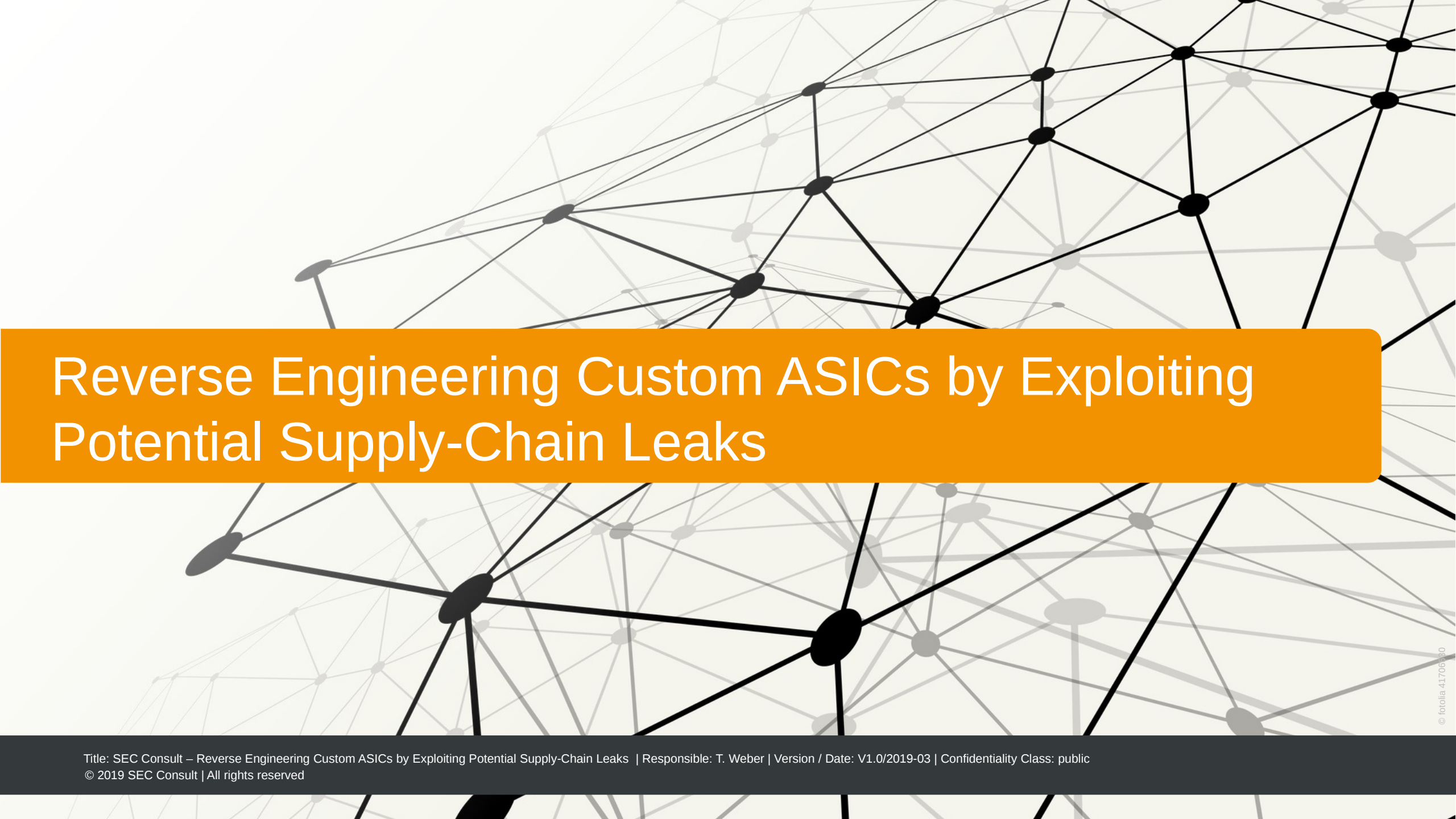
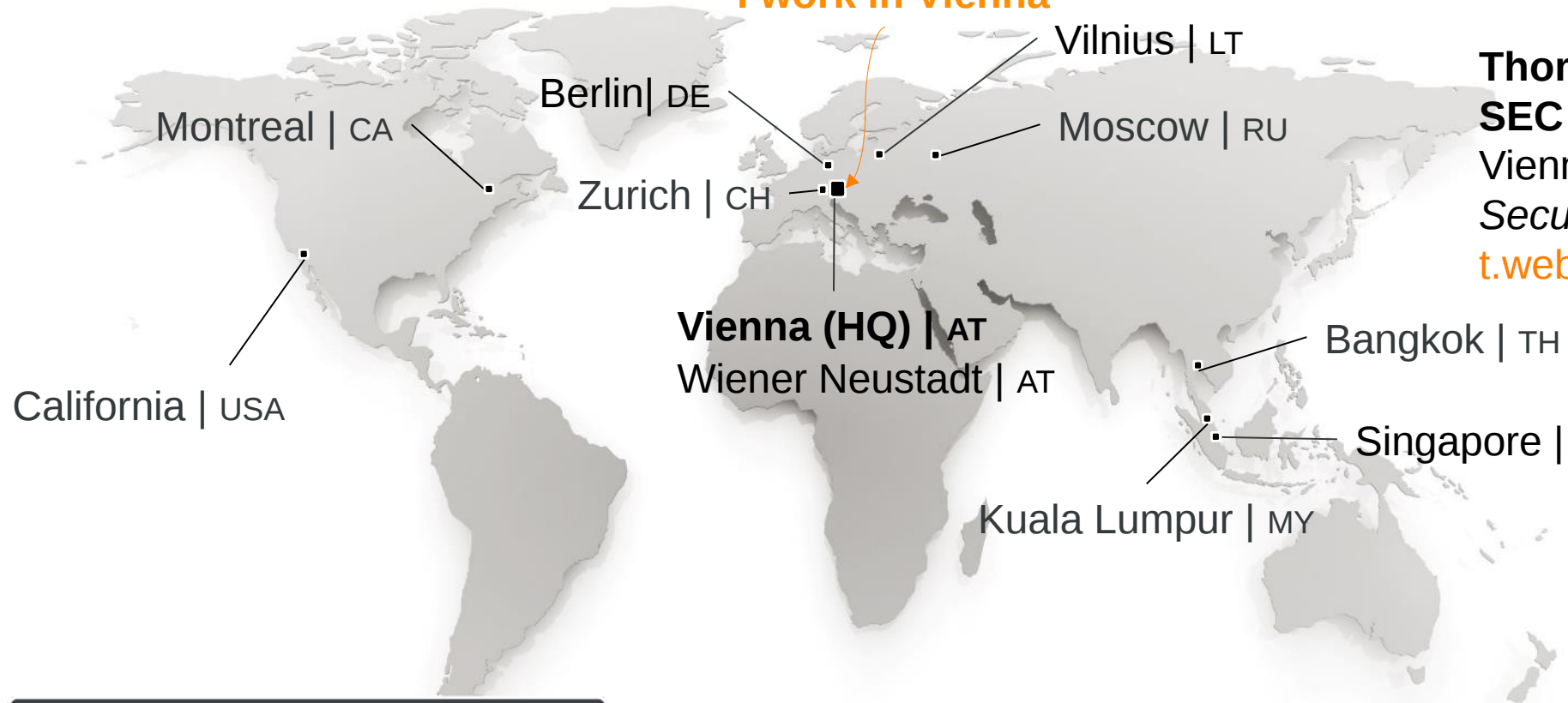


Reverse Engineering Custom ASICs by Exploiting Potential Supply-Chain Leaks

$ whoami



Thomas Weber
SEC Consult Group
Vienna, Austria
Security Researcher & Consultant
t.weber@sec-consult.com

SEC Consult

My employer since 2015

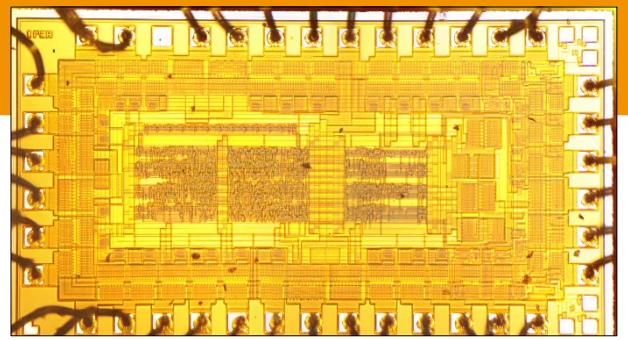
black hat ASIA 2019

Outline

At a glance:

- Introduction & motivation – important notes
- Dangers of supply-chains
- Reverse engineering methods
  - Deductive reasoning - probing methods
  - Deeper insights
- Live debugging & demo
- Fun fact
- Conclusion

Introduction



What is an ASIC?

→ Application Specific Integrated Circuit

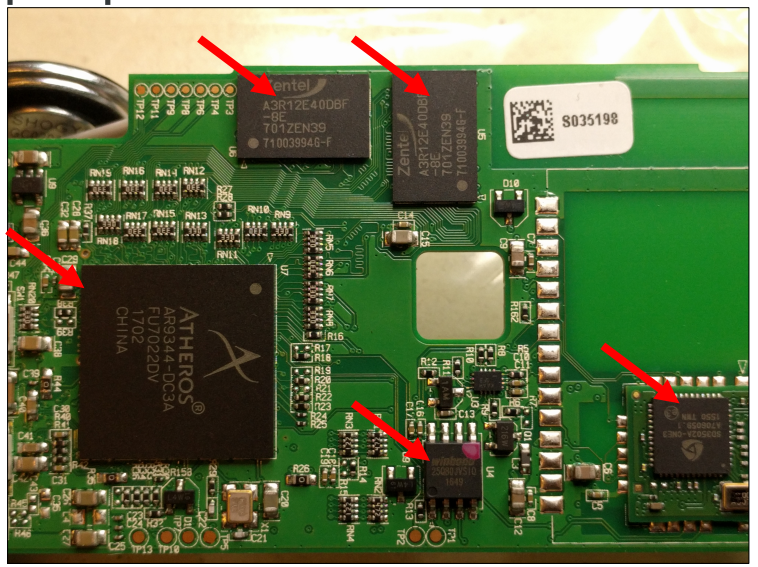
→ Can also be a System on Chip solution with customized peripherals (theoretically everything)

Who cares?

→ Vendors, security researchers, blackhat hackers...

Where is it used?

→ In every (embedded) computer system. There are more precise names for the specific applications like SoC, ASIP, NoC and so on.



black hat ASIA 2019

Introduction

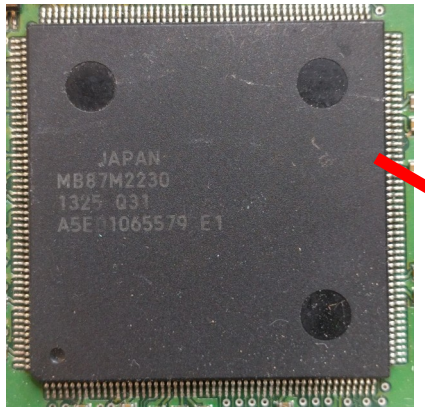
It gets hard when there are **complete custom** chips without public documentation.

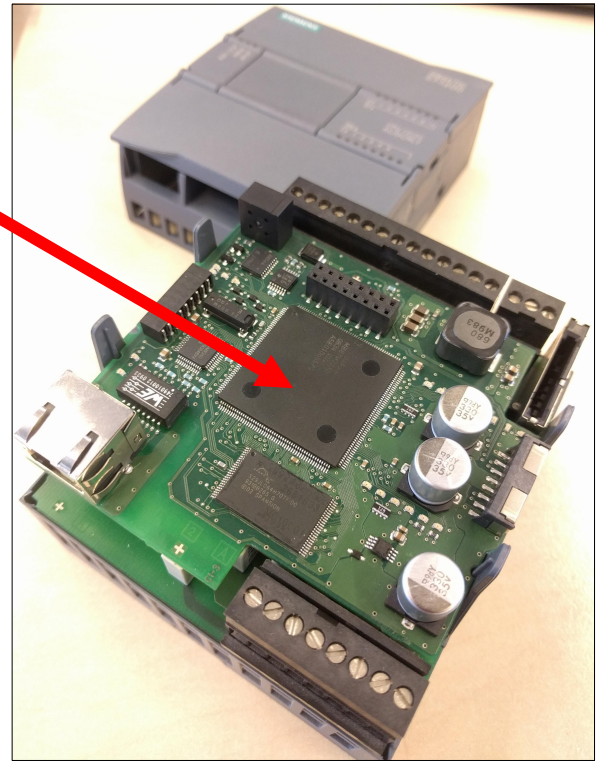
This means:

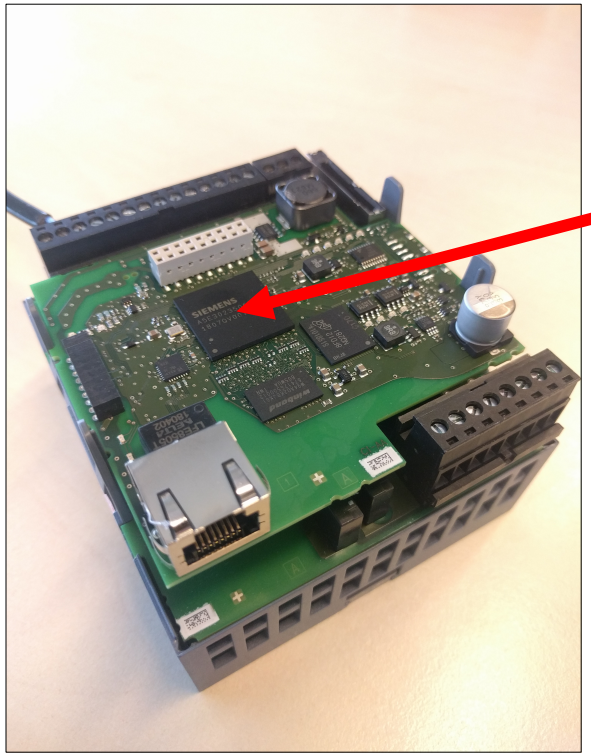
→ Architecture is **unknown**
→ Pinout is **unknown**
→ I/O memory map is **unknown**
→ Additional constraints are **unknown**
→ Sometimes, even the vendor is **unknown**

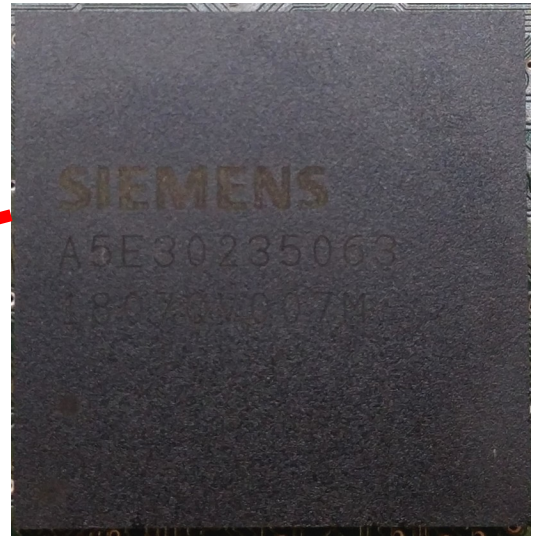
Motivation

A textbook example for custom ASICs can be found inside of industrial products like the PLC series S7-1200. There are even different hardware versions of this PLC series, and two different main chips. **Can we identify the JTAG port?**





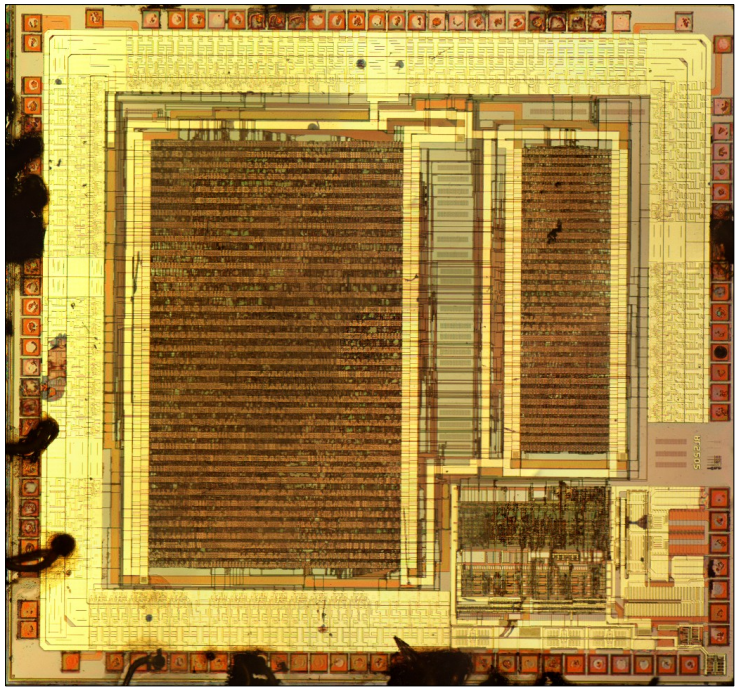




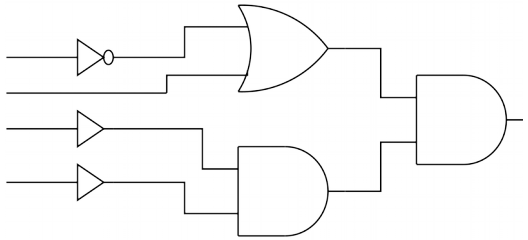
Motivation

Expensive option: Decapping, FIB or SEM and delayering, recover the hardware.

