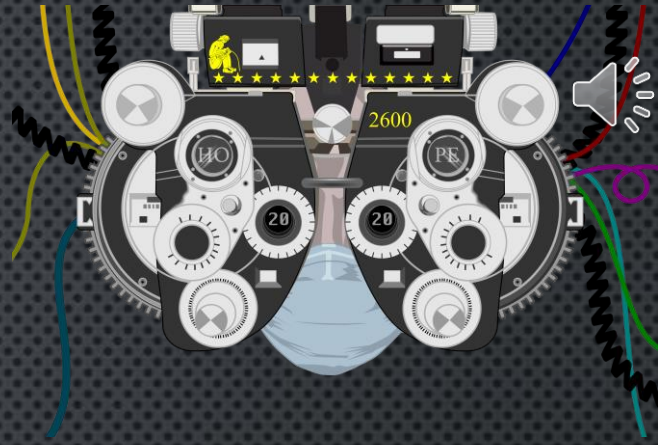
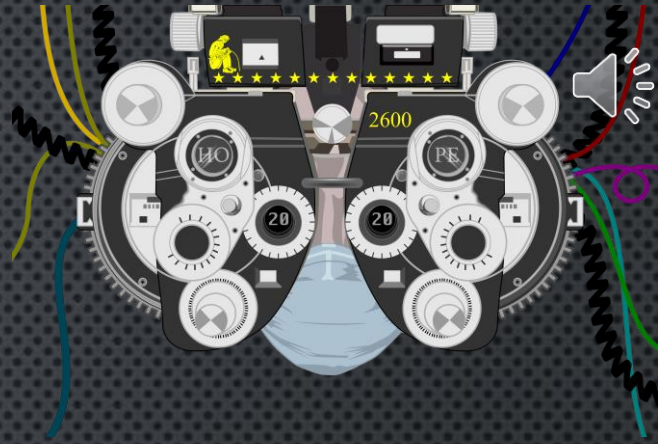
Let’s have a Board Level talk



An summary of various Protocol boards used to access hardware

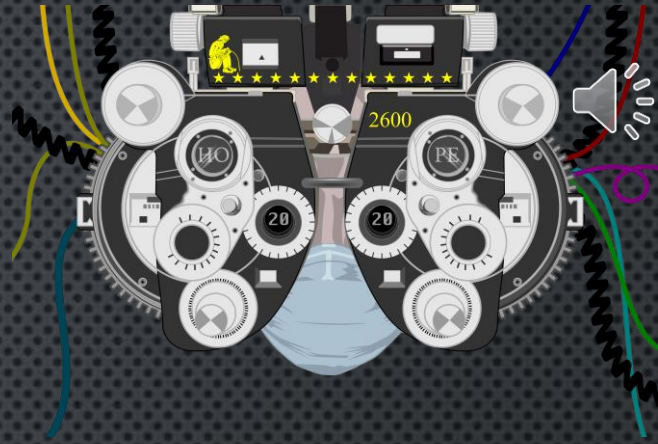
Bruce Barnett
@grymoire

Why Hardware Hacking?



- Access Console on embedded system
- Reflash firmware
- Unbrick a device
- Extract and examine firmware
- Change firmware
- Reverse Engineer systems
- Investigate security of IoT products
- Help improve consumer-device security
- Make your own stuff!

Why should you attend this talk?



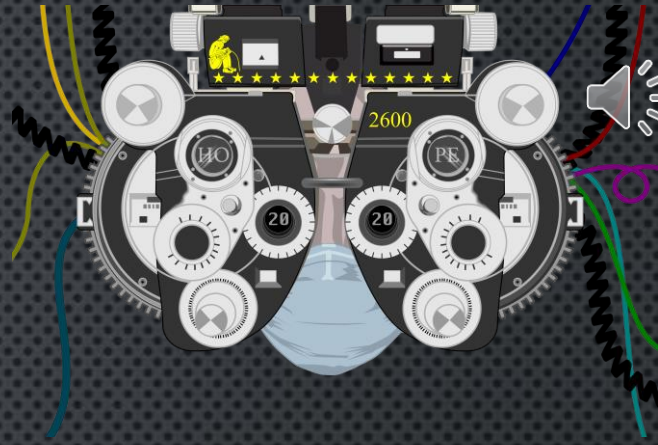
I'm starting out - what should I buy?

