Tools of the Hardware Hacking Trade

Joe Grand
Grand Idea Studio, Inc.

GRAND IDEA STUDIO

Finding the Right Tools for the Job

- Tools can help for design or "undesign"
- Access to tools is no longer a hurdle
- Can outsource to those with capabilities/equipment you don't have
- The key is knowing what tools are available and which one(s) are needed for a particular goal/attack

GRAND IDEA STUDIO

Tool Sets

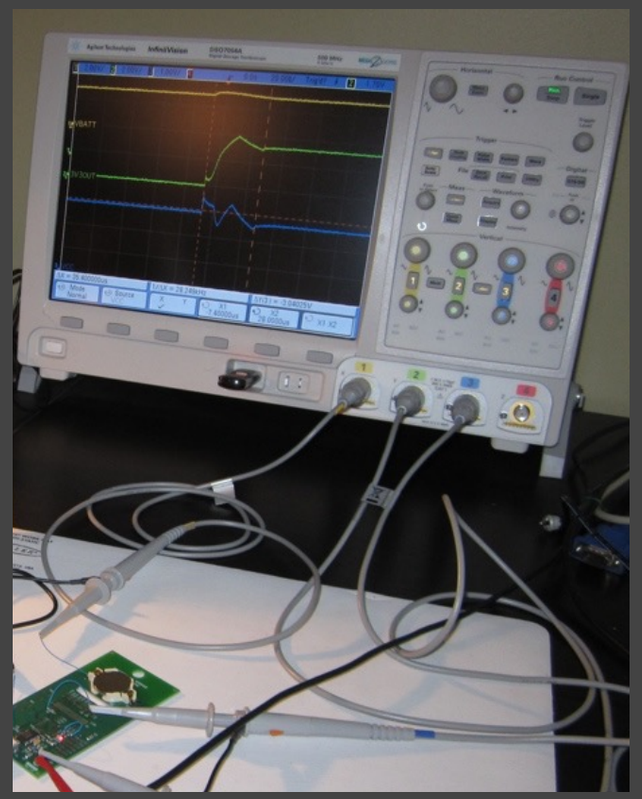
- Signal Monitoring/Analysis
- Manipulation/Injection
- Imaging

GRAND IDEA STUDIO

Signal Monitoring / Analysis

Oscilloscope

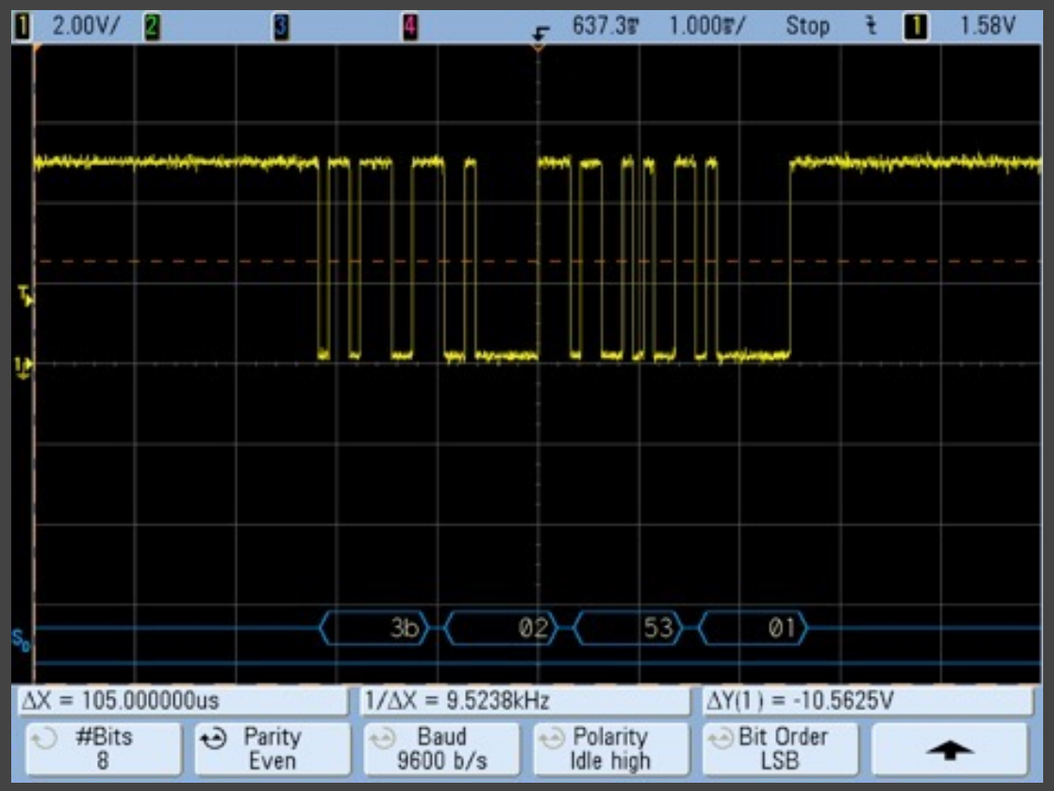
- Provides a visual display of electrical signals and how they change over time
- Range of capabilities/features
  - Analog/digital/mixed signal, # of channels (~1-8), bandwidth, sampling rate, resolution, buffer memory, trigger capabilities, math functions, protocol decoding, probe types
- Standalone: HP/Agilent/Keysight, Tektronix, Rohde & Schwarz, Teledyne LeCroy, Rigol
- PC-based: PicoScope, USBee, BitScope



Oscilloscope: Example

- SFMTA Smart Parking Meter (2009)
  - Joe Grand, Chris Tarnovsky, Jake Appelbaum
  - Monitored meter/card communication w/ oscilloscope
    - Slight variation in signal voltage determined direction of data
  - Created custom Microchip PIC-based smartcard emulator

– www.grandideastudio.com/portfolio/smart-parking-meters

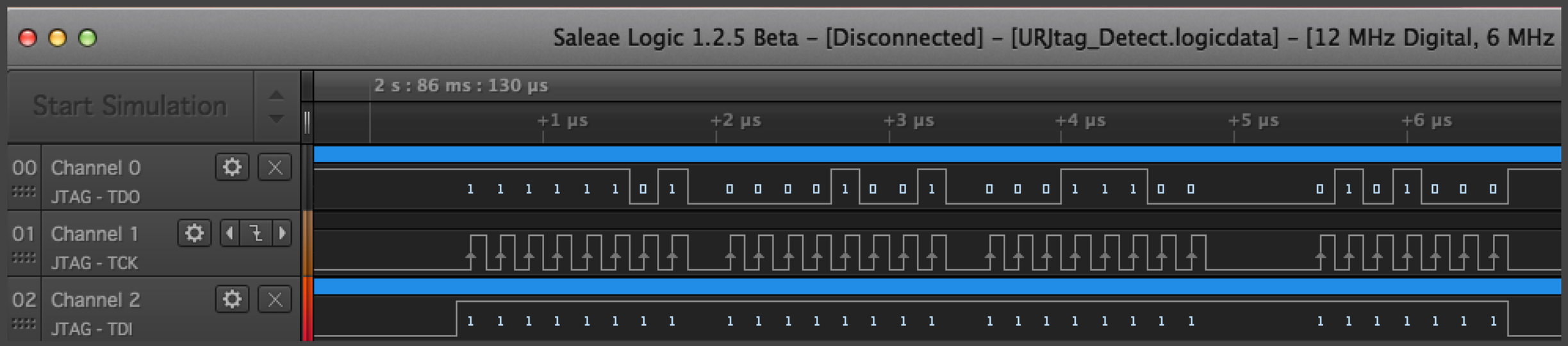


GRAND IDEA STUDIO

Oscilloscope: Example 2

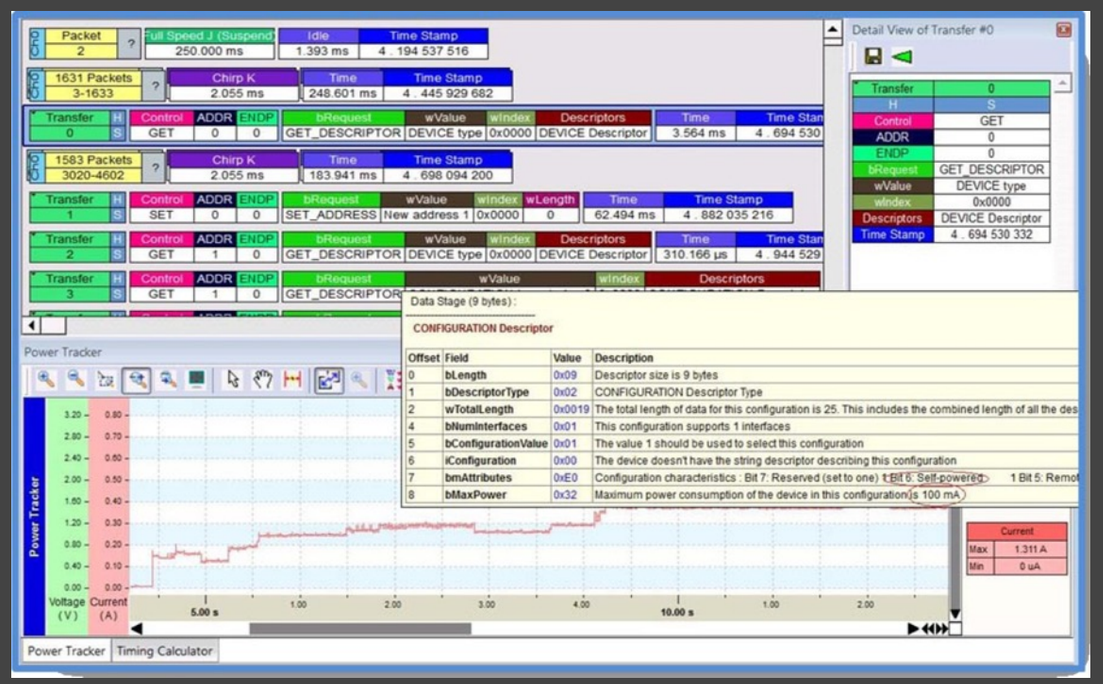
Logic Analyzer

- Used for concurrently capturing, visualizing, and decoding large quantities of digital data
  - # of channels (~>4), sampling rate, buffer memory, trigger capabilities, protocol decoding, probe types, accessories
- Standalone: HP/Agilent/Keysight, Tektronix
- PC-based: Saleae Logic, LogicPort, USBee, LeCroy LogicStudio, DigiView
- Open: sigrok, Open Bench Logic Sniffer