From silicon die… *… to hardware description.*



Source: https://www.capovani.com



Motivation

Cheap option: Search for similar hardware with the same chip on the internet.
Good sources are: strange online shops, eBay, AliExpress and Taobao (淘宝网)
Multiple PCBs with the same chip are even better to reverse engineer each pin functionality. The possibility to identify debug ports by having multiple different PCBs with the same chip is higher.

Bad:

Not all secrets of the hardware can be revealed in that way.

Good:

No need for super expensive equipment!

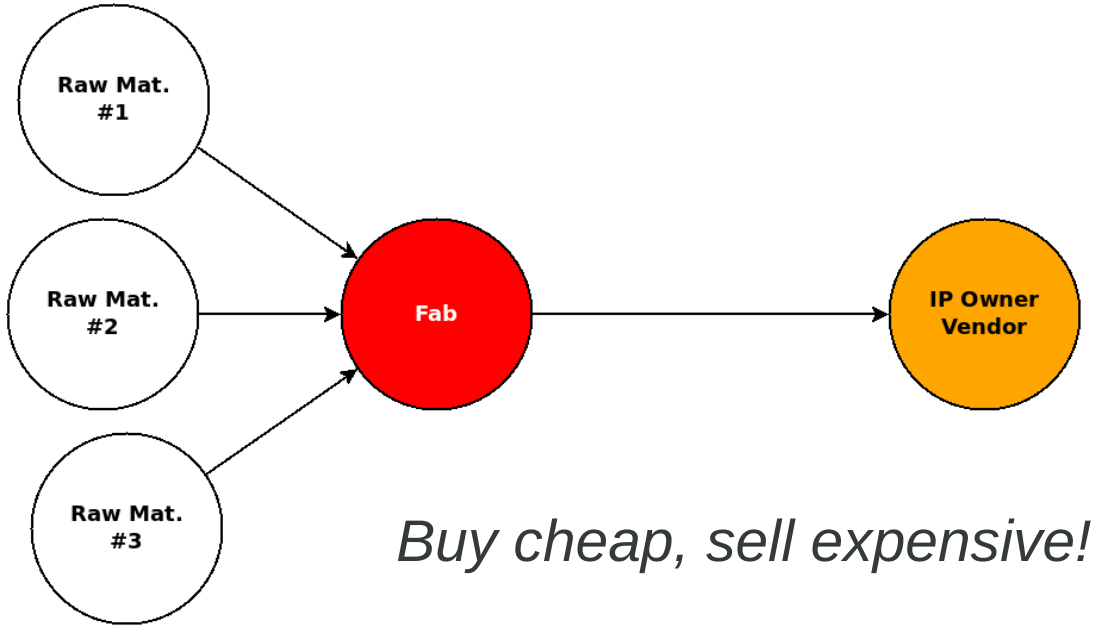
black hat ASIA 2019

Dangers of supply-chains

Supply-chains often involve exposure of IP(Intellectual Property) to 3rd parties.

Example: Super fancy chairs

I want to produce cheap fancy chairs in another part of the world to save costs. For that, I have to send the blueprints to the factory, which delivers the chairs → The intellectual property is exposed. For more complex chairs, I also need prototypes.



Buy cheap, sell expensive!

Dangers of supply-chains

Producing electronics is similar to the latter example but more complicated.

Example: PLC

Memory Chips

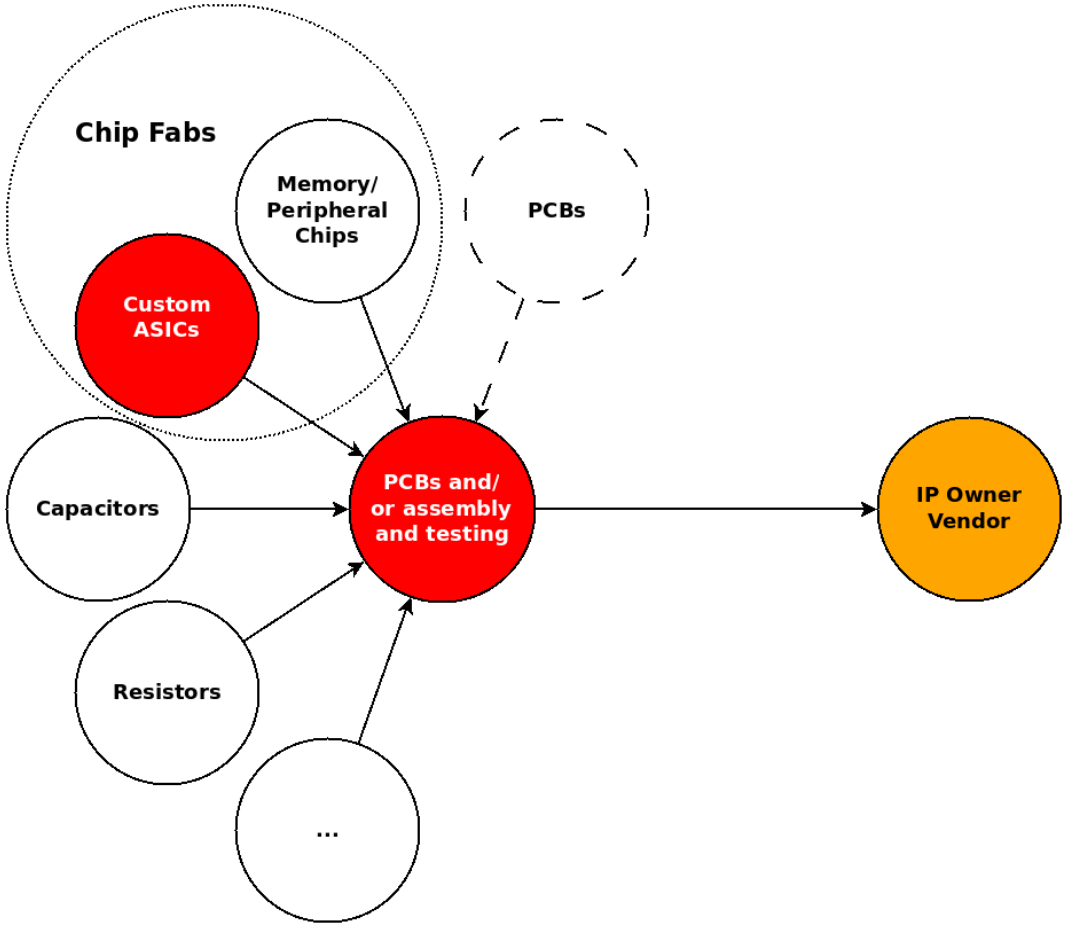
Custom Chips

Capacitors

Resistors

PCB and testing

and so on...



Dangers of supply-chains

Producing electronics is similar to the latter example but more complicated.

Example: PLC

Memory Chips

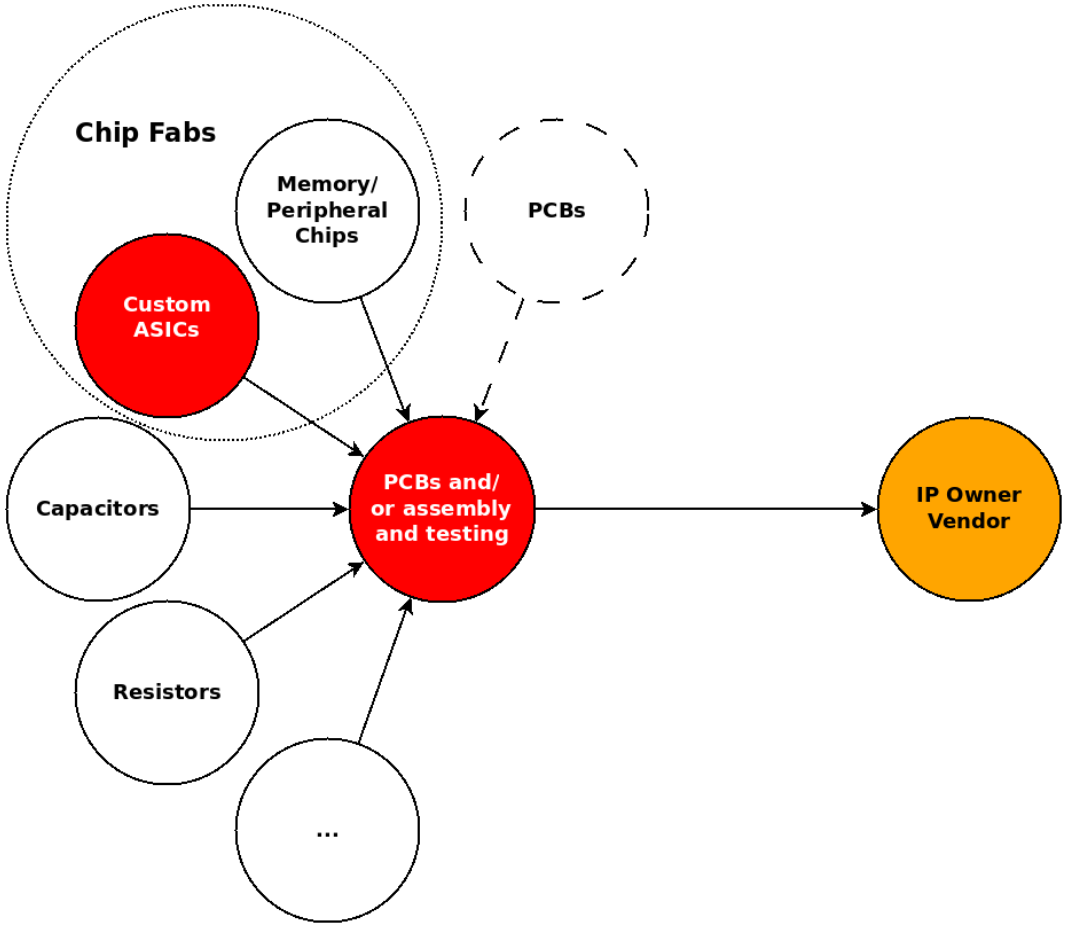
Custom Chips → IP

Capacitors

Resistors

PCB and testing → IP

and so on...



Dangers of supply-chains

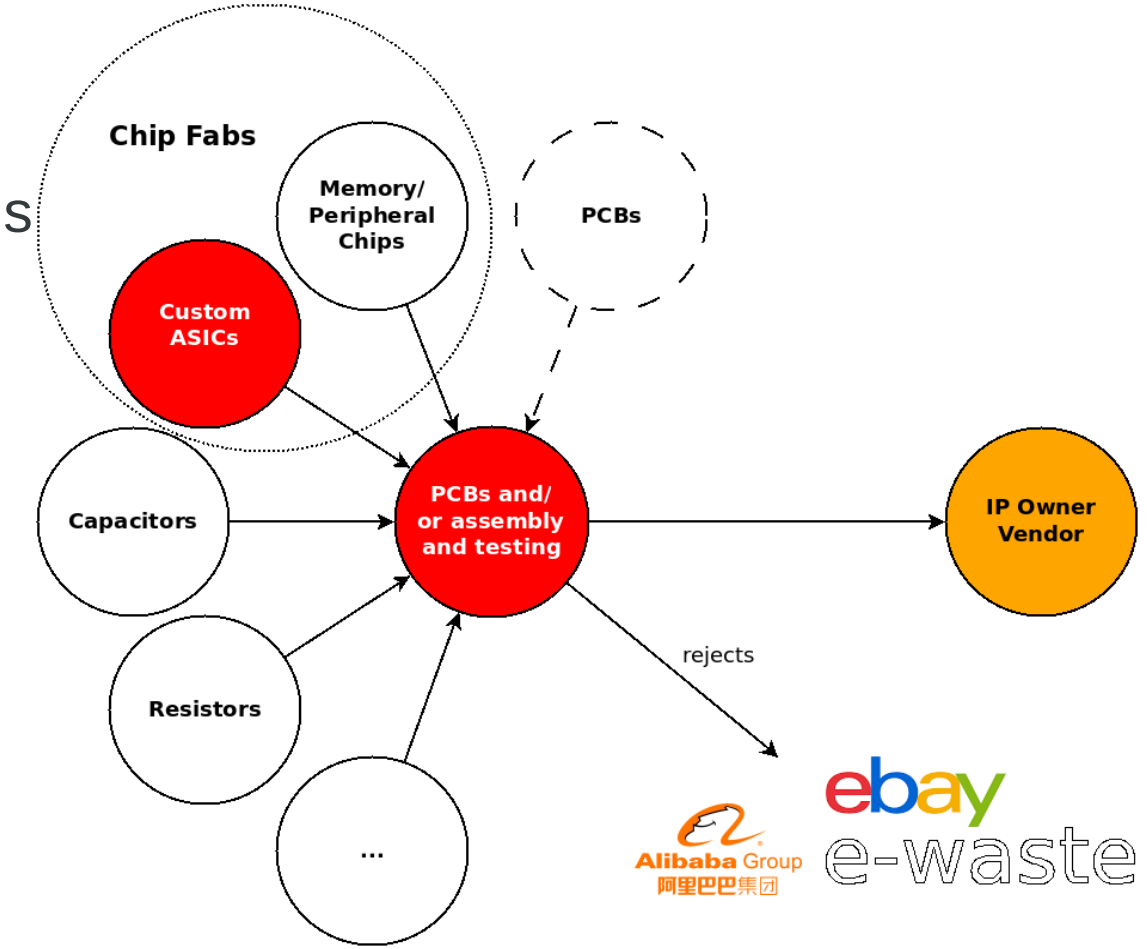
Where are the dangers of loosing IP?

Rejects:

- Leaks of prototypes
- Leaks of original boards
- Leaks of dev. boards

Espionage:

- Leaks of blueprints

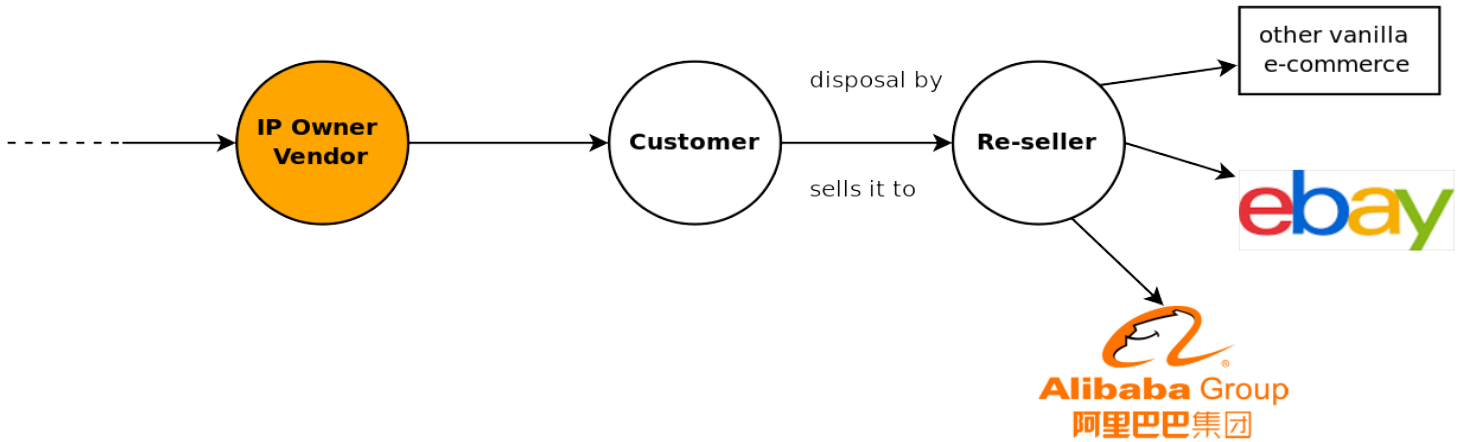


Dangers of supply-chains

Where are the dangers of losing IP?