- Some are very expensive, and some are cheap
- Some are obsolete
- Some require learning new(?) programming tools/interfaces
- Some are for those with more experience
- Which way to go? What board(s) should I get?

@grymoire

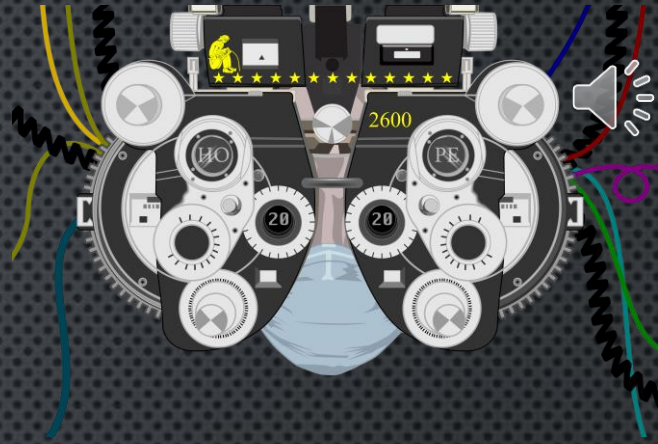
- Popular website on sed/awk
- Electronic Test Systems
- Research Scientist/Security Consultant
- Security Gadabout
  - Network analysis, expert systems (AI), advanced IDS, steganography, cryptography, RFID, SDR
- Magician (DEFCON 19 talk)
- Run Hardware Hacking Village for ANYCon in Albany NY

Basic Toolkit for Embedded System Hacking



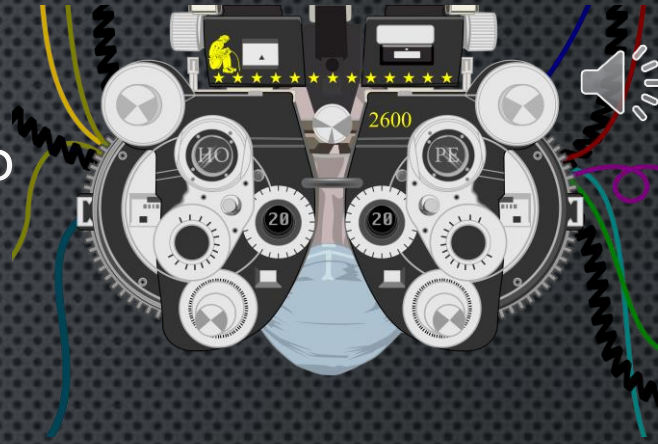
- Soldering gear ($20,$80,$150)
- Breadboard + components ($20,$50,$200)
- Multimeter ($13,$25,$50,$200,…)
- Arduinos, rPi, etc. ($5,$50,…)
- Protocol boards that allows you to interface with embedded electronics ($2,$30,$150,++)
- Magnifying device ($4,$40,$120,..)
- Logic Analyzer ($10,$100,$200,...)
- Oscilloscope ($40,$80,$400,...)

Out of Scope



- Wireless whatever
  - WiFi Hacking
  - Other wireless/Radio protocols
  - Software-Defined Radio
  - RFID
- Physical layer protocols (fiber,Ethernet)
- Car (CAN bus, OBDII)
- Chip-level de-capping/hacking
- Side Channel attacks
  - See ChipWhisperer products by NewAE & Colin O'Flynn

What protocols are used to connect into the system?