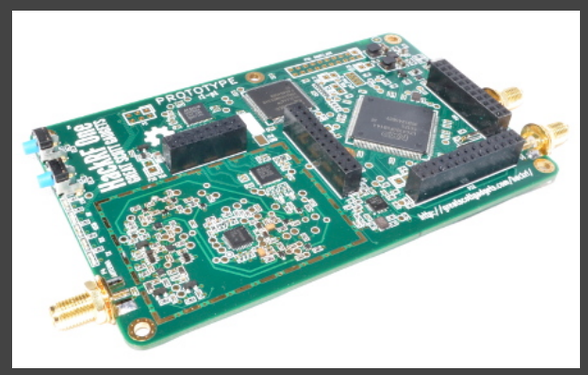
Protocol Analyzer

- Real-time, non-intrusive monitoring/capturing/decoding of complex interfaces
  - HW "man in the middle" to avoid any OS/SW overhead on host
  - Some also support data injection, power measurements
- Total Phase Beagle (USB/I2C/SPI), Komodo (CAN), LeCroy Voyager (USB 2.0/3.0), International Test Instruments (USB 2.0, PCIe 1.1), Finisar Bus Doctor (Modular)
- Open: OpenVizsla, Daisho



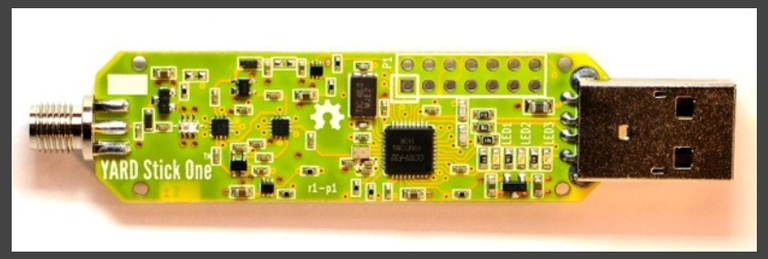
Software Defined Radio

- Communication system where digital signal processing is used to implement radio/RF functions
  - Ex.: Mixers, filters, amplifiers, modulators/demodulators, detectors
  - RF front end + general purpose computer to receive/transmit arbitrary radio signals
- Primary toolset for RF/radio hacking
  - Visualize RF spectrum (spectrum analyzer)
  - Modulate/demodulate/filter raw signal
  - Decode/inject data
- Ex.: RTL-SDR, HackRF One, Blade RF, Ettus Research, LimeSDR



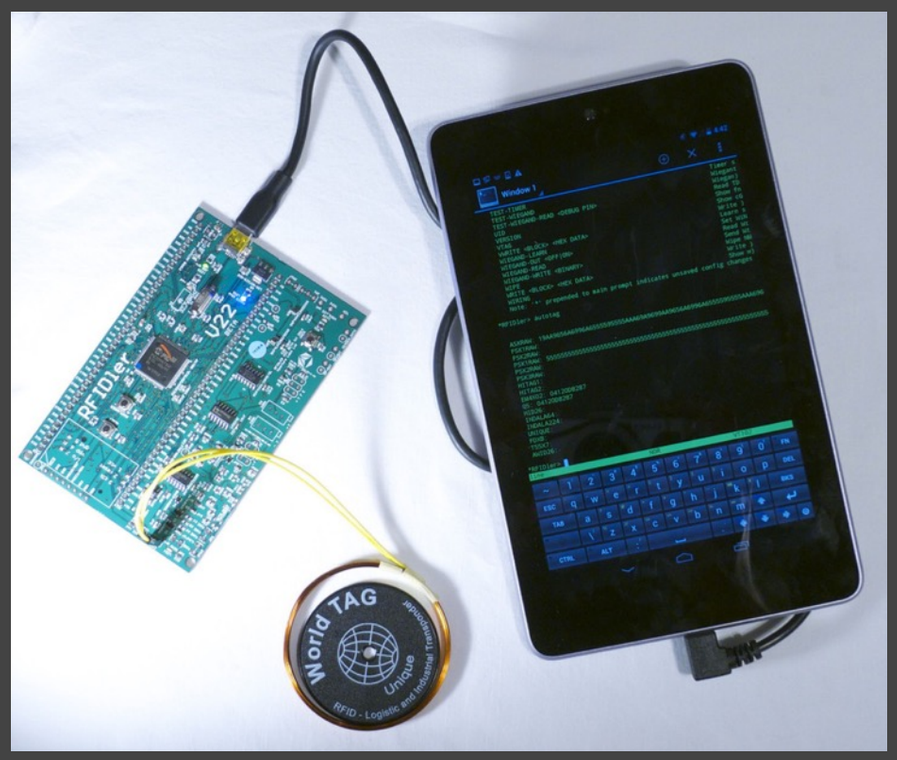
More Wireless

- WiFi Pineapple
  - Penetration testing/attacks
- Femtocell
  - Cellular data interception
- YARD Stick One
  - General purpose RF, < 1GHz
- Ubertooth One
  - Bluetooth/2.4GHz
- Bluefruit LE Sniffer (BLE 4.0)
  - Nordic nRF Sniffer firmware + Wireshark



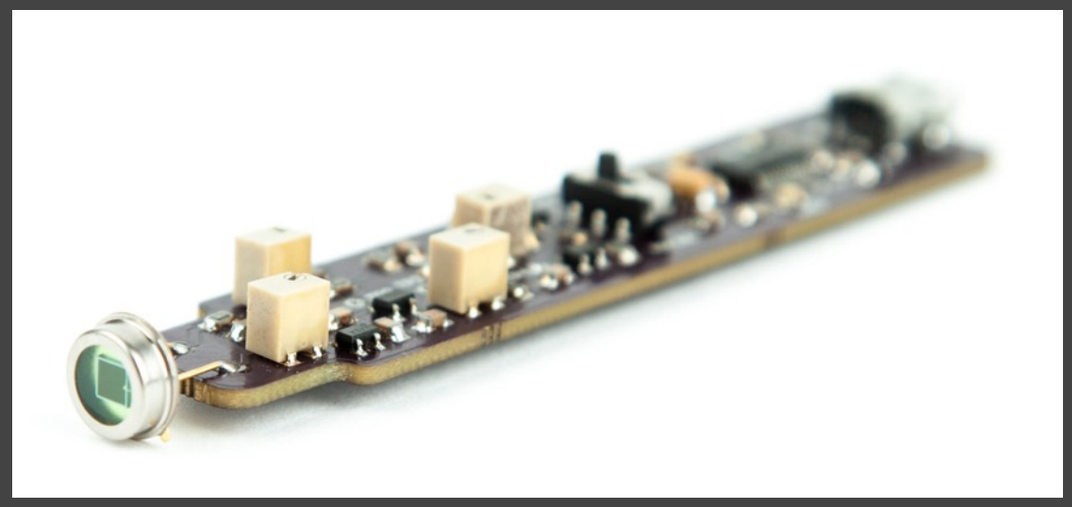
More Wireless 2

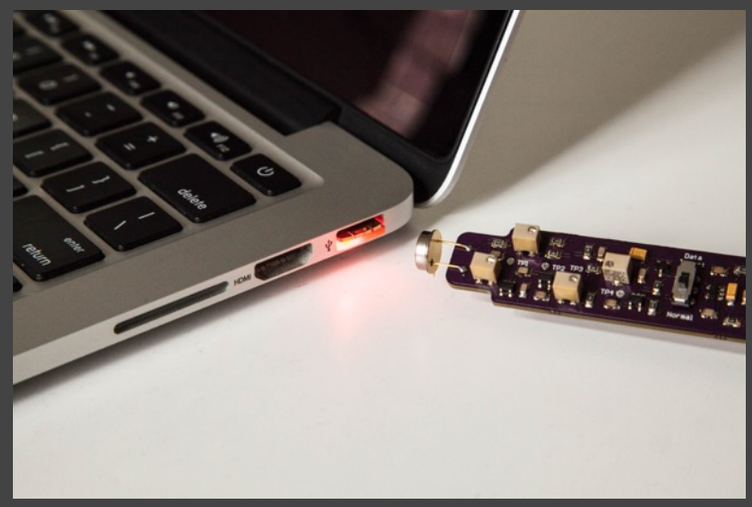
- RFIDiot, RFIDler, Proxmark3
  - RFID/NFC reading/writing/emulation
- RaspBee
  - ZigBee module for Raspberry Pi
  - Command injection via custom firmware
- EZ-Wave
  - Z-Wave and Z-Wave Plus
  - Discover/interrogate/sniff
- gr-lora
  - SDR implementation of physical layer LoRa