Aftermarket issues:

A product, which is hard to unearth (very expensive or just available when you have a contract with the vendor) is available in big cheap batches from a re-seller. This enables you to do reverse engineering even with a small budget.



→ Cheap option from previous slide!

Dangers of supply-chains – Material Chasing

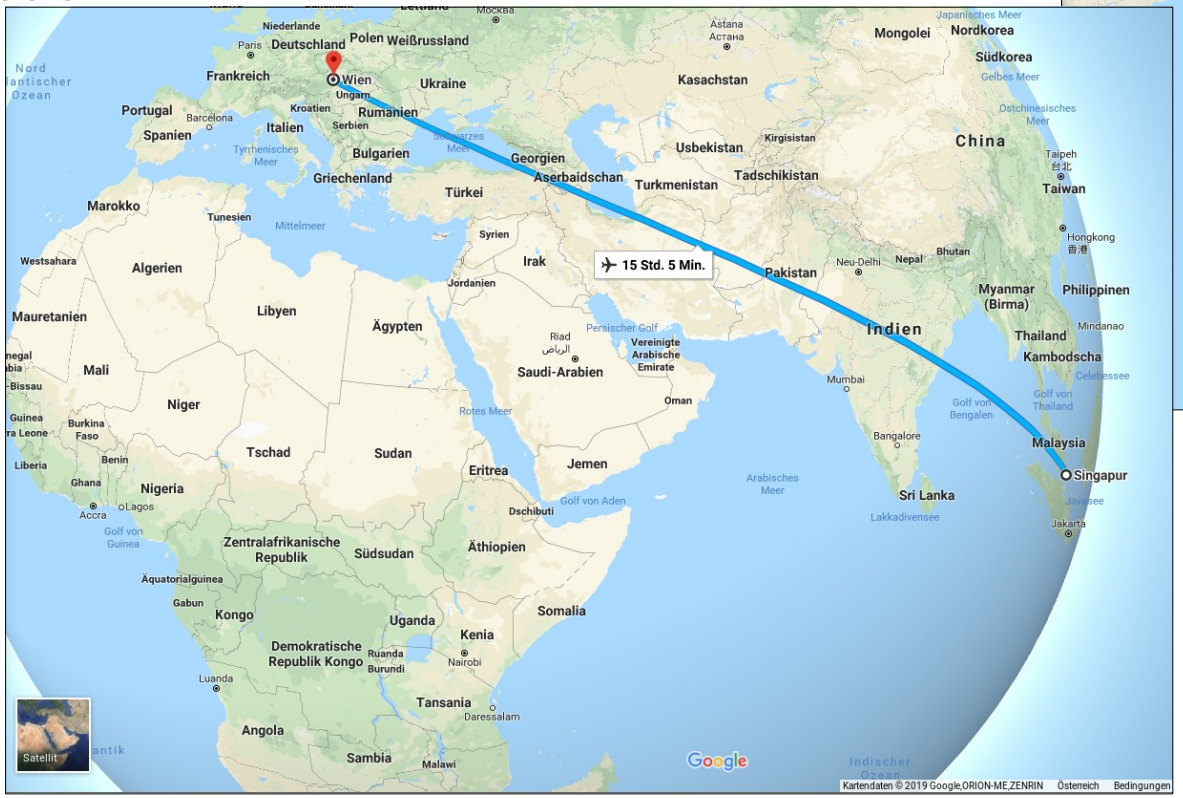
Searching for the label of the ASIC used in the older S7 1200v1 on Google gave some results, one of them was Taobao:

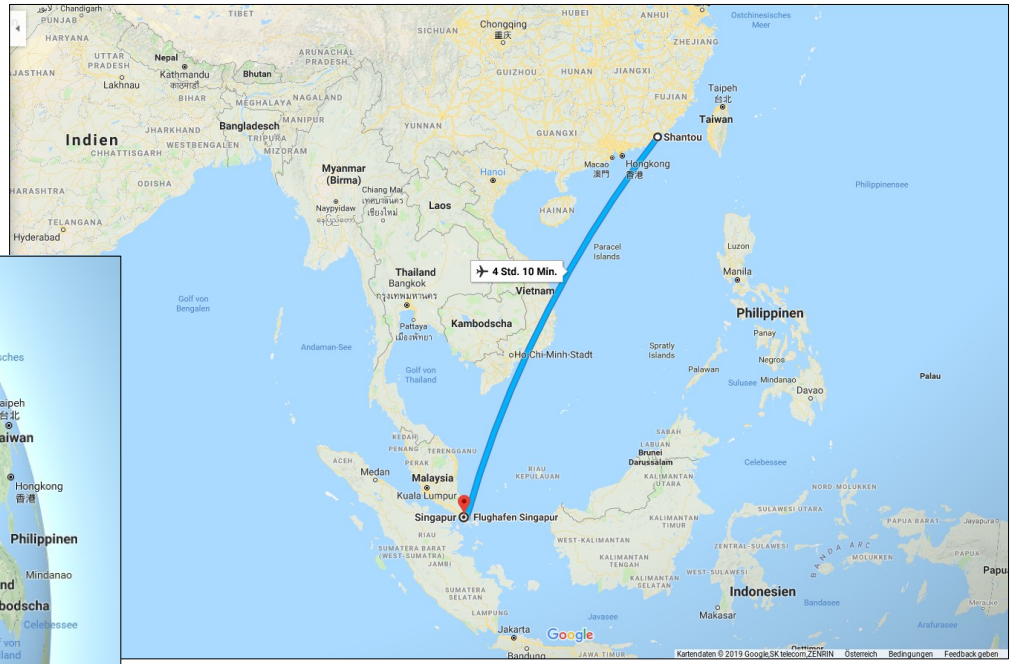
I paid 95¥

MB87M2230

Dangers of supply-chains – Material Chasing

Taobao just sells stuff inside China. Colleagues and friends from Singapore and China came to the rescue!





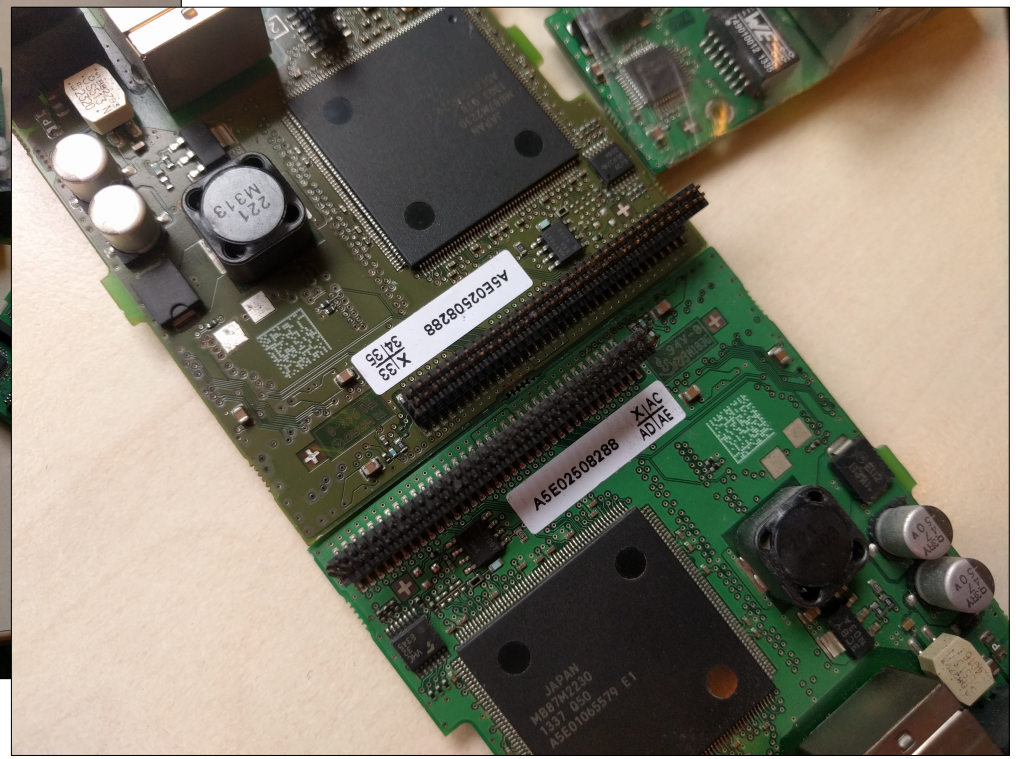
Two batches were ordered one after the other.

Dangers of supply-chains – Material Chasing

The first batch (MB87M2230)

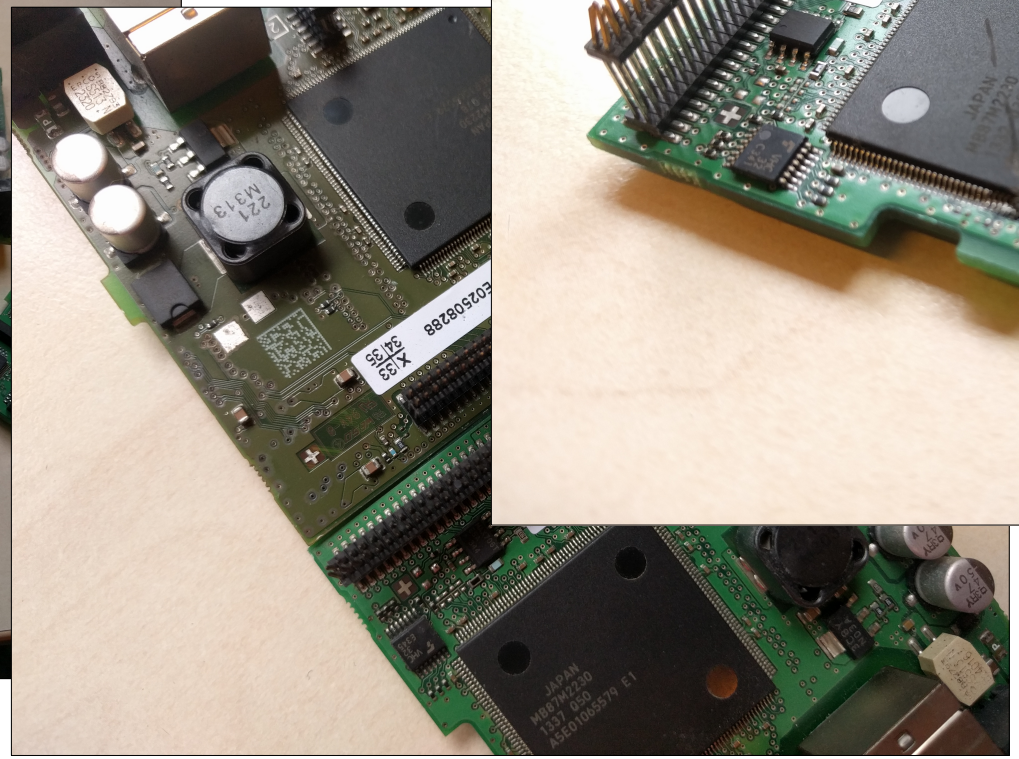
Dangers of supply-chains – Material Chasing

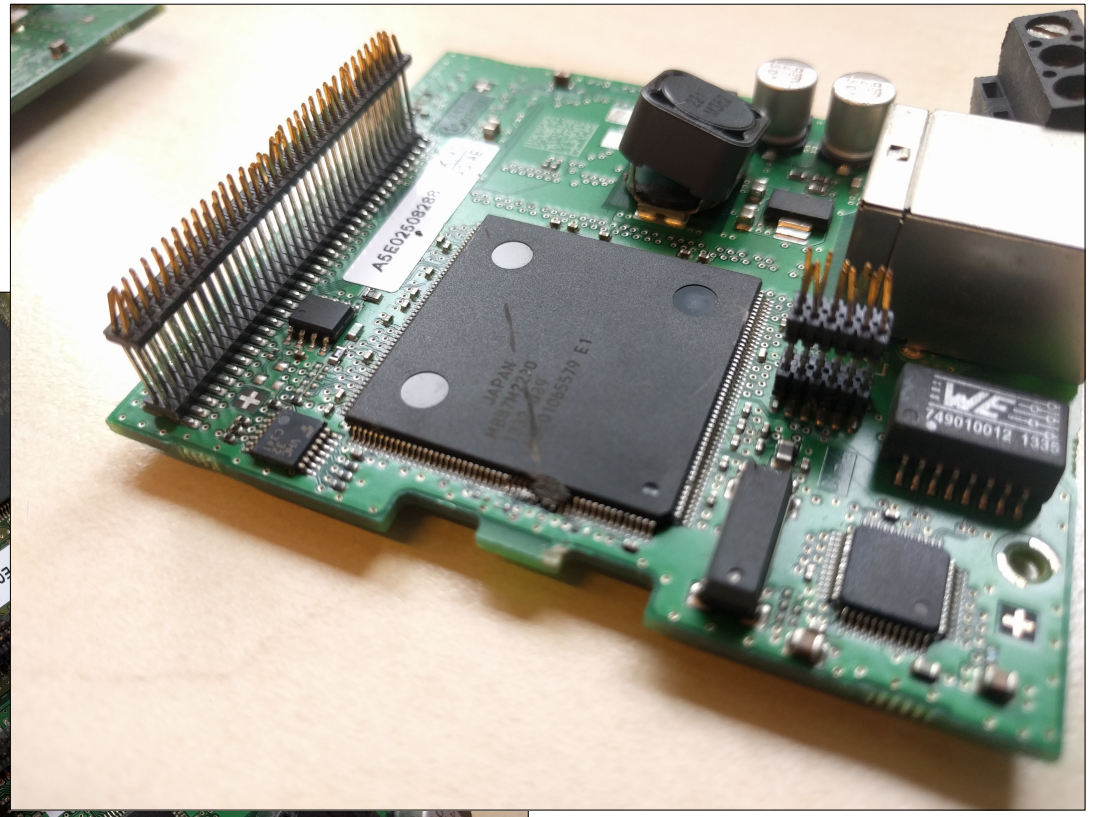
The first batch (MB87M2230)



Dangers of supply-chains – Material Chasing

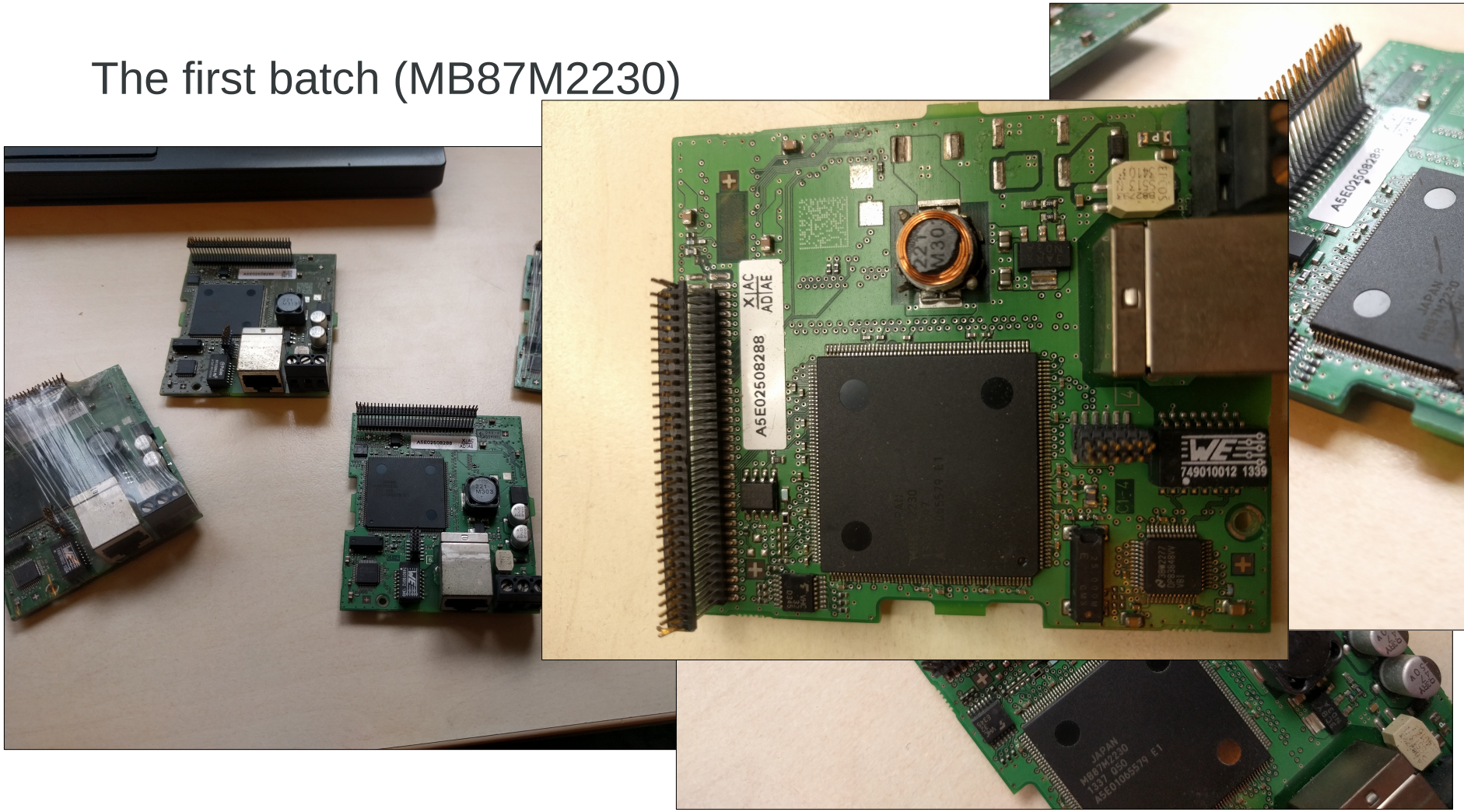
The first batch (MB87M2230)