Firmware/bootloader Installation (varies, depends on CPU family)

- ISP – In-System Programing
- ICSP – In-Circuit System Programming – i.e. Arduinos

Serial Terminal (UART)

I^2C - Inter-Integrated Circuit

- Multiple devices, one controller
- Used for sensors/LCD Displays

SPI - Serial Peripheral Interface

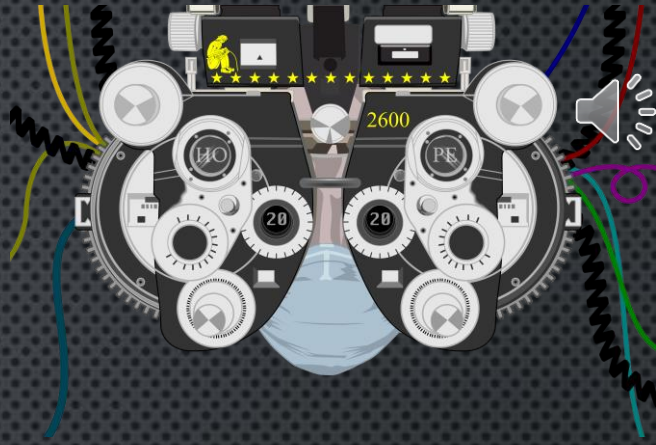
- Fast
- Used for displays, flash memory - read & write

JTAG (Joint Test Action Group)

- Versatile interface, allows in-circuit debug & test

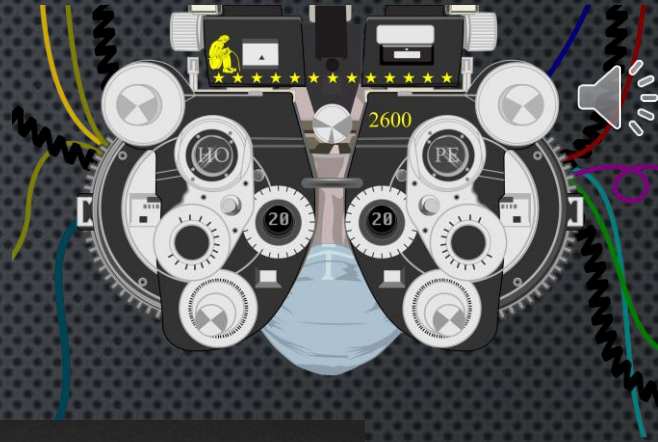
SWD - Single Wire Debug for ARM

USB



LET'S TALK ABOUT THE SIMPLEST SERIAL INTERFACE

UART (Terminal) Interfaces

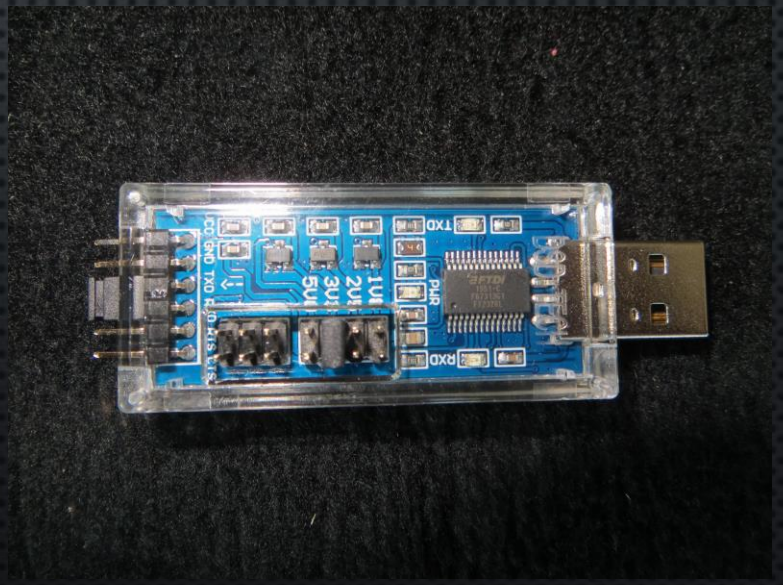


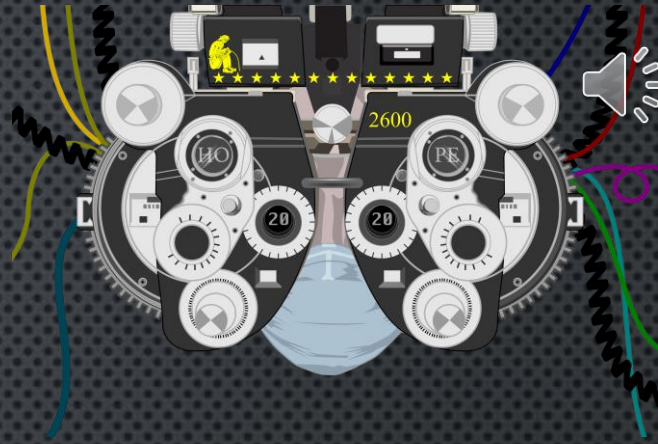
Used for print commands and console interfaces