OpticSpy

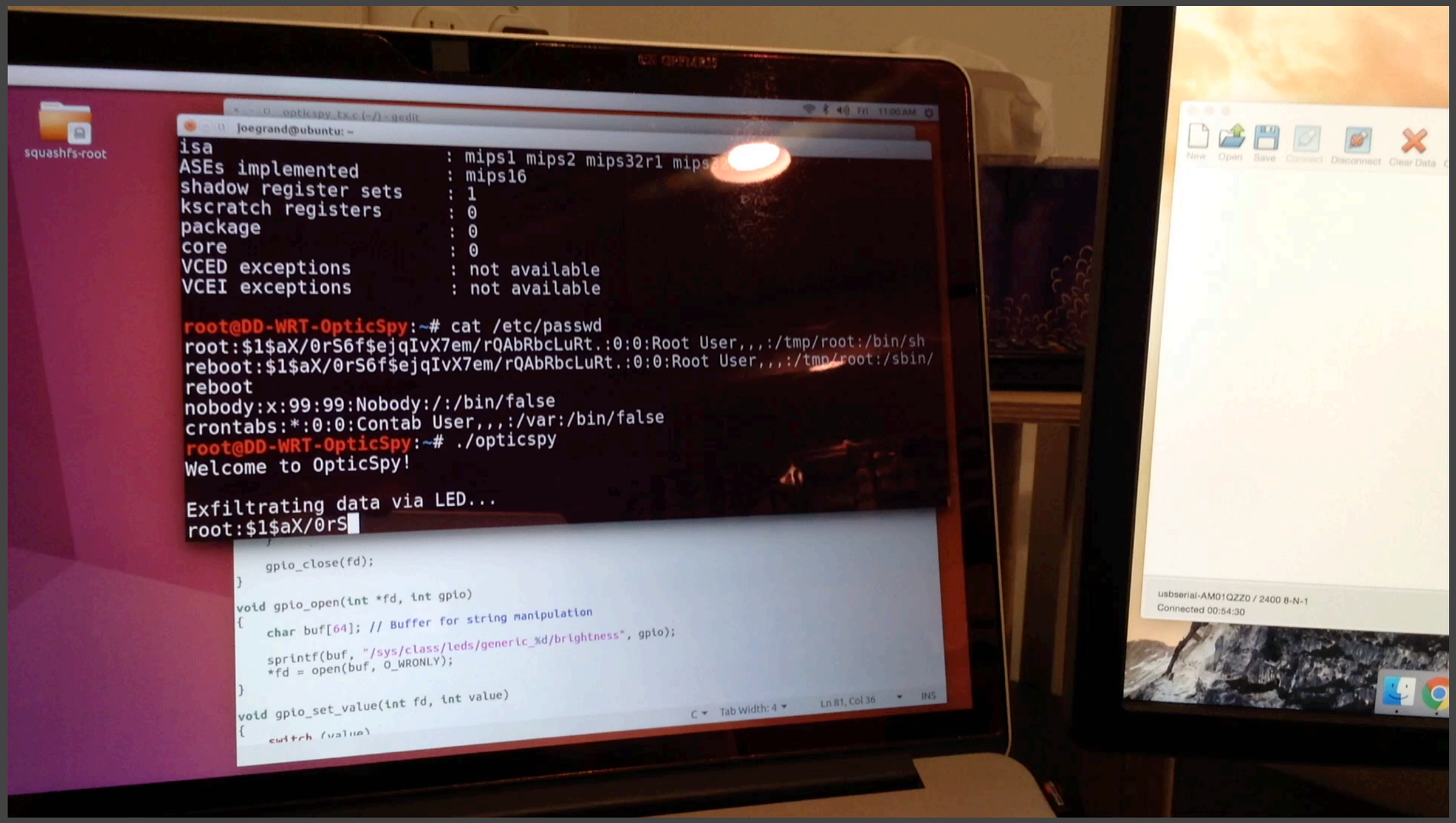
- Open source optical receiver module
  - Converts light into voltage
  - Gain and threshold adjustment via potentiometers
  - USB interface for direct connection to host PC
- Designed primarily for optoelectronic experimentation
  - Detect optical covert channels in existing devices
  - Create air-gapped data transfer functionality
  - Measure the world around you





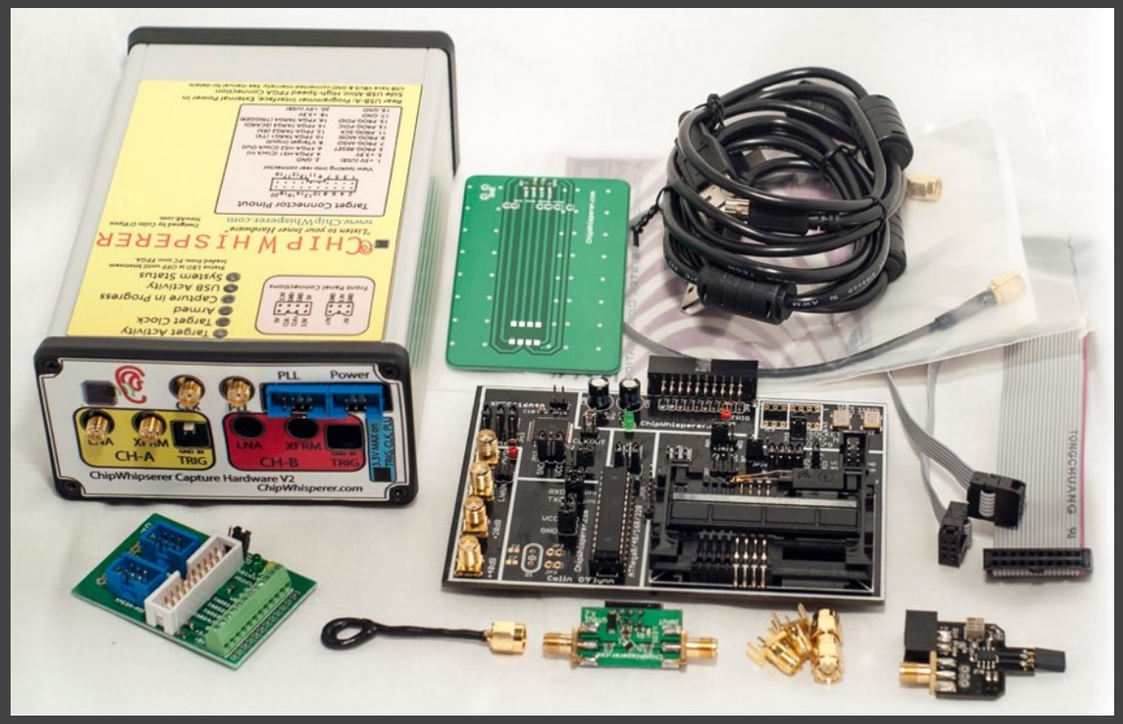
OpticSpy: Example

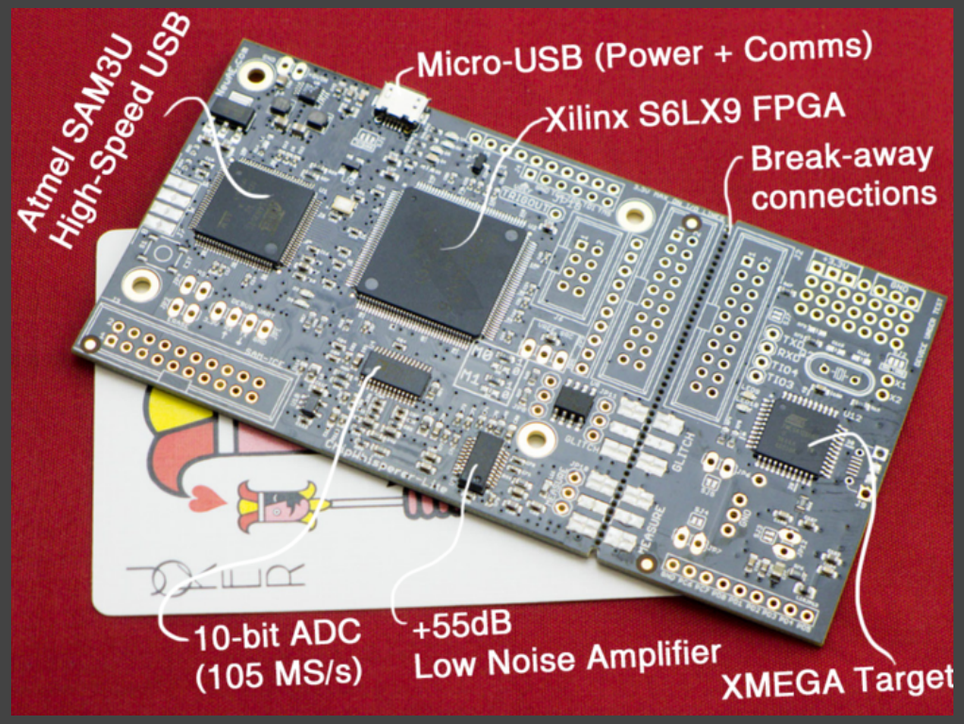
- TP-Link TL-WR841N (proof of concept)



ChipWhisperer (and -Lite)

- Colin O'Flynn
- Collection of open source HW/SW tools for side channel, timing, and glitching attacks
- Supports AES-128/256 key extraction via EM/power analysis
  - Correlate measured power w/ predicted power to guess byte of key
- **www.chipwhisperer.com**