Dangers of supply-chains – Material Chasing

The first batch (MB87M2230)



Dangers of supply-chains – Material Chasing

Searching for the label of the ASIC used in the newer S7 1200v4 on Google gave some results, one of them was Taobao, again from the same seller:

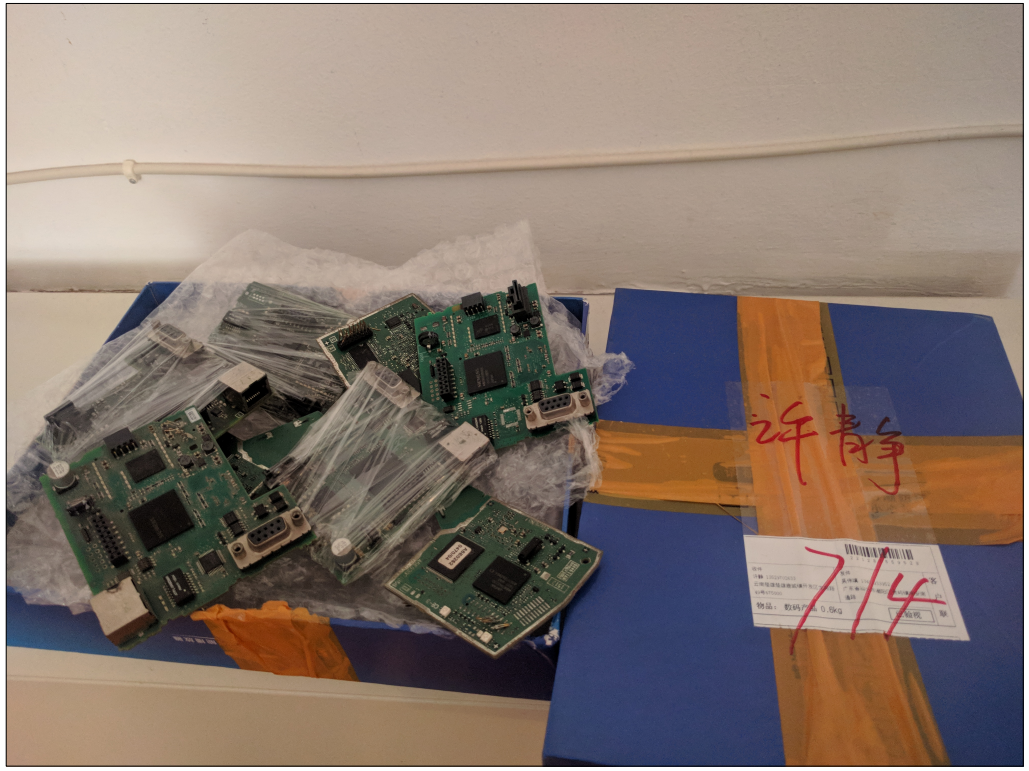
MB87M2230

Special Price!

This time just 80¥

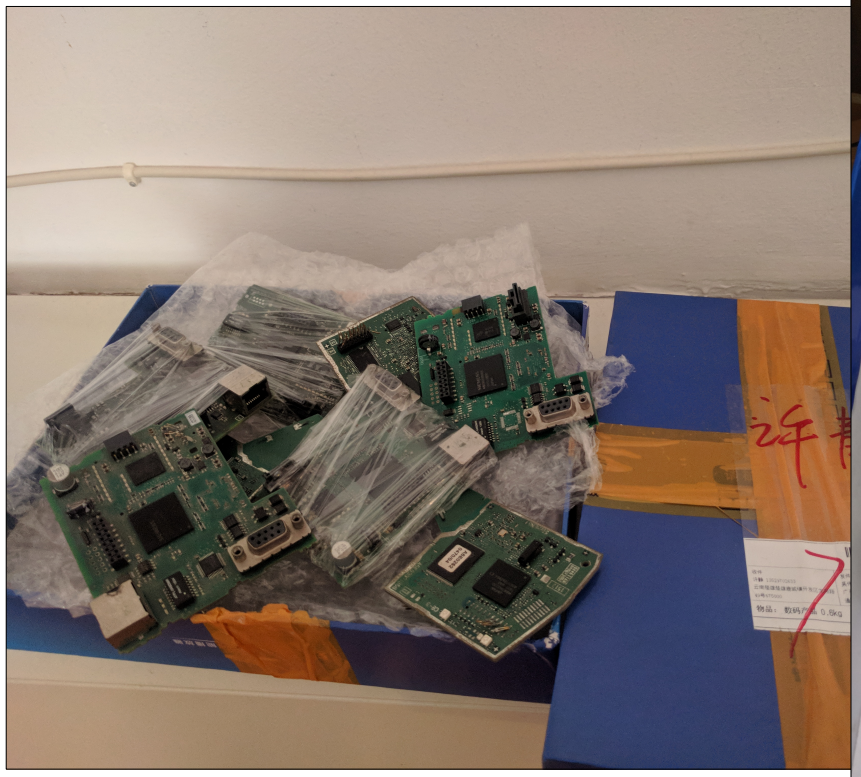
Dangers of supply-chains – Material Chasing

The second batch (A5E30235063)



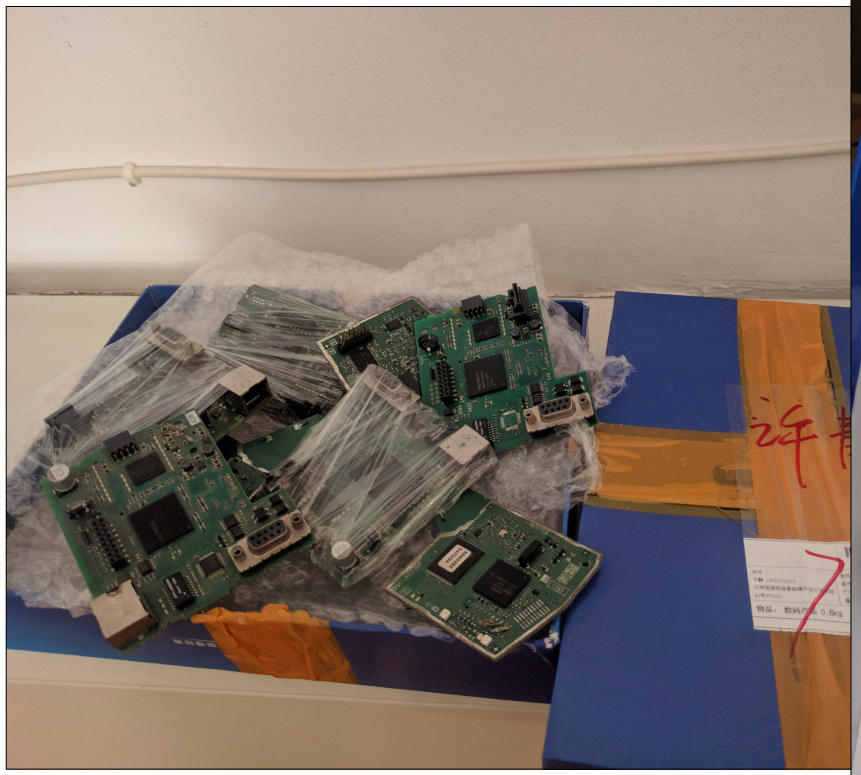
Dangers of supply-chains – Material Chasing

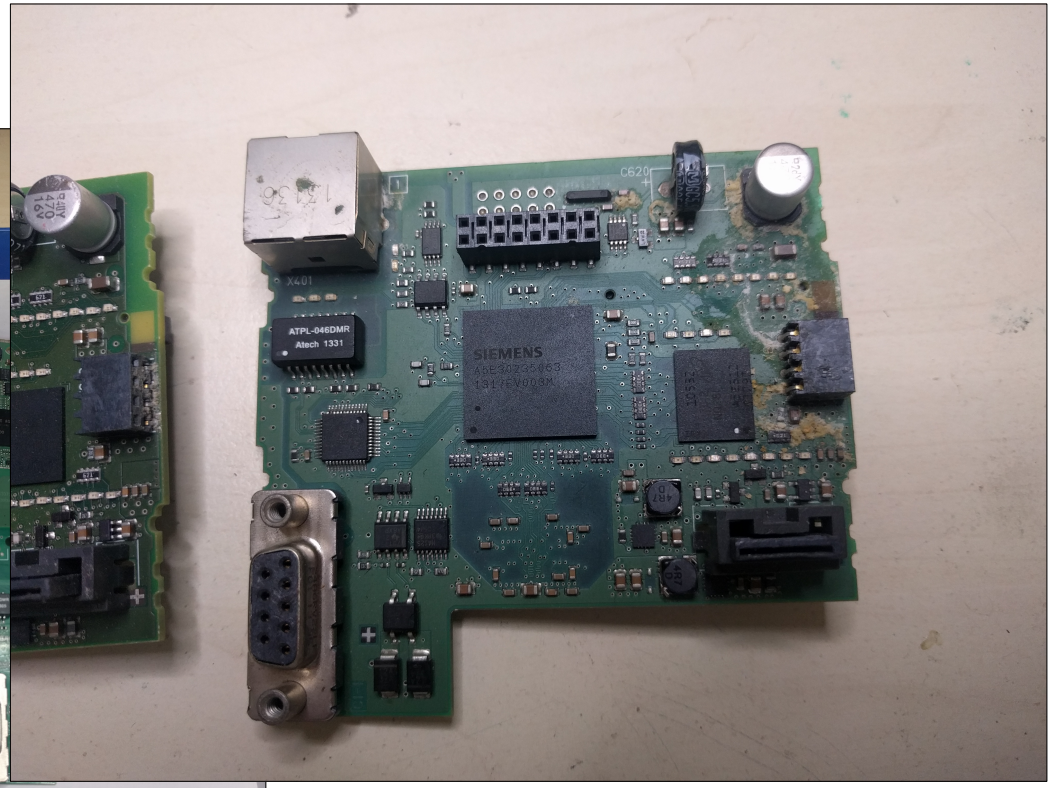
The second batch (A5E30235063)



Dangers of supply-chains – Material Chasing

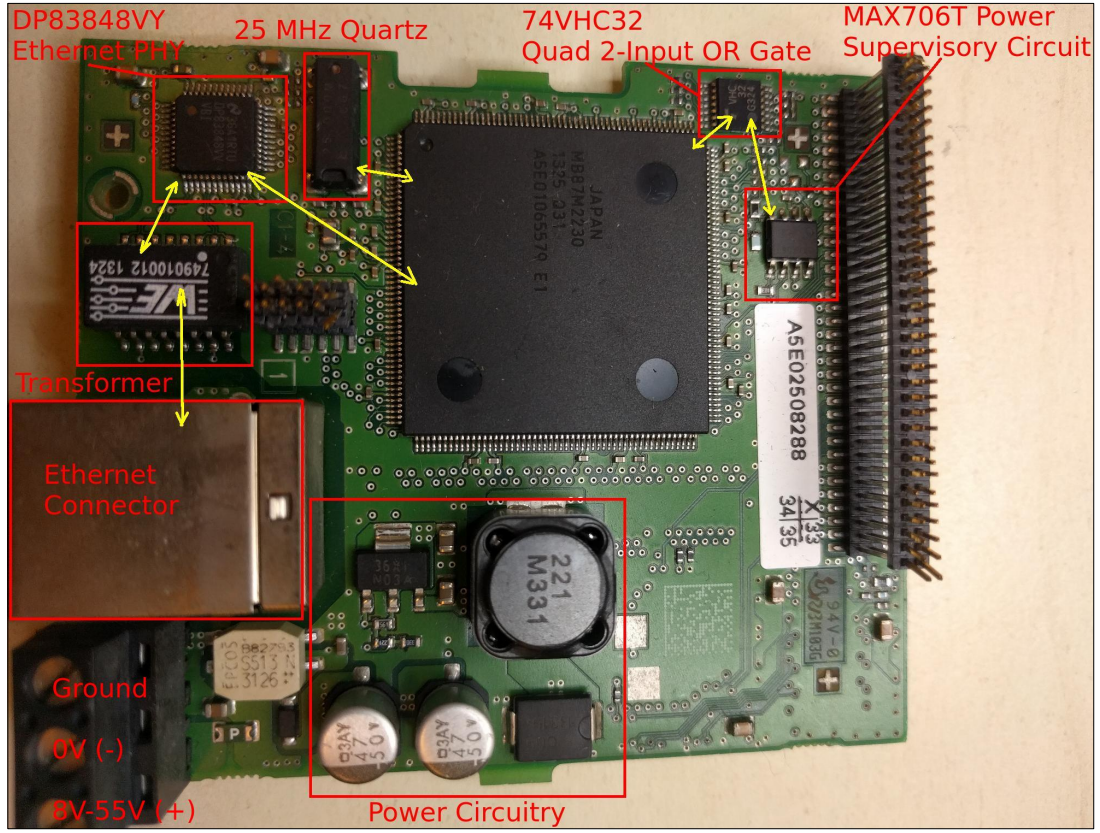
The second batch (A5E30235063)

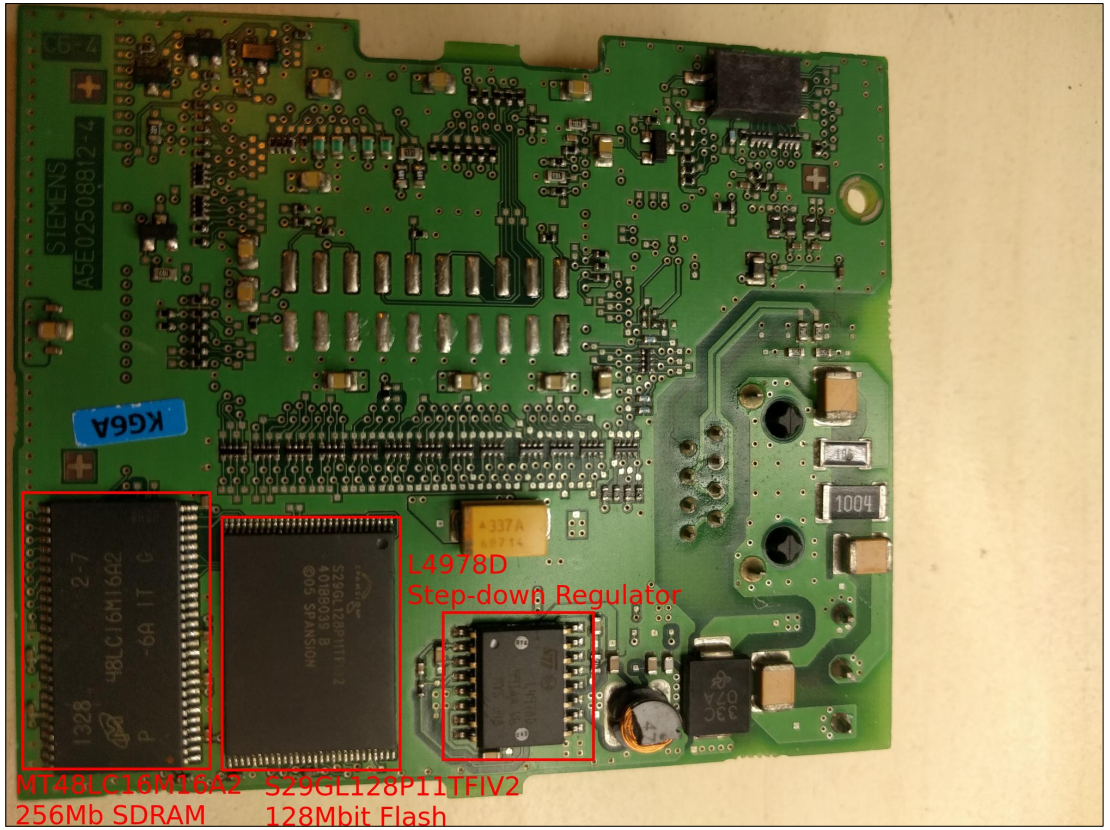




Reverse engineering methods – First batch of PCBs

Collecting datasheets by looking at the PCB:

