF FF C4 A4 FF FF 6B D6 11 47 B9 61 FF FF 4A 8A FF FF BA | .....k...G.a...J....
0000011d | FF FF FF 00 00 53 84 28 2E FF FF 7E FF FF FF E7 C6 FF FF | .....S.(...~.....
00000130 | 10 60 12 DC F0 83 FF FF 1B F9 FF FF BD 54 FF FF 21 38 83 | ^.....T...!8.
00000143 | D8 87 FC FF FF 19 8F FF FF 94 AC FF FF 68 74 ED D6 02 0E | .....ht....
00000156 | FF FF AE 08 FF FF 31 1A FF FF 58 20 E2 E1 77 37 FF FF 0A | .....1...X ...w7...
00000169 | 60 FF FF 2C 83 FF FF 37 5E FF FF D3 6F FF FF 4C 8D FF FF | ^.....7^...o...L...
0000017c | 75 96 FF FF 03 00 3D FA 9E D3 FF FF D3 45 FF FF F9 6C FF | u.....=.....E...l.
0000018f | FF 5C 3D 3B 6C 7C BD FF FF 2D 18 FF FF 1A 01 FF FF 00 00 | .\|=;|.....
000001a2 | 45 57 A1 8A FF FF 85 64 FF FF 75 02 FF FF 2F 1E 00 00 86 | EW.....d...u.../....
000001b5 | 3A FF FF 9D C9 FF FF BE 19 FF FF 08 00 A0 AB 21 6F FF FF | :.....f.....Y.....o.
000001c8 | 15 95 FF FF C5 66 FF FF C2 A5 F8 59 ED A3 FF FF 84 6F FF | .....f.....Y.....o.
000001db | FF 74 6D FF FF 00 08 34 7C 58 3B FF FF 63 BC FF FF B2 08 | .tm....4|X;..c....
000001ee | FF FF EA 7C FF FF 0D D5 FF FF 06 6B FF FF 76 19 FF FF 70 | ...|.....k...v...p
00000201 | 9E 9D 0D DB 50 FF FF 30 F8 FF FF 9B 97 FF FF 24 23 B4 E6 | ....P..0.....$#.
Signed 8 bit: 23 Signed 32 bit: 386442284 Hexadecimal: 17 08 A4 2C
Unsigned 8 bit: 23 Unsigned 32 bit: 386442284 Decimal: 023 008 164 044
Signed 16 bit: 5896 Float 32 bit: 4,415118E-25 Octal: 027 010 244 054
Unsigned 16 bit: 5896 Float 64 bit: 1,03014042409718E-197 Binary: 00010111 00001000 10
Show little endian decoding Show unsigned as hexadecimal ASCII Text: [hex]?,
Offset: 0x0 / 0x3d0900 Selection: None INSg
```


Conclusion

- IoT Device are (also) prone to vulnerabilities help you to find them
- Security policy need to be adpated, nowadays, it is not so difficult to extract data on IoT
- Designers need to design with security in mind
- Skills related to pentest a hardware device is mandatory for Security Experts (but training exist)
- Industry need to take care about device security



Thank you !

Hardsplit board is available at shop-hardsplit.com (250 € / 277 USD / 370 CAD excluding VAT)

To learn more about Hardsplit and follow the development

Hardsplit.io & Opale-Security.com

- Yann ALLAIN (CEO)
- yann.allain@opale-security.com
- +33 6 45 45 33 81
- Julien MOINARD (Project leader of Hardsplit)
- julien.moinard@opale-security.com
- +33 9 72 43 87 07

Hardware & Software, Pentest, Audit, Training