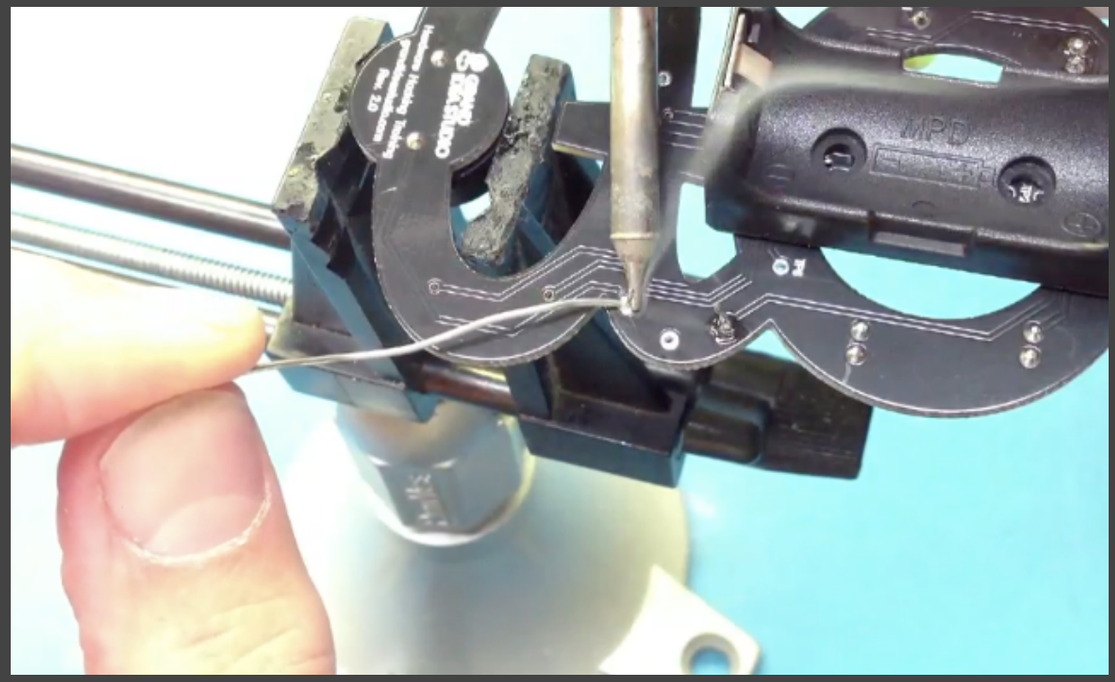


GRAND IDEA STUDIO

Manipulation / Injection

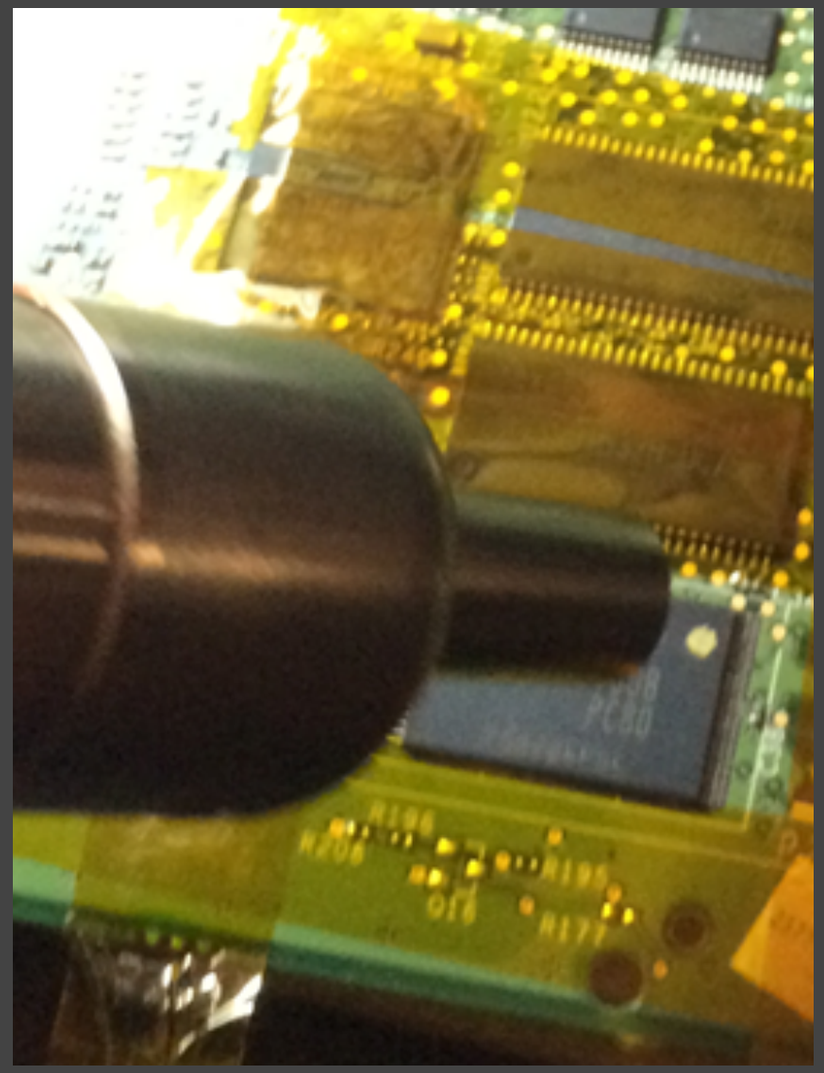
Soldering Iron

- Provides heat to melt solder that physically holds components on a circuit board
- Range from a simple stick iron to a full-fledged rework station
  - Interchangeable tips, adjustable temperature, hot air reflow
- Weller, Metcal, Hakko, Aoyue, Maplin



Rework Station

- Allows easier removal and reflow of individual SMD components (aka "chip off")
- Hot air convection
  - Most accessible, cost effective
  - Nozzles for different package types/ mechanical footprints
  - Difficult to focus heat on just the target component

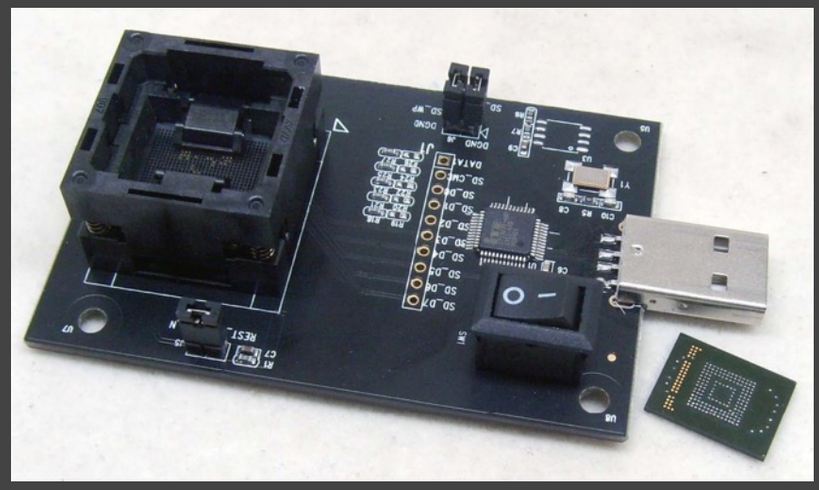


Rework Station 2

- Infrared
  - More complex, expensive systems
  - Provides focused heat on specific component
  - Many are programmable for various heating profiles
- Beware of repeated thermal cycling, which could damage IC
- Ex.: Weller, Metcal, Hakko, ZEVAC, Zephyrtronics

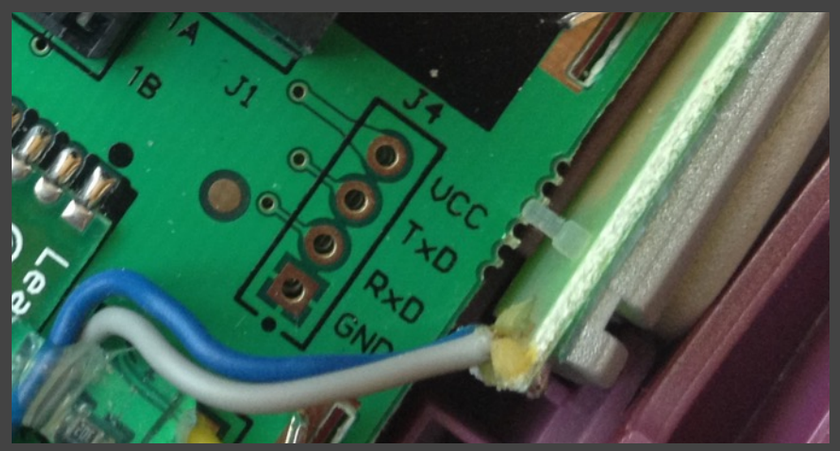
Device Programmer

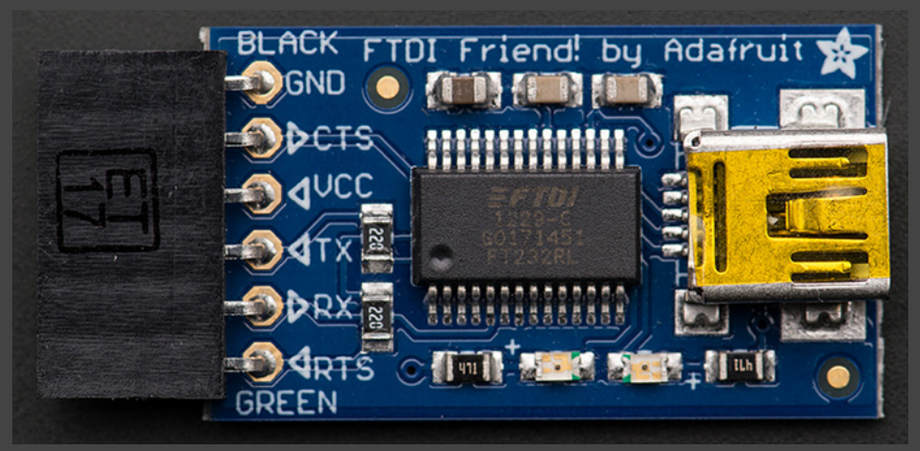
- Used to read/write most devices that contain memory
  - Standalone or internal to MCU
  - Ex.: Flash, E(E)PROM, ROM, RAM, PLD/CPLD, FPGA
- Many support > 100k (!) different devices
- Some devices can be manipulated in-circuit
- Code protection mechanisms exist, may be bypassed
  - Security bit/fuse, PIN/password
- Xeltek, EE Tools, BP Microsystems, DediProg, MiniPro