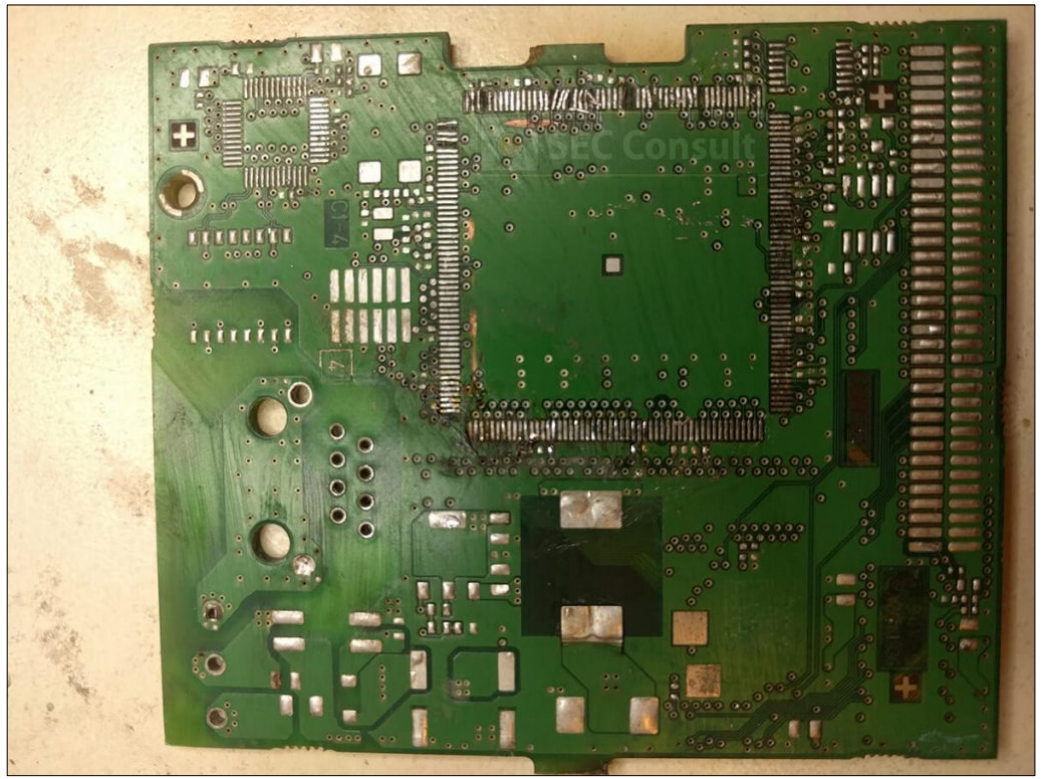


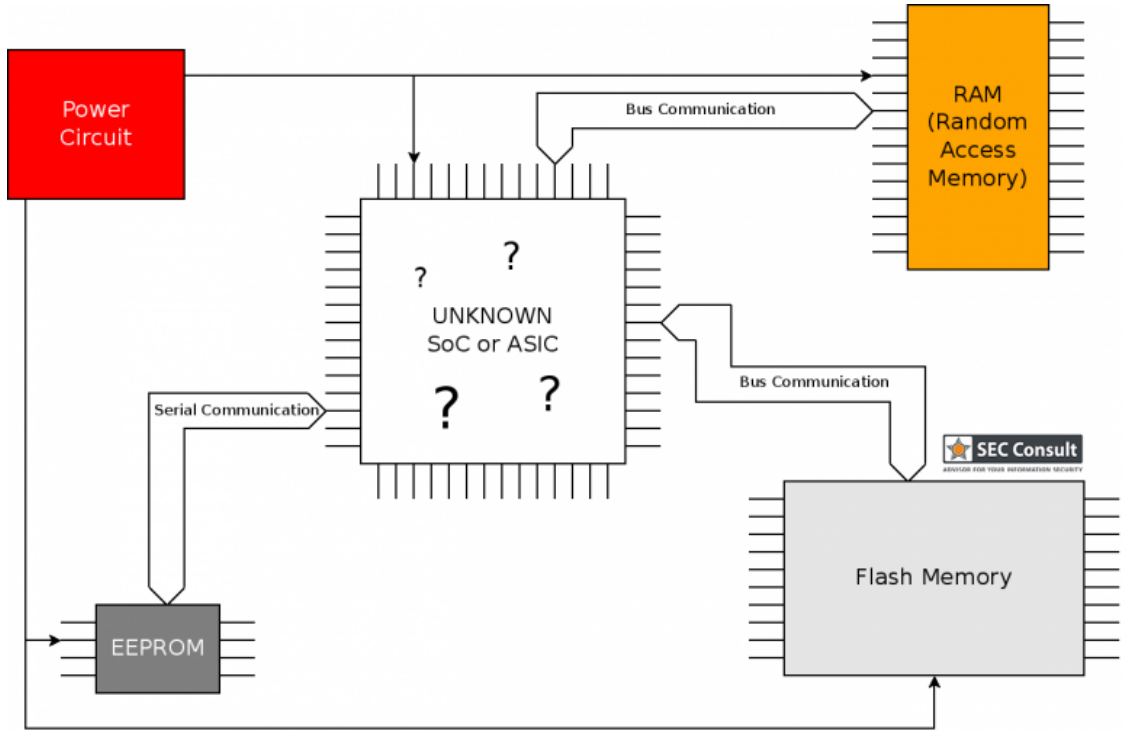


Reverse engineering methods – Deductive reasoning

Remove all parts from one PCB to be able to track all connections.

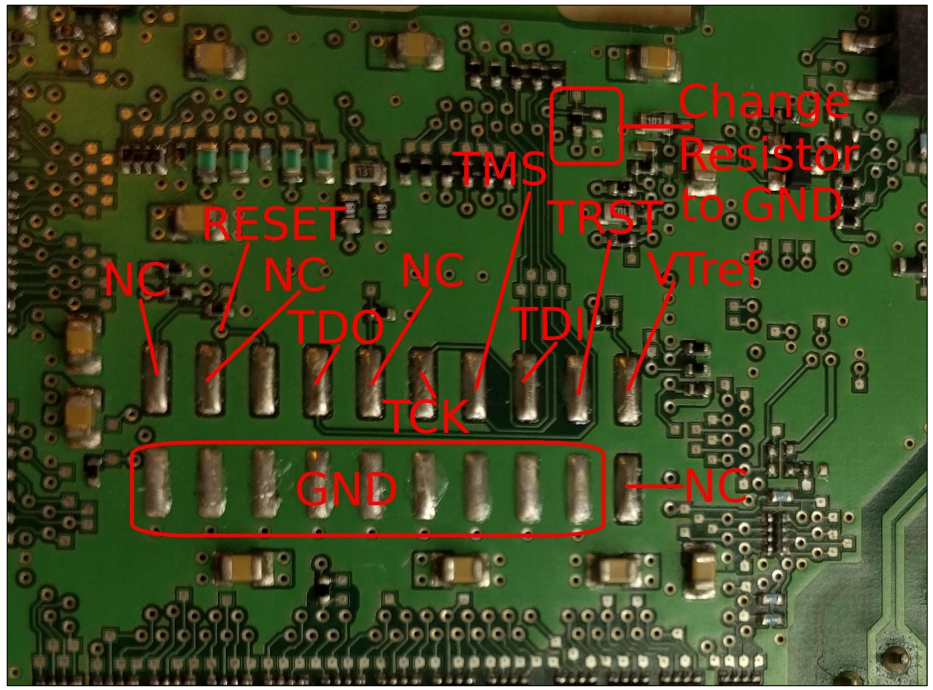
Determining the obvious Vdd pins.

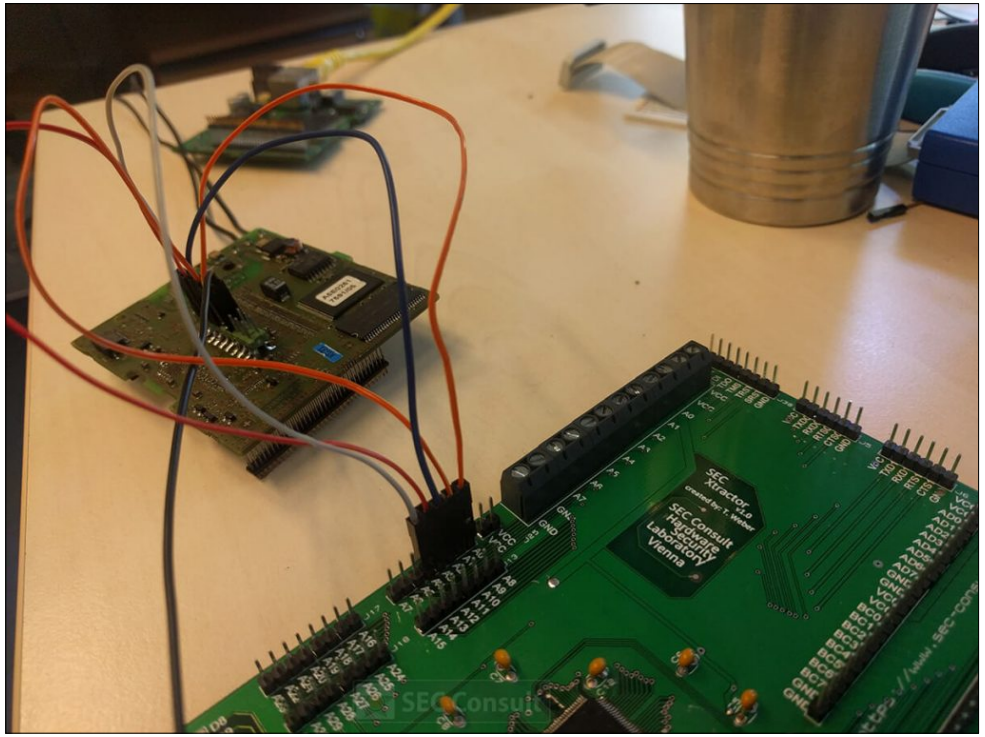




Reverse engineering methods – Deductive reasoning

Actively probing for debug interfaces, in this case for JTAG. Some pins were excluded from this test because of the prior step.





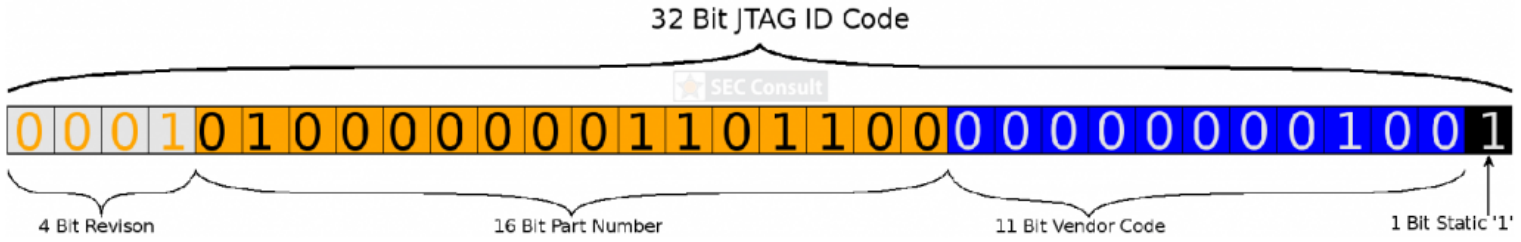
These pins are often pulled to Vdd by using a pull-up resistor!
They may be close to SPI or UART!

Reverse engineering methods – Deductive reasoning

After finding such a JTAG port, the ID-code can be fetched and interpreted:

```
[root@003-0089-0053 ~]# openocd -f /home/[redacted].cfg
Open On-Chip Debugger 0.10.0-dev-00247-g73b676c (2016-05-02-15:42)
Licensed under GNU GPL v2
For bug reports, read
        http://openocd.org/doc/doxygen/bugs.html
adapter speed: 500 kHz
jtag
Info : clock speed 500 kHz
Warn : There are no enabled taps.  AUTO PROBING MIGHT NOT WORK!!
Info : JTAG tap: auto0.tap tap/device found: 0x1406c009 (mfg: 0x004 (Fujitsu), part: 0x406c, ver: 0x1)
Warn : AUTO auto0.tap - use "jtag newtap auto0 tap -irlen 5 -expected-id 0x1406c009"
Warn : gdb services need one or more targets defined
```

Refer to JEDEC “JEP106AV”!



Reverse engineering methods – Deductive reasoning

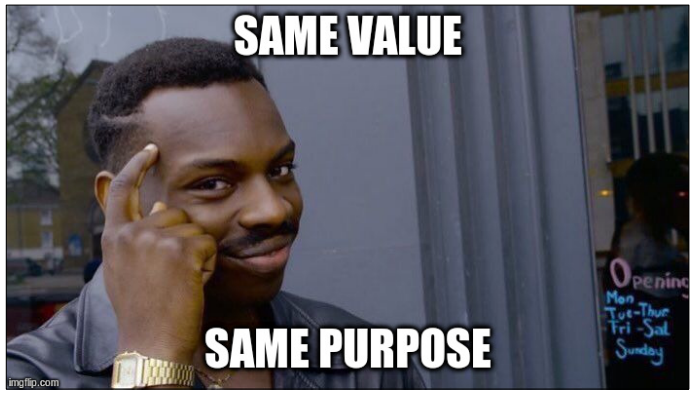
Besides JTAG, another challenging task is the detection of reset pins (**SRST** not **TRST**).

Common design patterns can help here, e.g.:

- The reset pin might be **bound to Vdd by** the **same** pull-up **resistor value** like all other ICs.

- The reset pin might be **switched** from Vdd to GND **by using a transistor**.

→ These two cases are very likely!

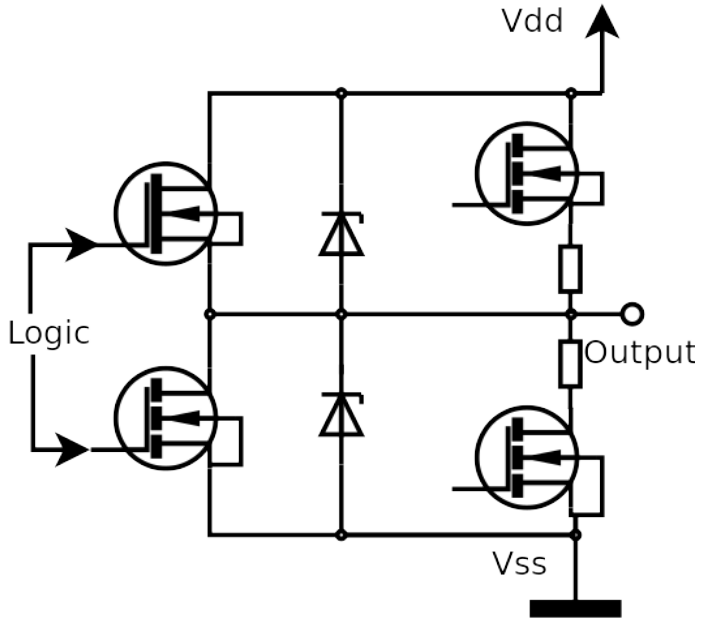


Quick test: Short circuit the pin to GND
(be sure to not kill the power IC)
BINGO! → When the CPU jumps to its reset vector!

Reverse engineering methods – Deductive reasoning

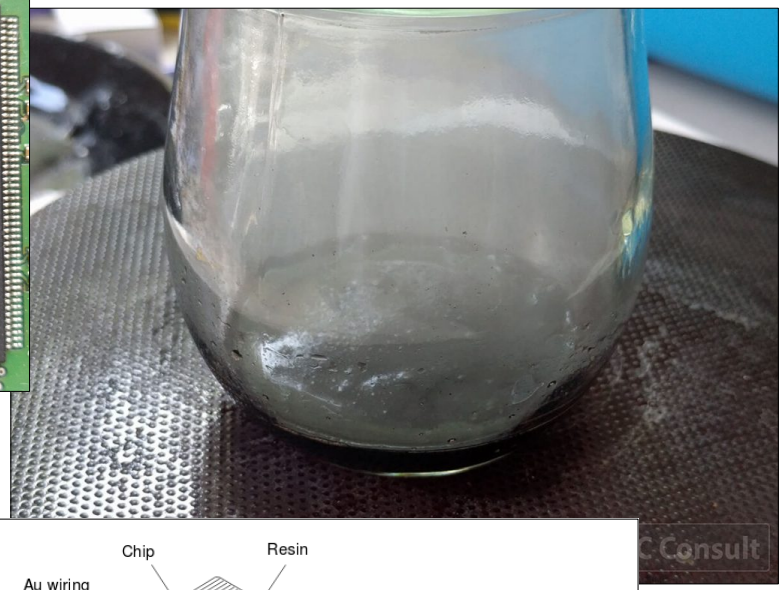
Whether a pin is an input, output or inout pin, can be determined by measuring the resistance of a pin. This is different from chip to chip and can be used as last step to identify the possible purpose of a pin.

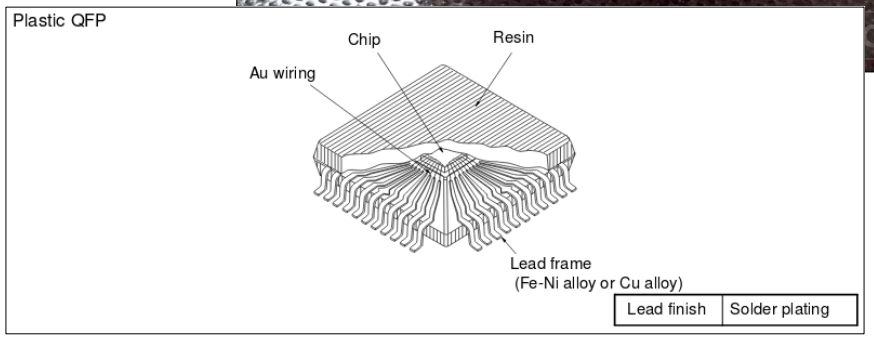
For example: Output logic

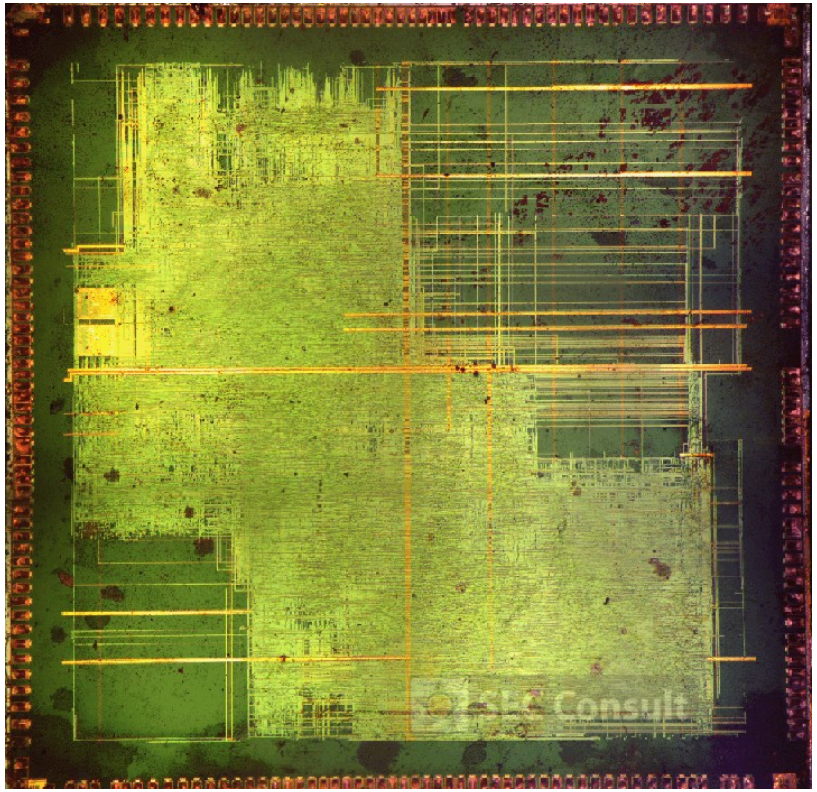


Deeper insights

Delicious cooking in sulfuric acid!







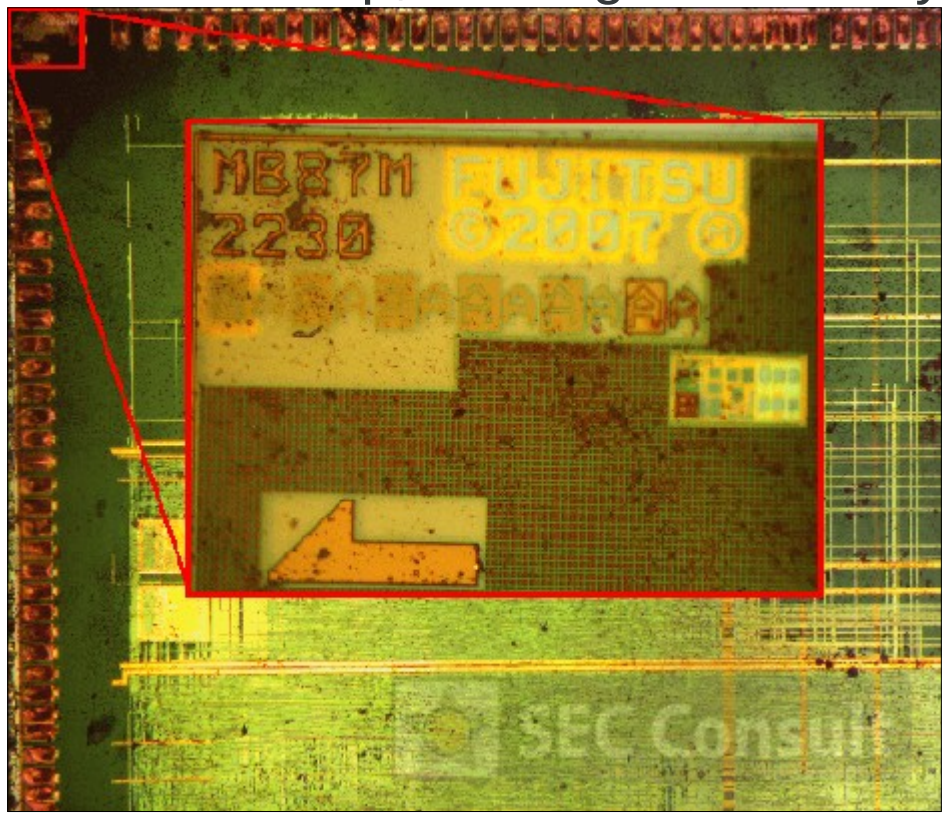
Don’t do that at home!

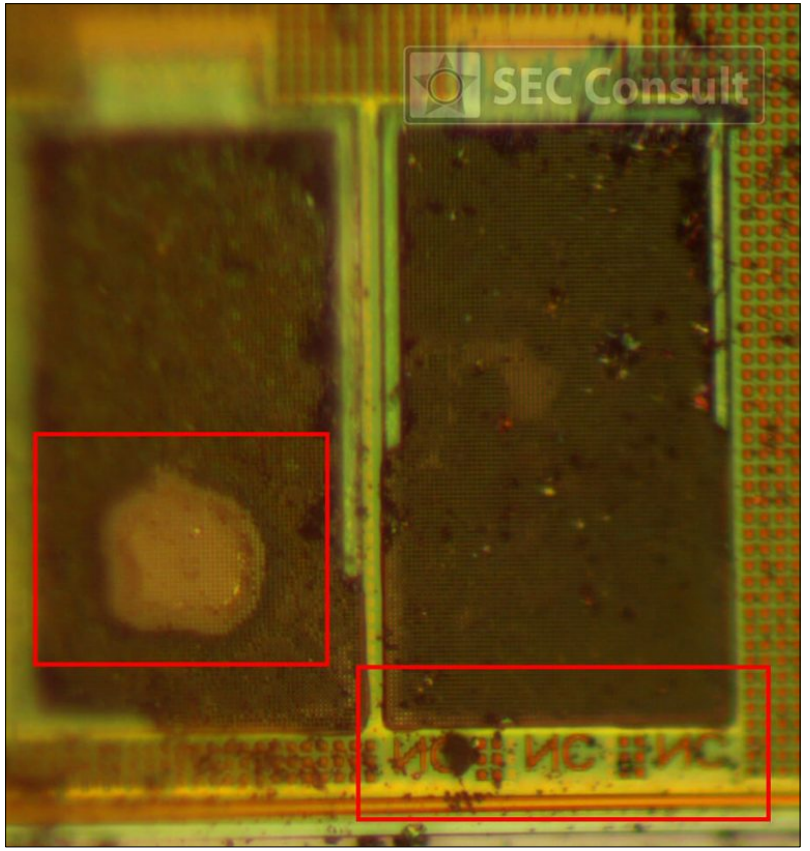
https://www.fujitsu.com/downloads/MICRO/fma/pdfmcu/packageguide-contents-x1.pdf

Deeper insights

The labels on the bare die sometimes reveal important information.

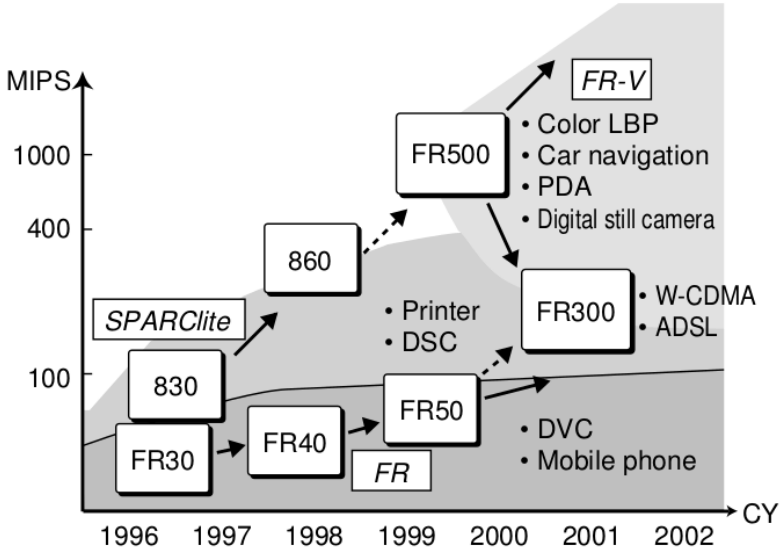
For this chip, it was good to verify the JTAG output – it was designed by Fujitsu.



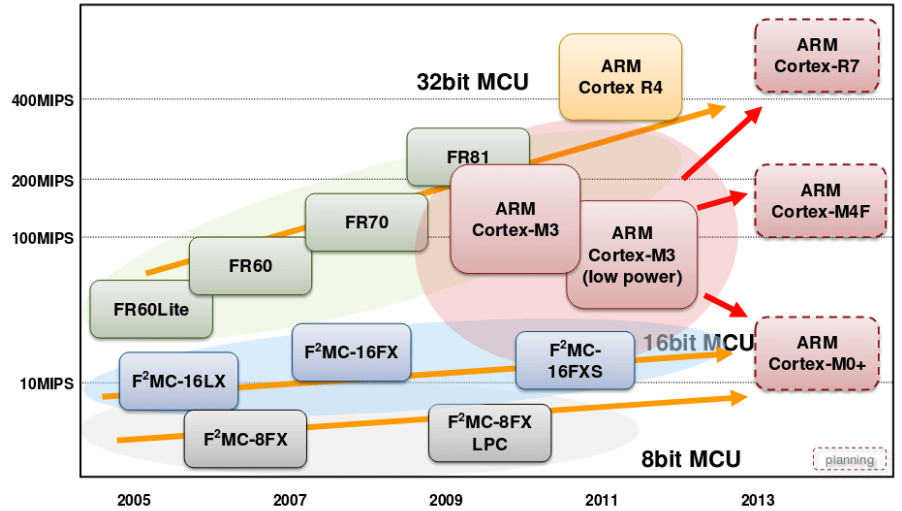


Deeper insights – Digging through the literature

So many possibilities!



https://www.fujitsu.com/global/documents/about/resources/publications/fstj /archives/vol36-1/paper06.pdf

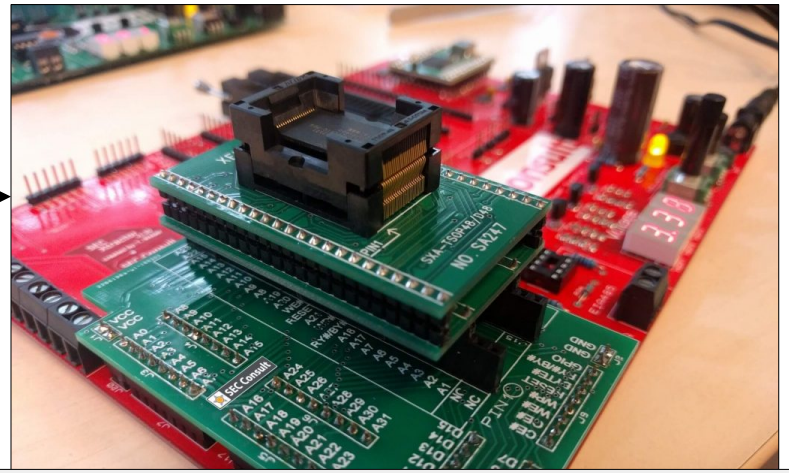


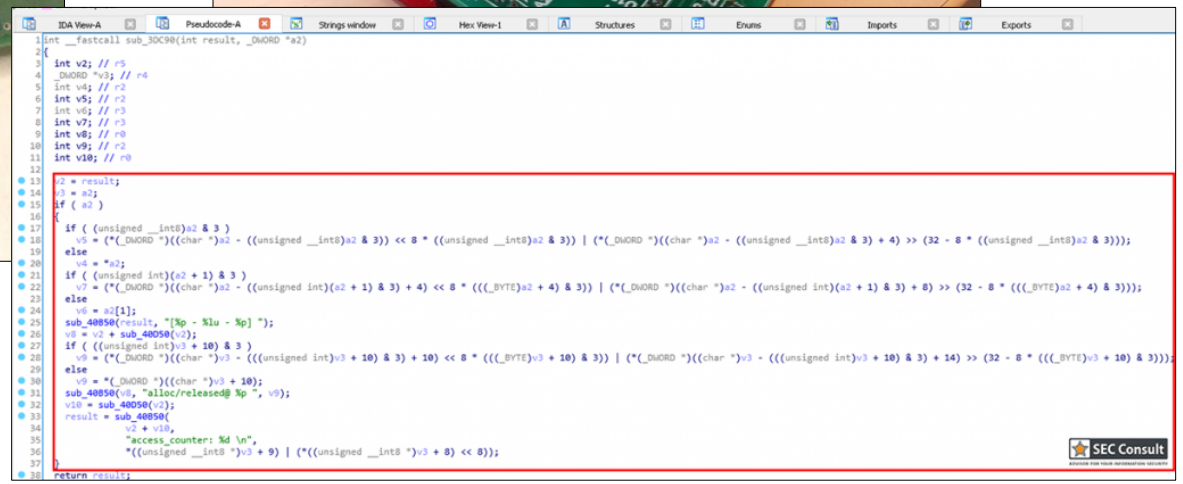
http://docplayer.net/4207609-Right-sized-solutions-for-embedded-applications.html

Other MB8xMxxxx chips have ARC Tangent processors, or Fujitsu RISC (FR). It can also be F²MC….