USB-to-Serial Adapter

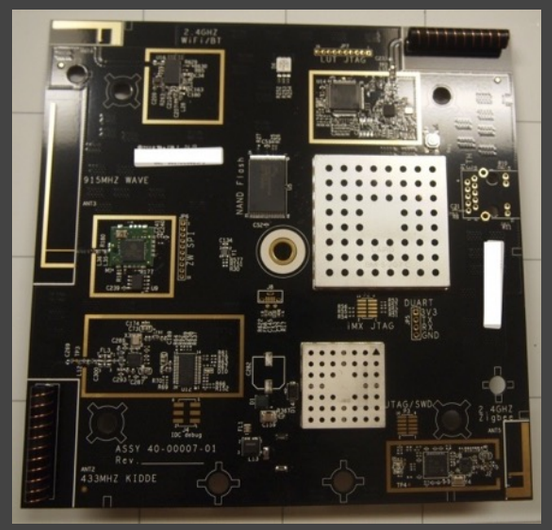
- Many embedded systems use UART as a console
  - Boot log, debug output, bootloader menu, recovery interface, login prompt, root shell
- Converts logic level asynchronous serial to Virtual COM Port
  - → TXD = Transmit data (to target device)
  - ← RXD = Receive data (from target device)
  - ↔ DTR, DSR, RTS, CTS, RI, DCD = Control signals (often unused)
- Easily connects to PC, Mac, Linux w/ suitable drivers





USB-to-Serial Adapter: Example

- Wink Hub
- Force bootloader into interactive shell and modify parameters to get root access
  - Pull NAND I/O 0 (pin 29) to GND while kernel loads
  - Image will fail to read properly, dropping user into shell
  - Modify boot arguments and re-start kernel boot process
  - **http://exploitee.rs/index.php/Wink_Hub%E2%80%8B%E2%80%8B**

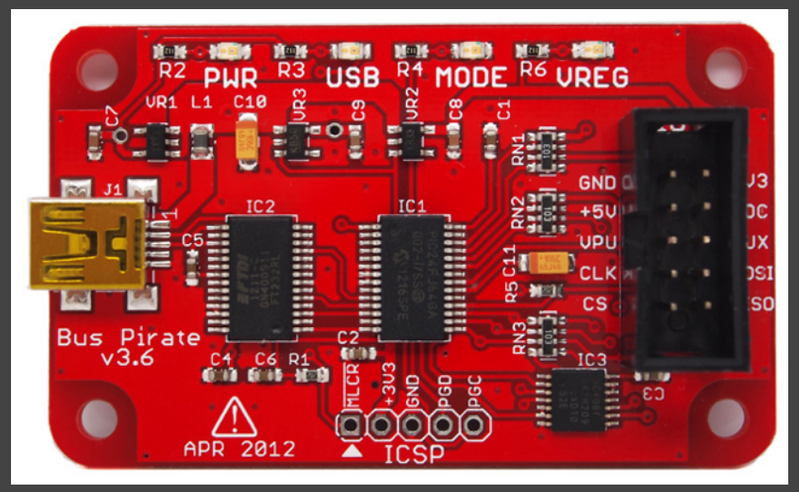


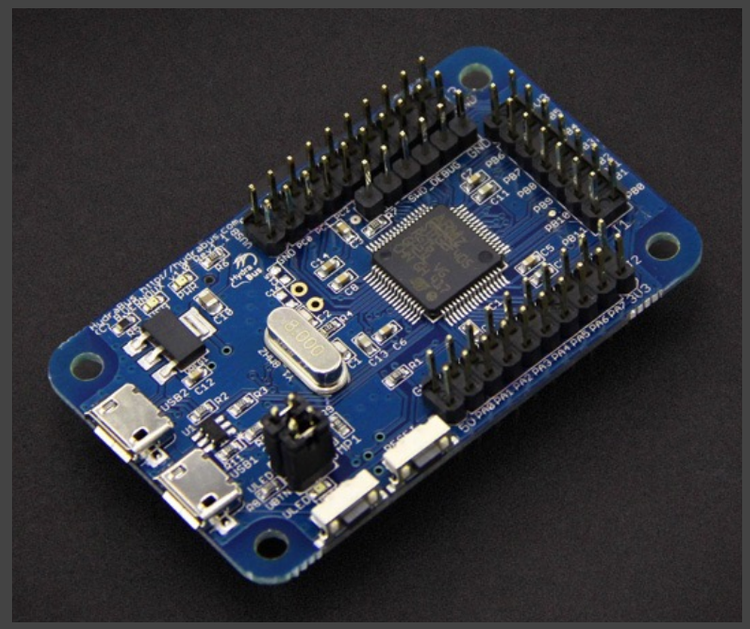
USB-to-Serial Adapter: Example 2

```
Falling back to updater...

NAND read: device 0 offset 0x300000, size 0x400000
NAND read from offset 300000 failed -74
 0 bytes read: ERROR

NAND read: device 0 offset 0x2b00000, size 0x400000
NAND read from offset 2b00000 failed -74
 0 bytes read: ERROR
Wrong Image Format for bootm command
ERROR: can't get kernel image!
=> ◆ri◆◆◆◆◆◆
=> printenv
app_boot=run appboot_args && nand read ${loadaddr} app-kernel 0x00400000 && bootm ${loadaddr}
app_boot_bad=run updater_args; setenv bootargs ${bootargs} badapp; nand read ${loadaddr} updater-kernel 0x00400000; bootm ${loadaddr}
appboot_args=setenv bootargs 'noinitrd console=ttyAM0,115200 rootfstype=ubifs ubi.mtd=5 root=ubi0:rootfs rw gpmi';
baudrate=115200
bd_addr=0021CC06D0EB
boot_app=run app_boot || run app_boot_bad
boot_getflag=mtdparts default && ubi part database && ubifsmount ubi0:database && mw 42000000 0 8 && ubifsload 42000000 DO_UPDATE 1 && run boot_logic
boot_logic=mw 42000004 30; if cmp 42000000 42000004 1; then run boot_app; else run boot_updater; fi;
boot_updater=run updater_boot || run updater_boot_bad
bootargs=noinitrd console=ttyAM0,115200 rootfstype=ubifs ubi.mtd=5 root=ubi0:rootfs rw gpmi badupdater
bootcmd=mtdparts default; run boot_getflag || echo Falling back to updater...; run boot_updater
bootdelay=0
bootfile=uImage
ethact=FEC0
ethaddr=00:04:00:00:00:00
ethprime=FEC0
loadaddr=0x42000000
mtddevname=u-boot
mtddevnum=0
mtdids=nand0=gpmi-nand
mtdparts=mtdparts=gpmi-nand:3m(u-boot),4m(updater-kernel),28m(updater-rootfs),8m(database),8m(app-kernel),-(app-rootfs)
partition=nand0,0
serialno=142301503WZD1
stderr=serial
stdin=serial
stdout=serial
updater_args=setenv bootargs 'noinitrd console=ttyAM0,115200 rootfstype=ubifs ubi.mtd=2 root=ubi0:rootfs rw gpmi';
updater_boot=run updater_args && nand read ${loadaddr} updater-kernel 0x00400000 && bootm ${loadaddr}
updater_boot_bad=run appboot_args; setenv bootargs ${bootargs} badupdater; nand read ${loadaddr} app-kernel 0x00400000; bootm ${loadaddr}
ver=U-Boot 2014.01-14400-gda781c6-dirty (Apr 30 2014 - 22:35:38)

Environment size: 1762/16379 bytes
=> setenv bootargs 'noinitrd console=ttyAM0,115200 rootfstype=ubifs ubi.mtd=5 root=ubi0:rootfs rw gpmi init=/bin/sh';
=> nand read ${loadaddr} app-kernel 0x00400000 && bootm ${loadaddr}
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB0
```

Multi-Tools: Bus Pirate, HydraBus, Good/GreatFET

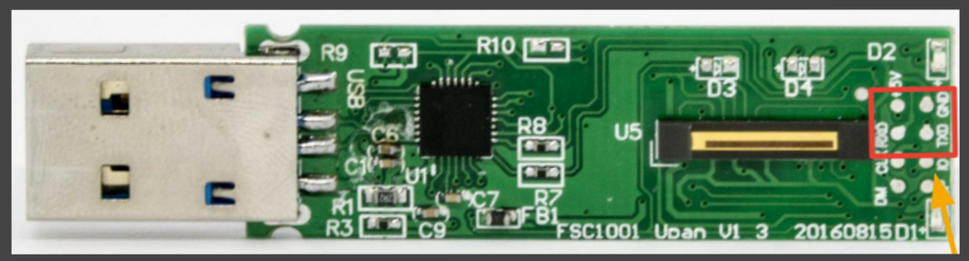
- Open source, general purpose hacking platforms
  - SPI, I2C, 1-Wire, CAN, LCD, JTAG, USB, MCU/FPGA/memory programming, bit bang, scriptable, etc.
  - Sniffing, injection, logic analyzer/digital decoding