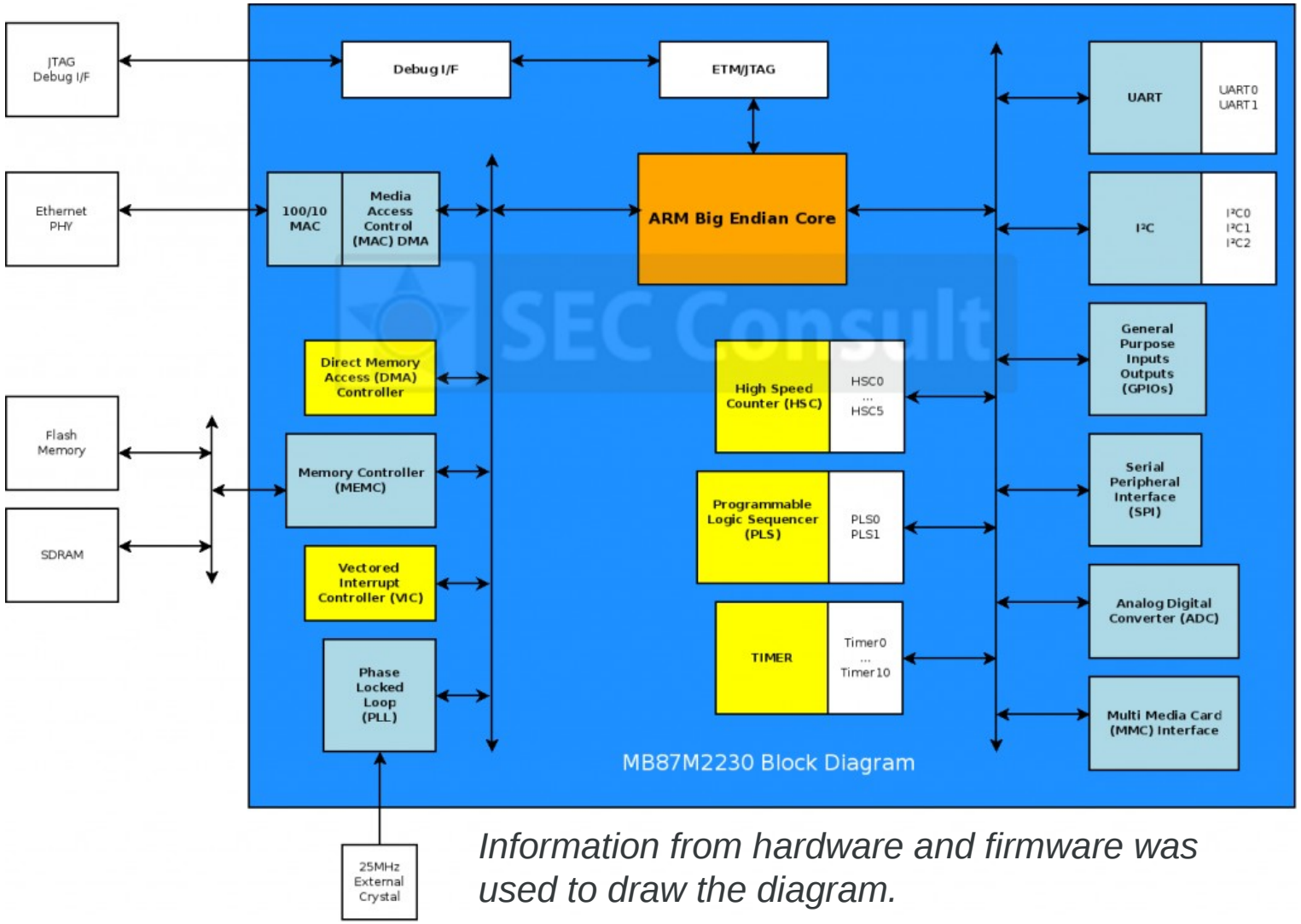
Deeper insights

Removing the flash memory and reading out its content always helps.





Deeper insights

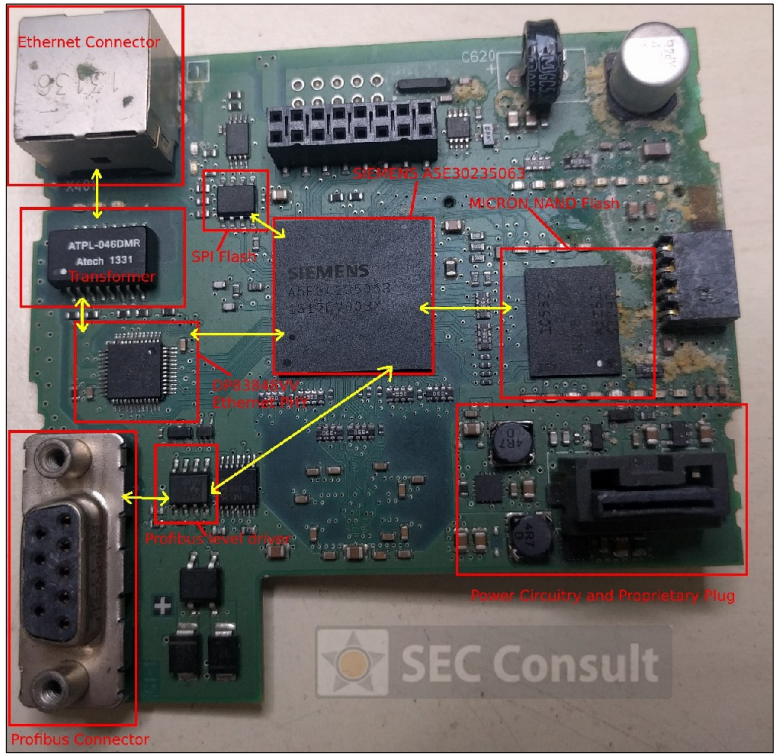


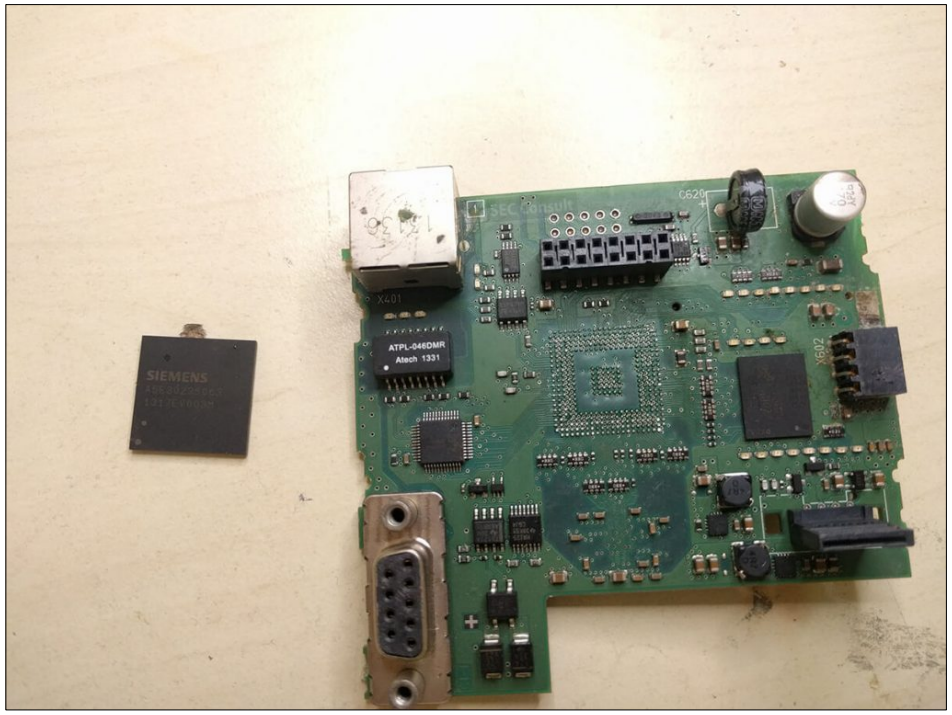
Information from hardware and firmware was used to draw the diagram.

By combining the information of the used CPU core, the year and the available IP cores from Fujitsu at that time we can be pretty sure that ARM926/ARM946 is used.

Deeper insights – Second batch

The second batch of PCBs can be analyzed in the same way as the first one.

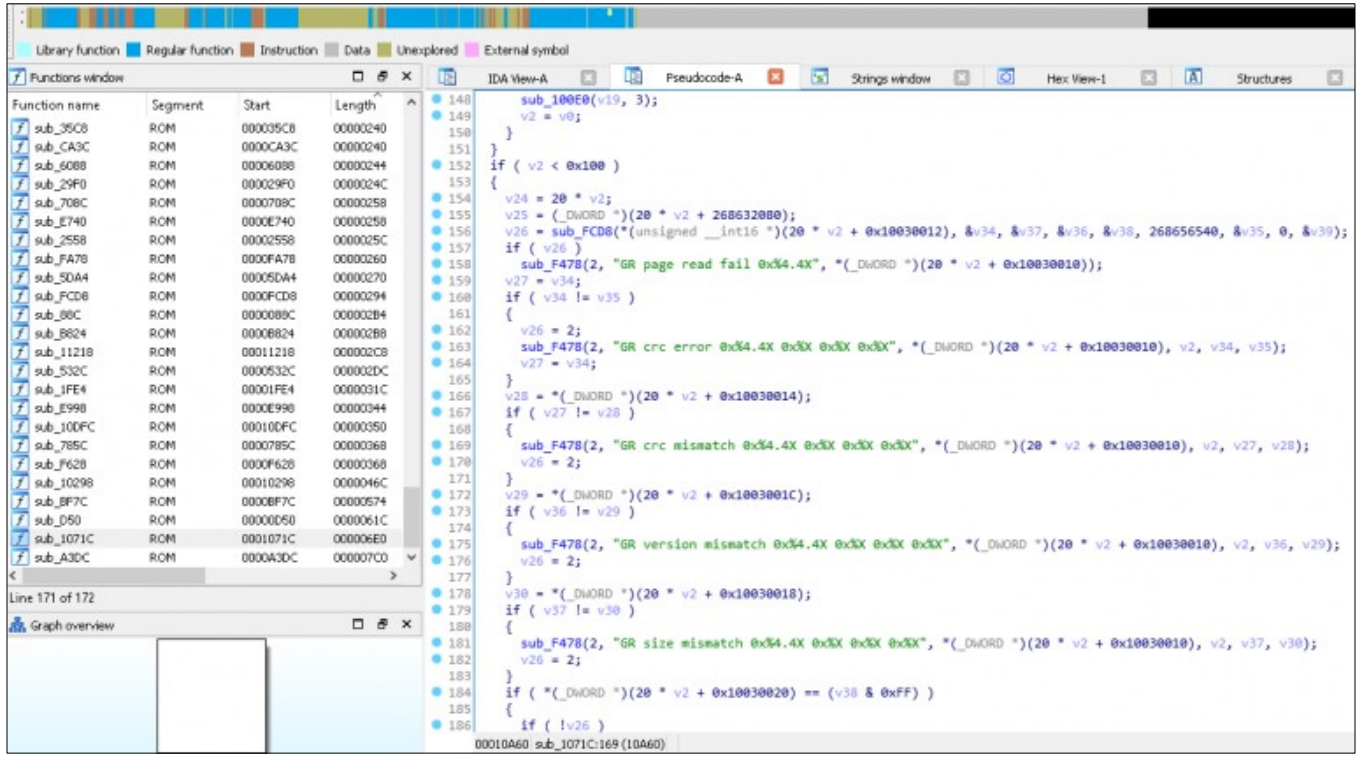




The different architectures of SPI flash + NAND flash were one of the first observations.

Deeper insights – Second batch

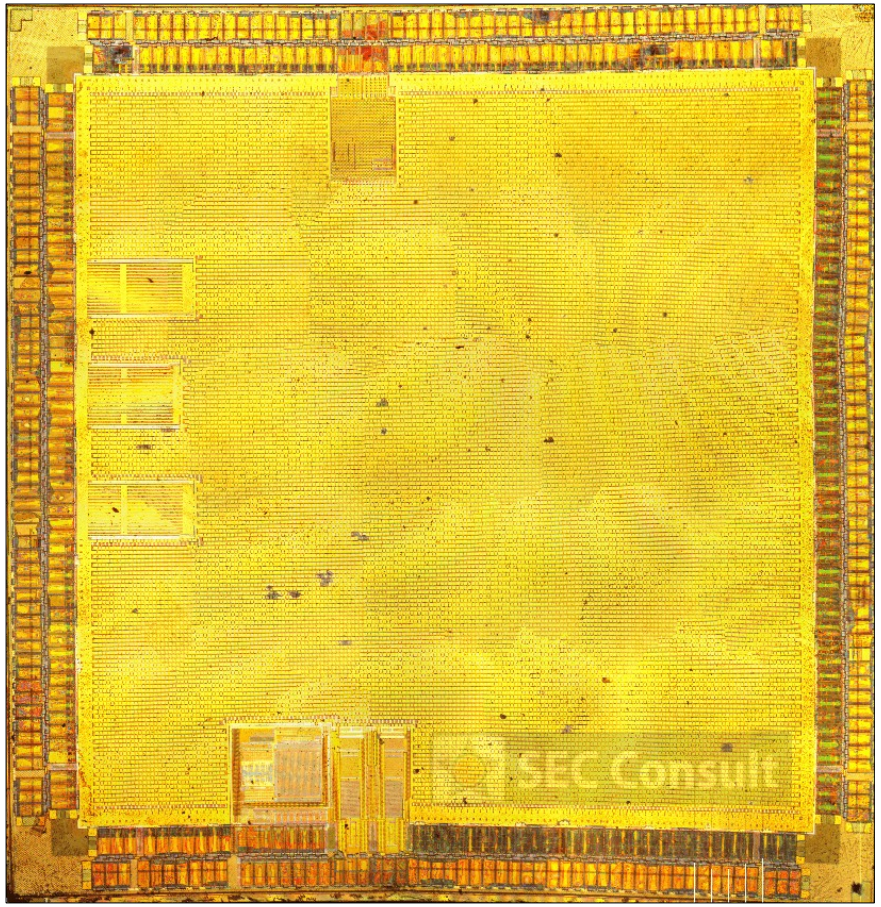
The bootloader, which is located at the SPI flash memory was dumped and loaded into IDA Pro:

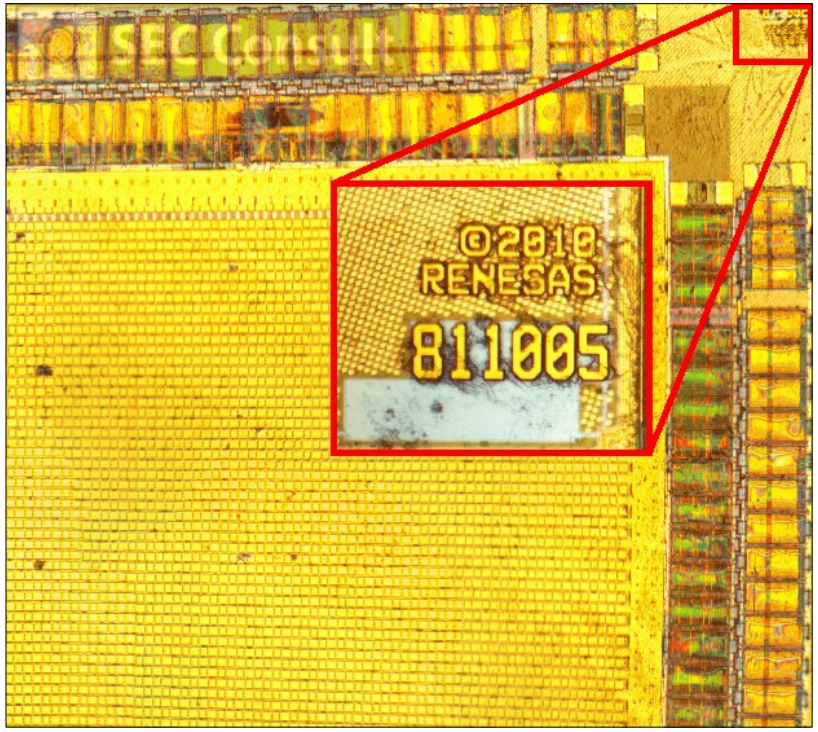


Most strings were referenced immediately, ARM big endian was used here too.

Deeper insights – Second batch

It turned out that the newer chip (A5E30235063) was designed by Renesas.



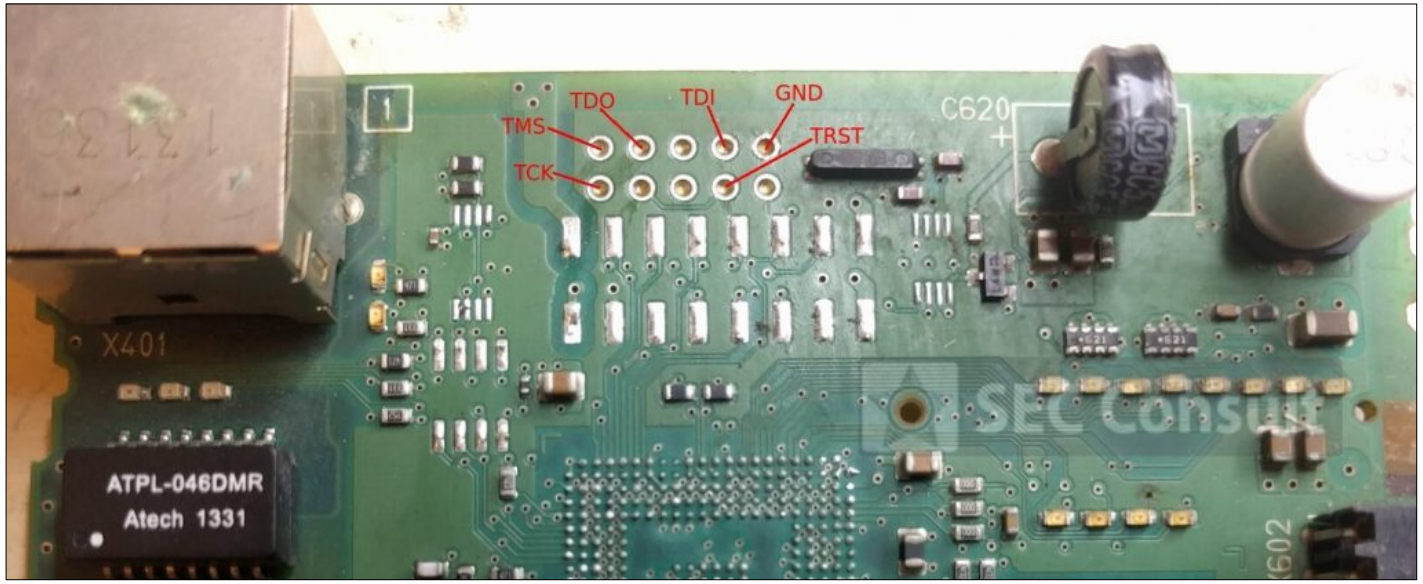


There are some similarities to ERTEC 200P/400 (Siemens/Renesas).

Deeper insights – Second batch

By brute forcing the 10-pin header of the PCB a JTAG port was found!

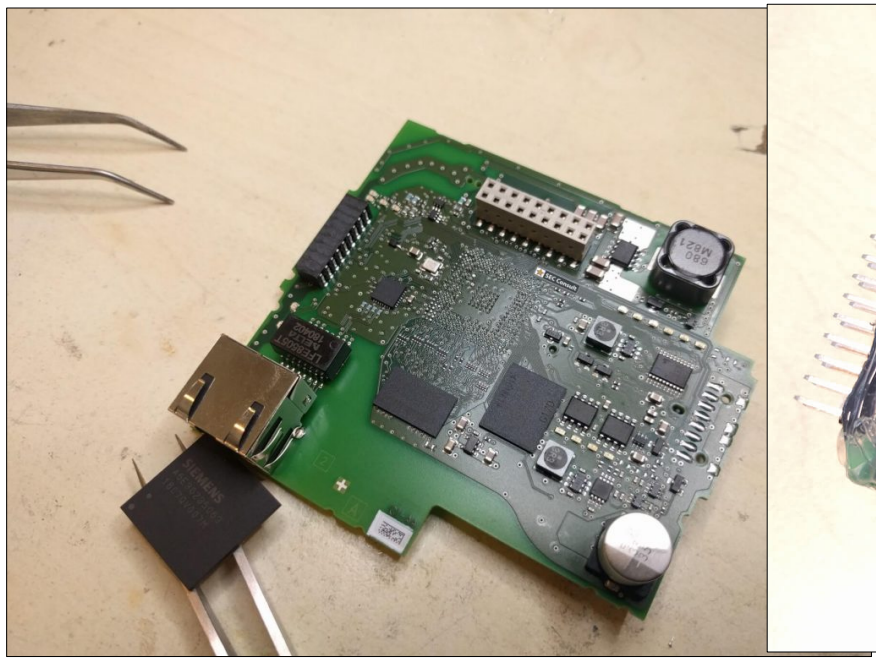
```
Connecting to target via JTAG
TotalIRLen = 4, IRPrint = 0x01
JTAG chain detection found 1 devices:
 #0 Id: 0x4BA00477, IRLen: 04, CoreSight JTAG-DP
Scanning AP map to find all available APs
AP[3]: Stopped AP scan as end of AP map has been reached
AP[0]: AHB-AP (IDR: 0x44770001)
AP[1]: APB-AP (IDR: 0x24770002)
AP[2]: JTAG-AP (IDR: 0x14760010)
Iterating through AP map to find AHB-AP to use
AP[0]: Skipped. Not an APB-AP
AP[1]: APB-AP found
ROMTbl[0][0]: CompAddr: 80008000 CID: B105900D, PID:04-003BB907 ETB
ROMTbl[0][1]: CompAddr: 80003000 CID: B105900D, PID:04-003BB906 CTI
ROMTbl[0][2]: CompAddr: 80004000 CID: B105900D, PID:04-001BB908 CSTF
ROMTbl[0][3]: CompAddr: 80002000 CID: B105900D, PID:04-007BBC14 Cortex-R4
Found Cortex-R4 r1p3
8 code breakpoints, 8 data breakpoints
Debug architecture ARMv7.0
Data endian: big
Main ID register: 0x411FC143
I-Cache L1: 16 KB, 128 Sets, 32 Bytes/Line, 4-Way
D-Cache L1: 16 KB, 128 Sets, 32 Bytes/Line, 4-Way
TCM Type register: 0x00010001
MPU Type register: 0x00000C00
System control register:
  Instruction endian: big
  Level-1 instruction cache disabled
  Level-1 data cache disabled
  MPU disabled
  Branch prediction enabled
Memory zones:
  Default  Default access mode
  AHB-AP (AP0)  DMA like acc. in AP0 addr. space
  APB-AP (AP1)  DMA like acc. in AP1 addr. space
Cortex-R4 identified.
```

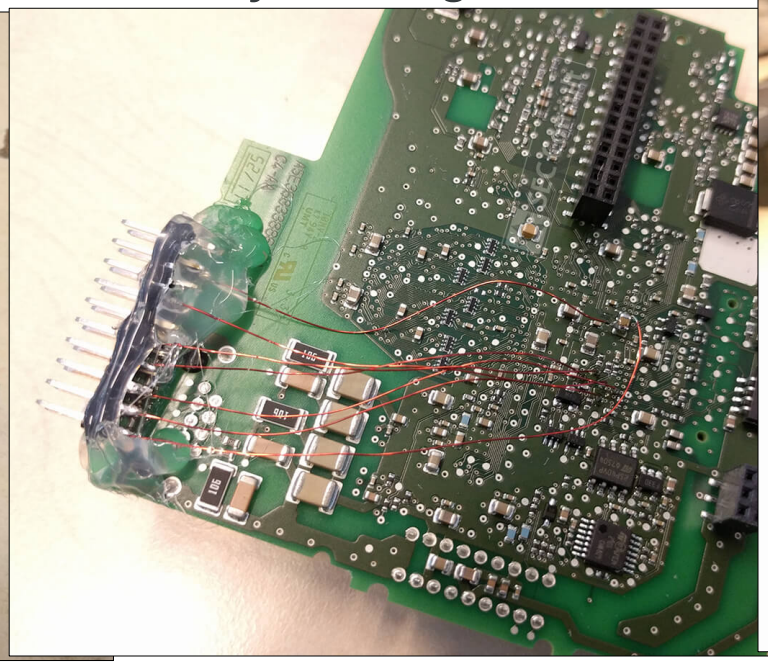


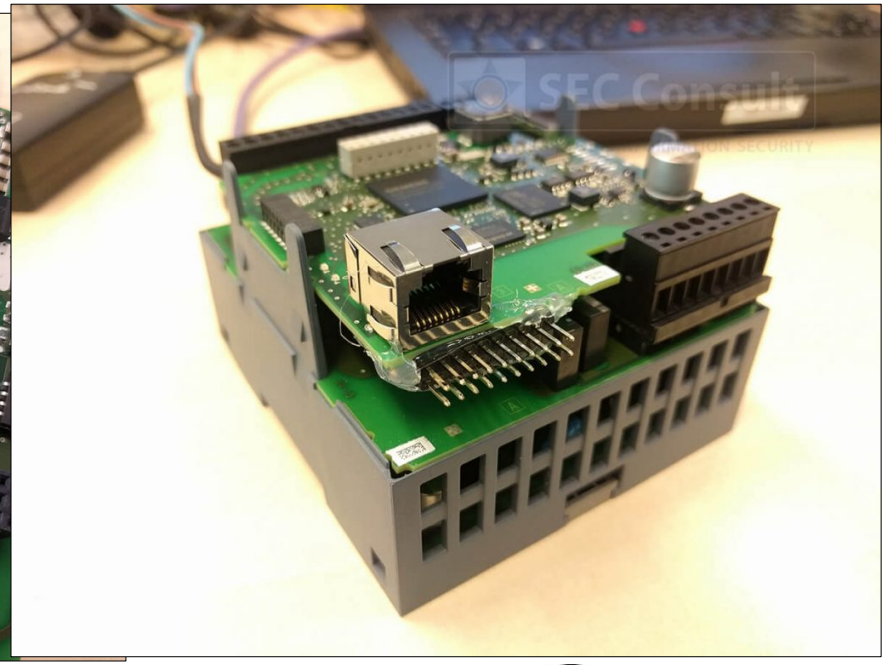
ARM Cortex R4 was identified. Now it was easy to trace the connections back to the chip!

Deeper insights – Second batch

After removing the chip of an original S7 1211C, the traces can be followed back to the backside. JTAG can be enabled by adding an additional header to the PCB.







Beware, when you attach the debugger! It seems that Siemens have implemented a hardware module for deleting the flash memory when the CPU is stopped!!!

Demo time!

To provide a proof of concept, a small assembly program was written and uploaded to the PLC via the JTAG interface.

Special thanks goes to Dr. Ali Abbasi for providing me the UART MMIO address.
https://www.syssec.ruhr-uni-bochum.de/chair/staff/aliabbasi/

Fun Fact

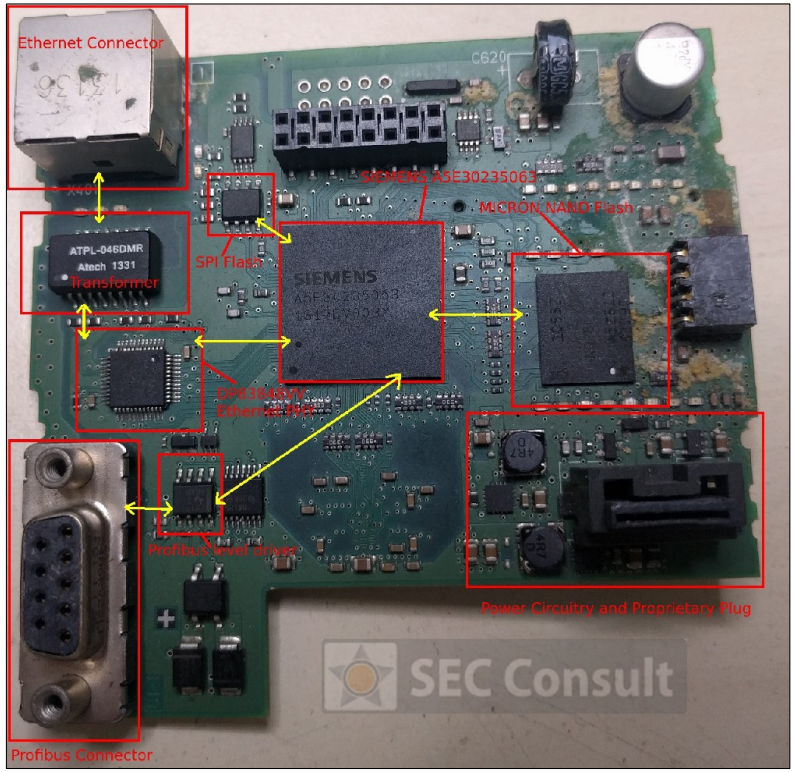
Few days before publishing our research, we received the following statement from Siemens:

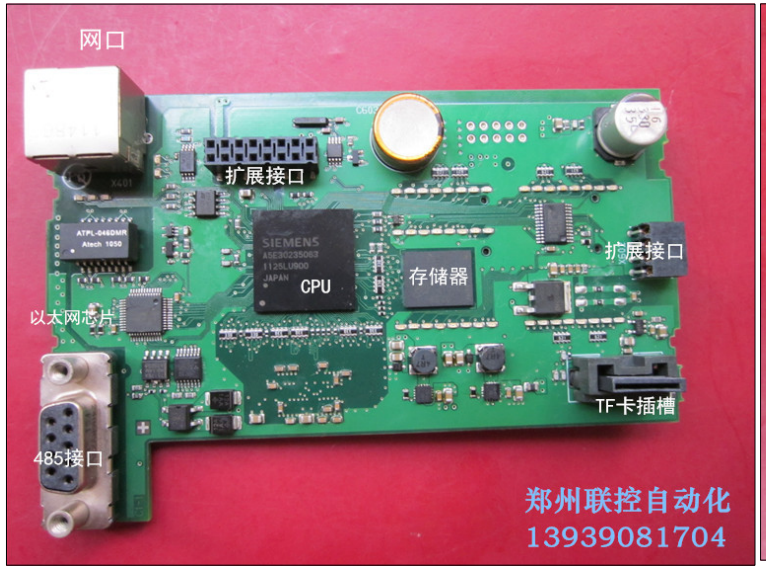
> *"The boards purchased by SEC Consult were not development boards but previously used or refurbished boards from Siemens devices. Siemens does not see a supply chain leak."*

As it turns out, I was looking at boards from another series. The seller from Taobao fooled me. He offered boards from the older **S7-200 SMART** series labeled as **S7-1200** series … but no bad feelings: the board had **JTAG**!

Fun Fact

Can you spot the similarities?

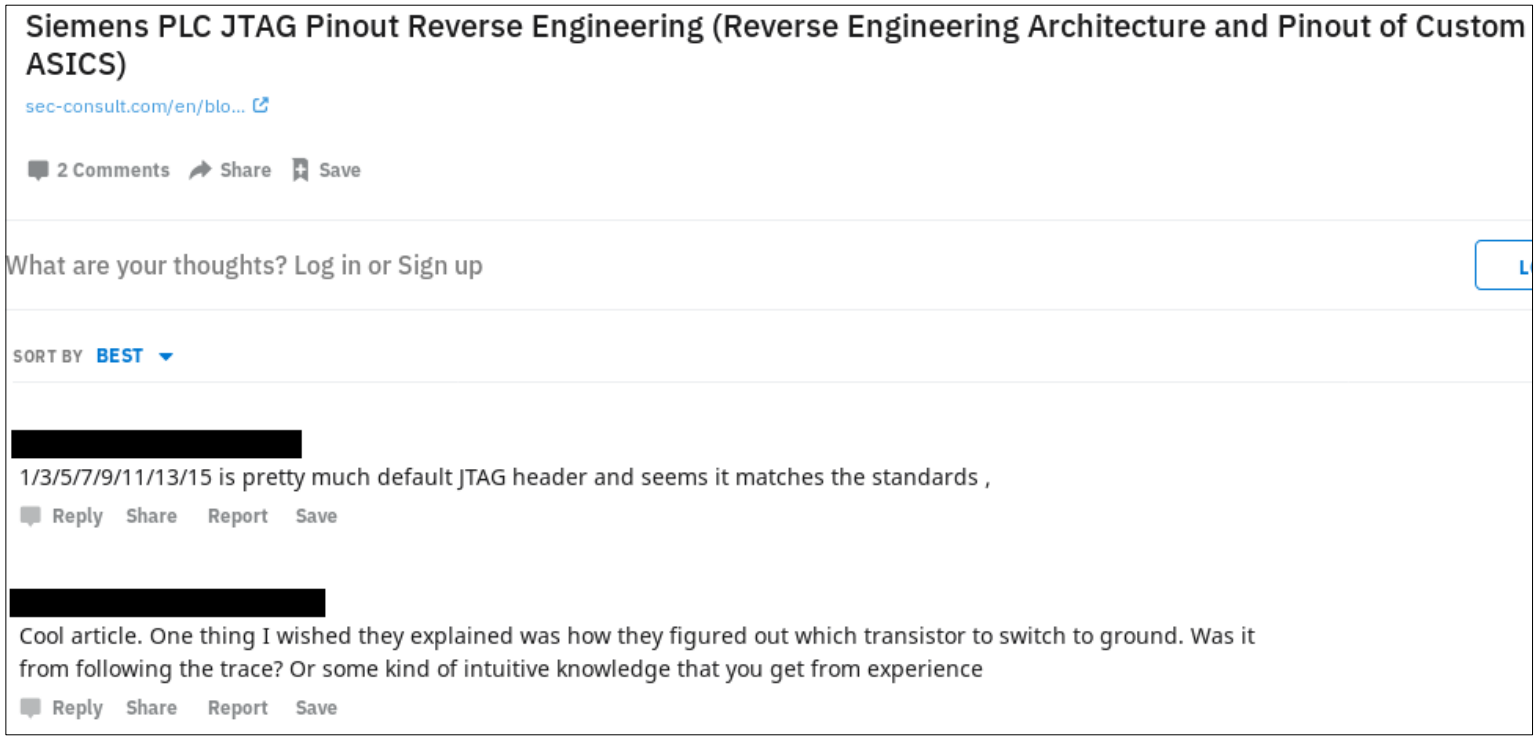




S7-200 SMART http://www.plcweixiu.com/news/html/390.html

Q&A

One question on reddit, do you have another one?



Thank you!

Find the full blogpost here:

https://sec-consult.com/en/blog/2019/02/reverse-engineering-architecture-pinout-plc/