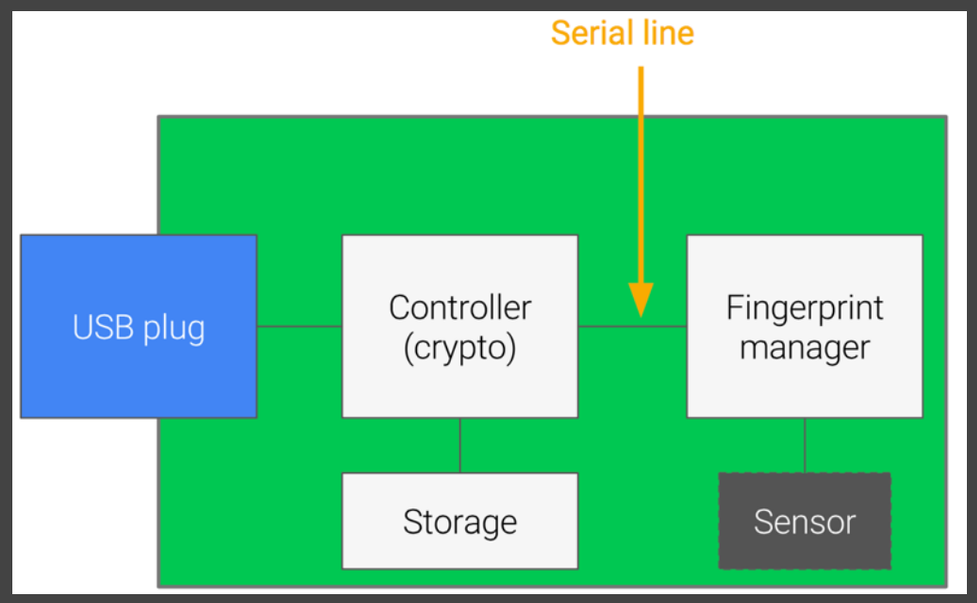




Multi-Tools: Example

- Biometric encrypted thumb drive
  - Picod, Audebert, Blumenstein, Bursztein, BH USA 2017
  - Serial interface between fingerprint scanner & MCU
  - Sniff and replay command to unlock device
  - **https://cdn.elie.net/talk/attacking-encrypted-usb-keys-the-hardware-way**



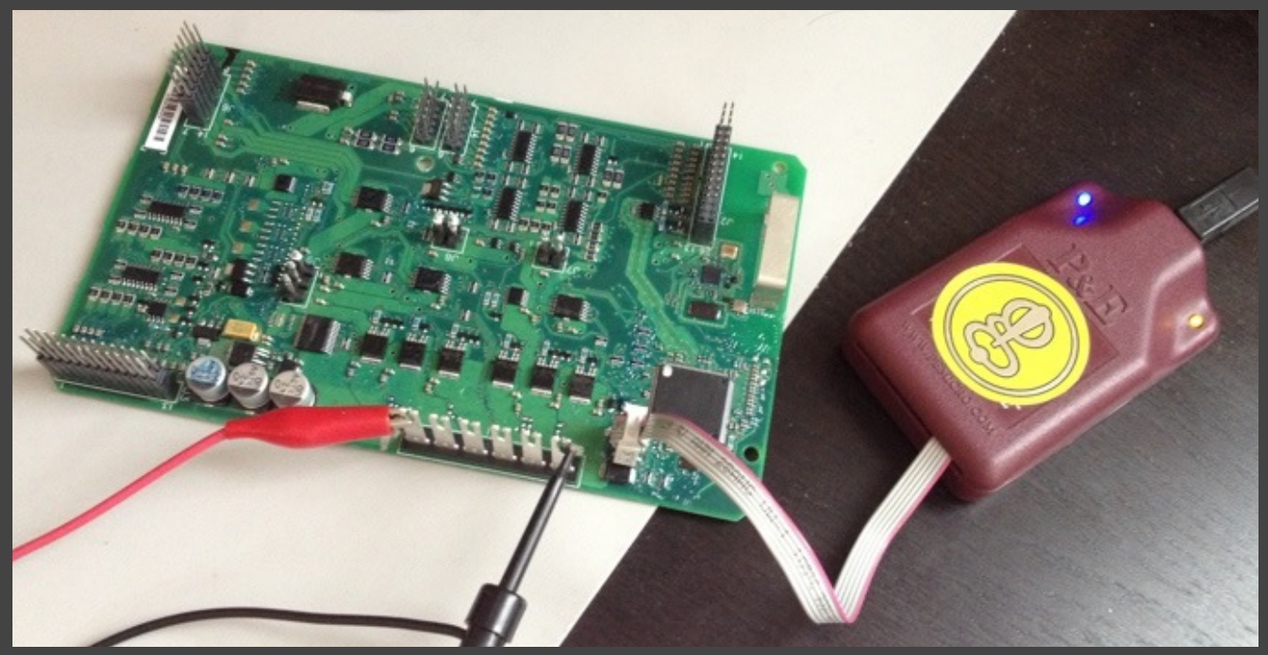


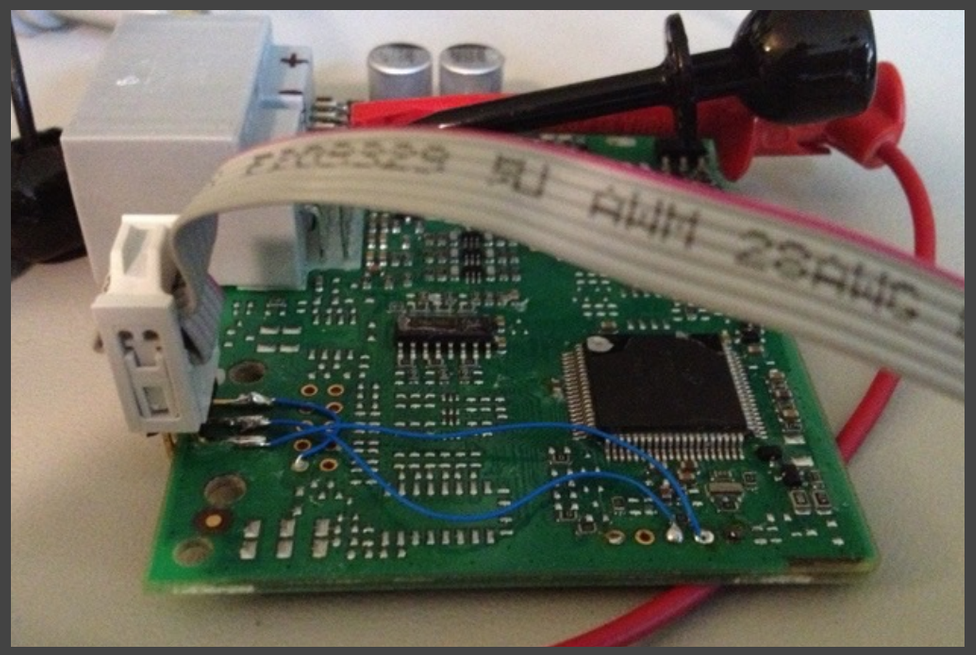
Debug Tools

- Off-the-shelf HW tools designed for interaction w/ target device
  - Can provide chip-level control (single step, access registers)
  - Modify memory contents
  - Extract program code or data
  - Affect device operation on-the-fly
- Either vendor-specific or industry standard (JTAG)
- Many different types available
  - Ensure tool supports your target architecture
  - Find out what vendor recommends for legitimate engineers

Debug Tools: Example

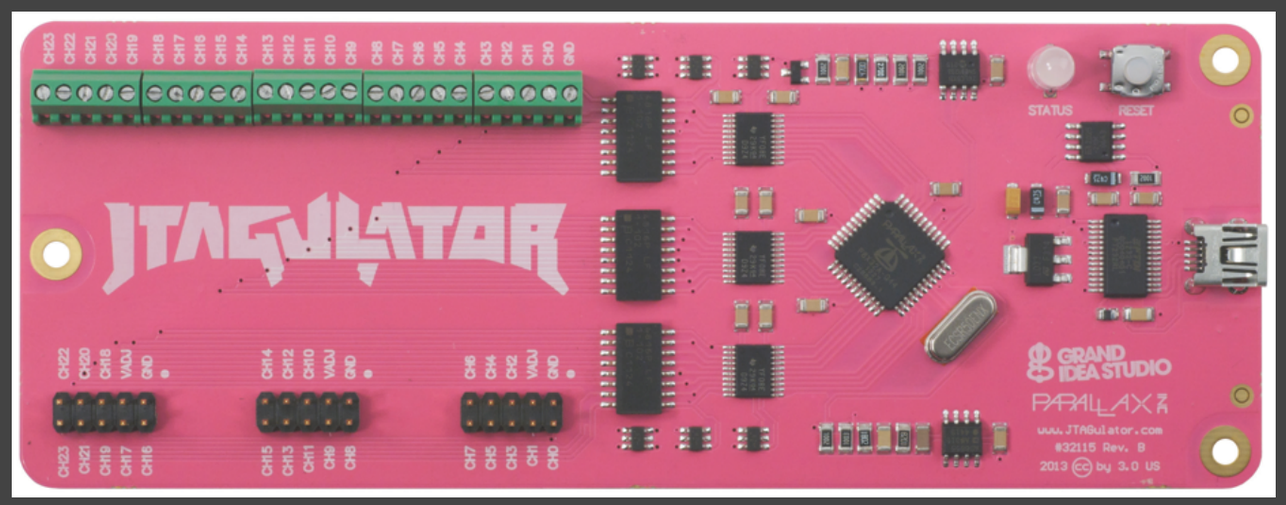
- Ford Electronic Control Units (ECUs) (2013)
  - For Charlie Miller & Chris Valasek
  - Complete firmware extraction led to understanding typical CAN traffic/functionality and arbitrary code execution
  - **`http://illmatics.com/car_hacking.pdf`**
  - Used standard, off-the-shelf development tools
    - Freescale CodeWarrior for S12(X) v5.1 + P&E Multilink USB Rev. C

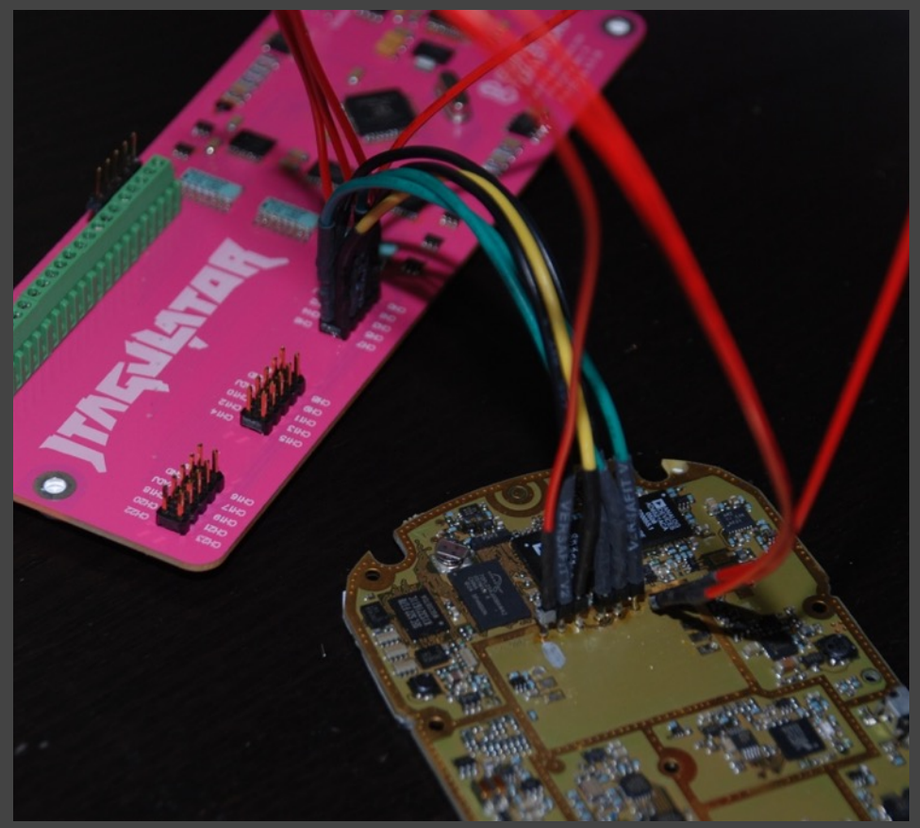




Debug Tools: JTAGulator

- Open source tool to assist with discovery of on-chip program/ debug interfaces
- Currently detects JTAG & UART/asynchronous serial
- Supports up to 24 connections to unknown points on target circuit board, adjustable target voltage (1.2V-3.3V), input protection, firmware upgradable





GRAND IDEA STUDIO

Debug Tools: JTAG HW/SW

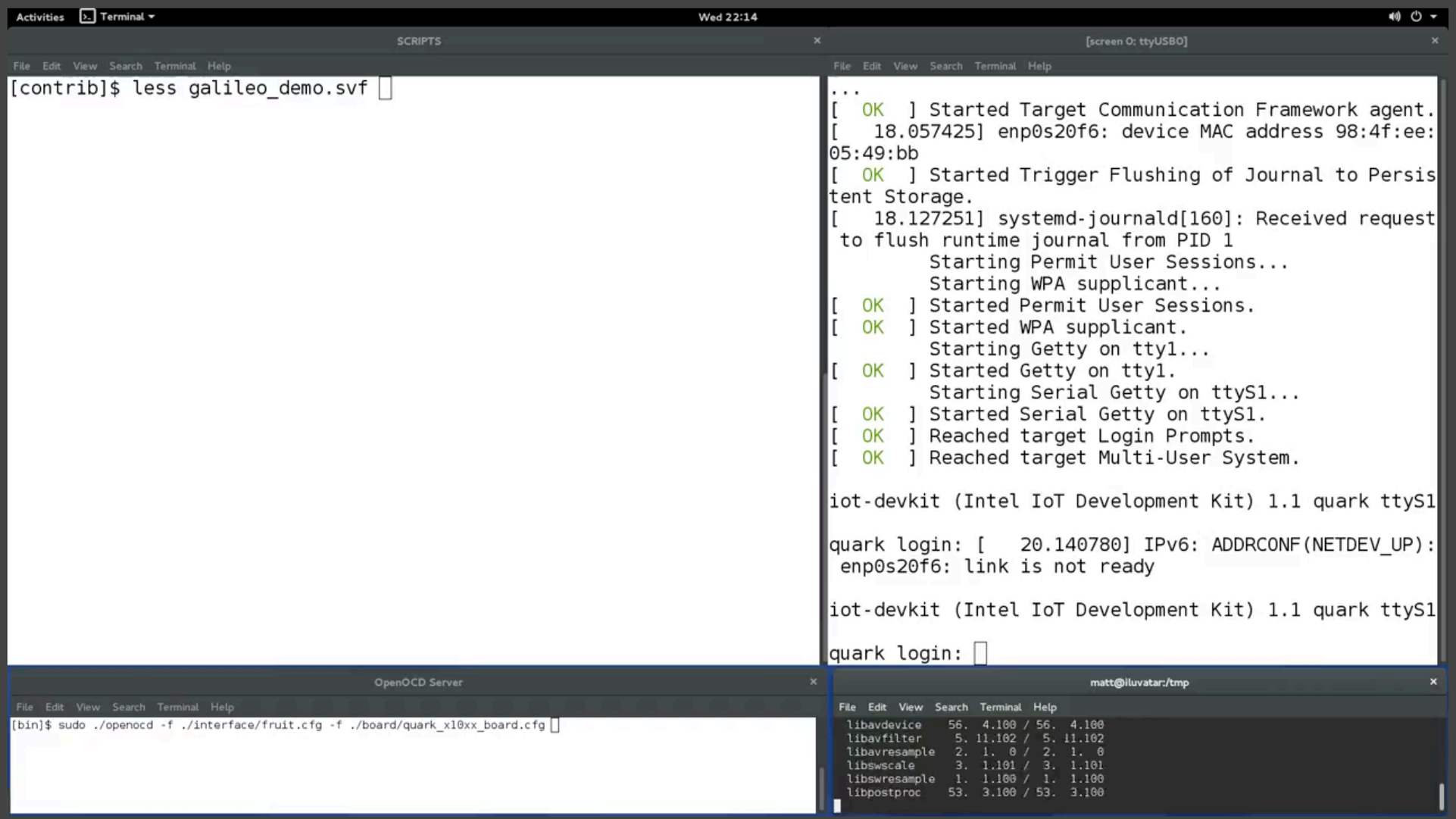
- Bus Blaster
- FT232H Breakout Board
- Black Magic Probe
- SEGGER J-Link

- OpenOCD
- UrJTAG


```
OpenOCD
Open On-Chip Debugger 0.6.0 (2012-09-07-10:44)
Licensed under GNU GPL v2
For bug reports, read
        http://openocd.sourceforge.net/doc/doxygen/bugs.html
adapter speed: 1000 kHz
srst_only separate srst_nogate srst_open_drain
Info : clock speed 1000 kHz
Info : stm32f0x.cpu: hardware has 4 breakpoints, 2 watchpoints
Info : accepting 'gdb' connection from 3333
Info : device id = 0x20006440
Info : flash size = 64kbytes
Warn : acknowledgment received, but no packet pending
undefined debug reason 6 - target needs reset
target state: halted
target halted due to debug-request, current mode: Thread
xPSR: 0xc1000000 pc: 0x08000124 msp: 0x20002000
target state: halted
target halted due to breakpoint, current mode: Thread
xPSR: 0x61000000 pc: 0x2000003a msp: 0x20002000
```

Debug Tools: JTAG Example

- Linux ACL Check Patch
  - JTAG to Root: 5 Ways, FitzPatrick & King, BSides PDX 2015
  - **`https://github.com/syncsrc/jtagsploitation`**

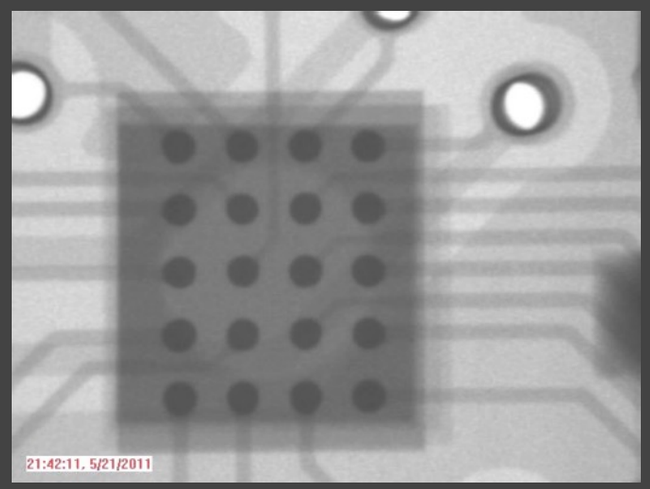


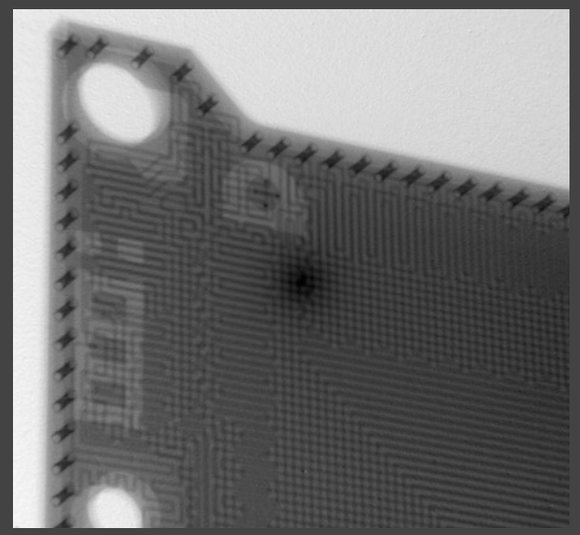
GRAND IDEA STUDIO

Imaging

X-Ray (2D)

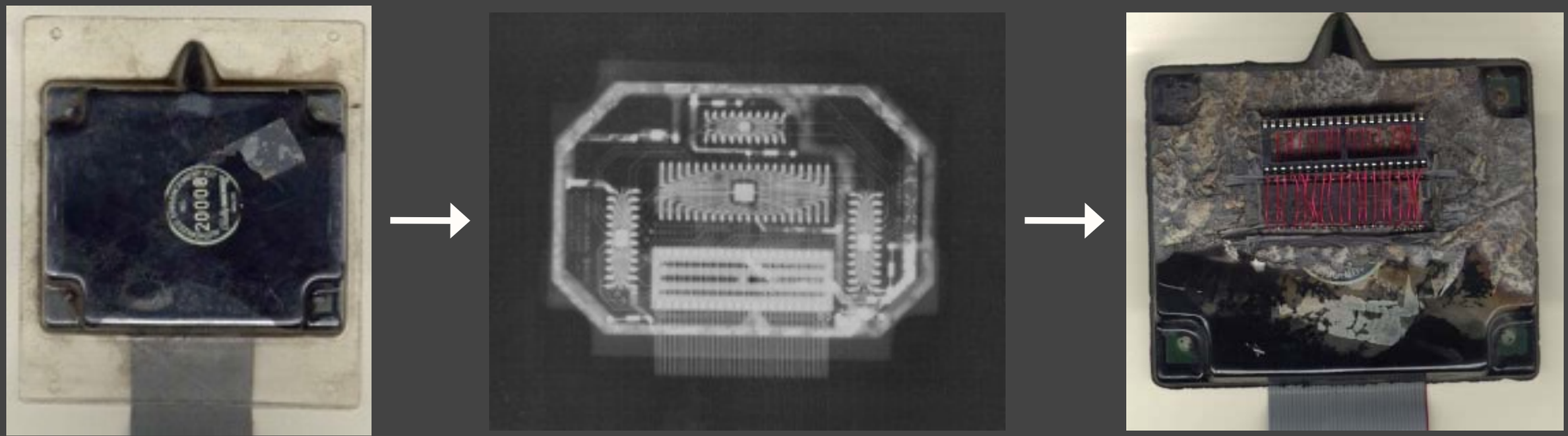
- X-rays passed through target and received on detector
  - All materials absorb radiation differently depending on density, atomic number, and thickness
- Provides a composite image of all layers in target
  - Quality inspection, failure analysis, fabrication/assembly techniques, component location, hidden/embedded features, defeating encapsulation





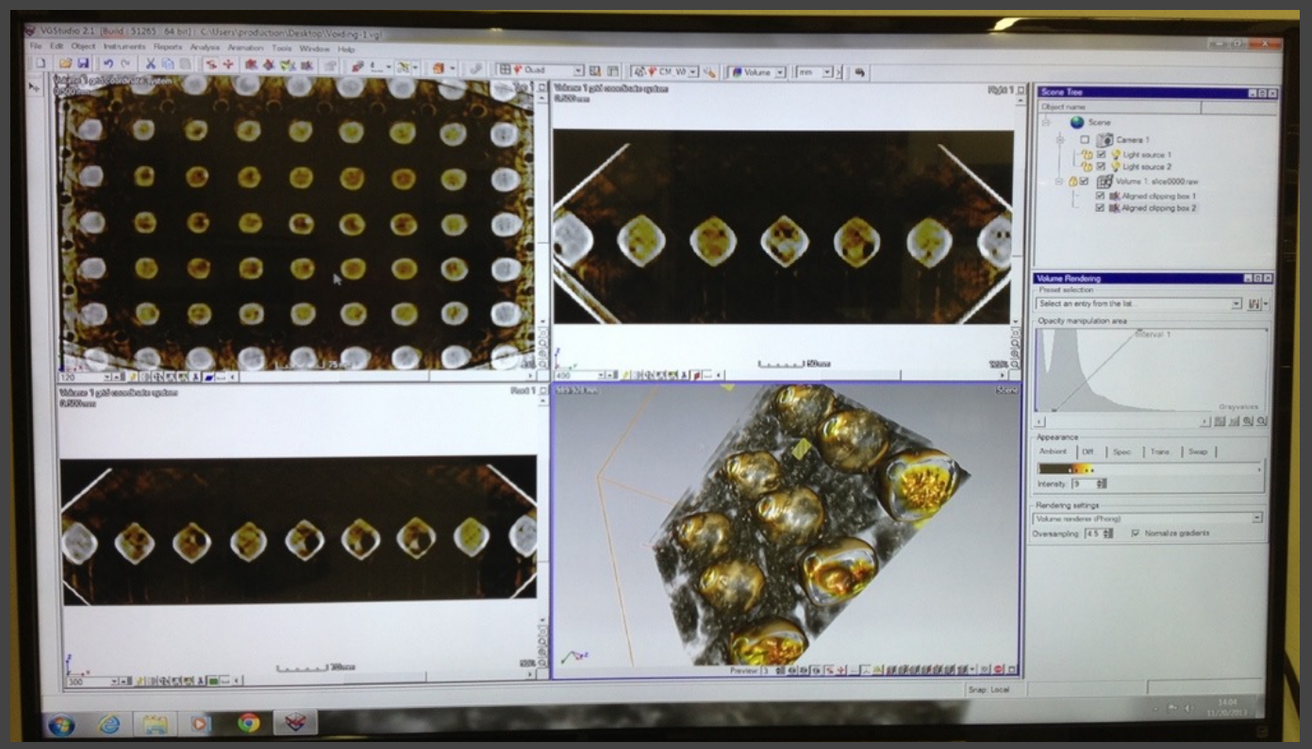
X-Ray (2D): Example

- Pac Man Plus Upgrade Module
  - Official conversion kit produced by Bally/Midway, 1982
  - Eventually defeated/reverse engineered by Clay Cowgill, 2000
  - **`www.multigame.com/pacplus.html`**



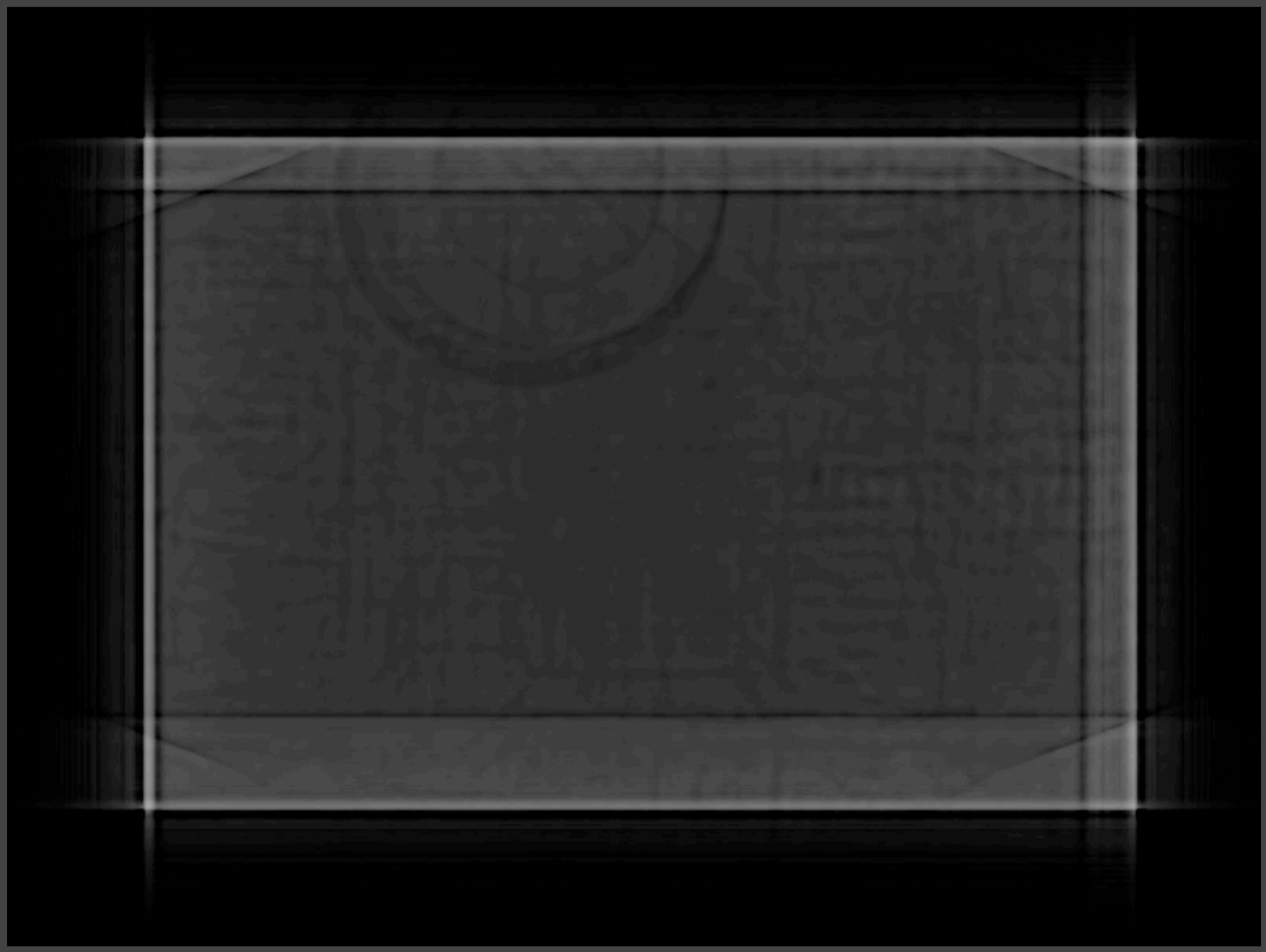
X-Ray (3D/CT)

- Computed Tomography (CT)
  - A series of 2D X-ray images post-processed to create cross-sectional slices of the target
  - X-ray beam rotated 360° in a single axis around the target
  - Post-processing results in 2D slices that can be viewed in any plane
  - Can be manipulated with 3D modeling software



GRAND IDEA STUDIO

X-Ray (3D/CT): Example



GRAND IDEA STUDIO

What Now?

- Create a hardware hacking lab (if you haven't already)
- Keep an eye out for new tools by hackers and industry
- Collaborate with others who may have complementary skills/tools
- Use these tools to validate your product's security or to better understand attack techniques

GRAND IDEA STUDIO

Thanks for your time!