- FTDI makes popular but expensive chip with free Windows driver
  - 2012 Clones enter the market
  - 2014 FTDI released windows driver that bricked clones (FTDI-gate)

Current products

- FTDI
  - Cable ($10)
  - FTDI Friend ($15 - Adafruit)
  - FT232RL boards (DSD-Tech, $12, 1.8V/2.5V/3.3V/5V)
- Silicon Labs CP210x ($1,$6)
- CH340/CH341 ($1,$10)
  - 3V/5V switchable
- Prolific PL2303 ($1,$10)
- MCP2221 ($15) - also supports I^2C

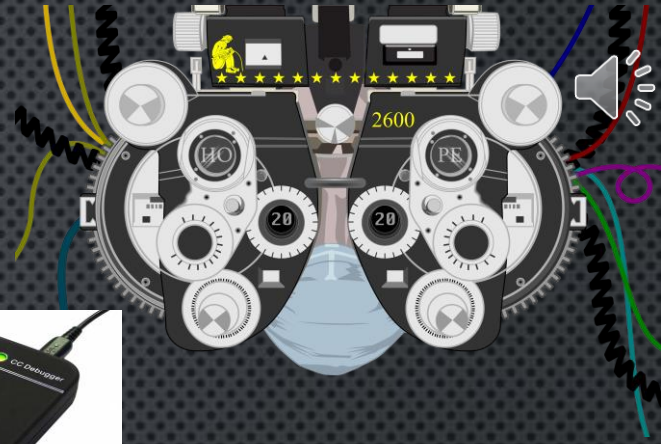




THE THING'S HOLLOW -- IT GOES ON FOREVER -- AND -- OH MY GOD! -- IT'S FULL OF 1’S AND 0’S!

https://commons.wikimedia.org/wiki/File:Stanley_Kubrick_The_Exhibition_-_LACMA_-_2001_A_Space_Odyssey_-_Monolith_(8999726452).jpg

CPU Families and Proprietary programmers/Debuggers



- Texas Instruments MSP430, 8051, CCxxxx
  - CC Debugger ($50), MSP-FET($120),
- 32-bit AVR
  - ATMEL ICE ($130)
- 32-bit PIC
  - J-32 Debug Probe ($190)
- STM32 - STM ST-Link/V2 ($2,$22)
- Microchip MPLabs ICD4, PICKit4, Snap
- XGenu TL866II plus ($50+)
- ARM7/9/11, Cortex-A5/A8/A9, Renesas, Cortex-M0/M0+/M1/M3/M4, Cort
  - Segger J-Link ($20,$60,$600)

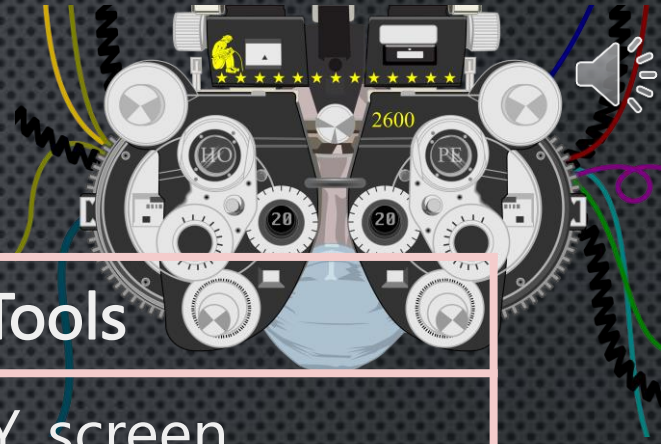
FET == Flash Emulation Tool

Note: there are more than 10,000 different microcontrollers
No single programmer can do it all

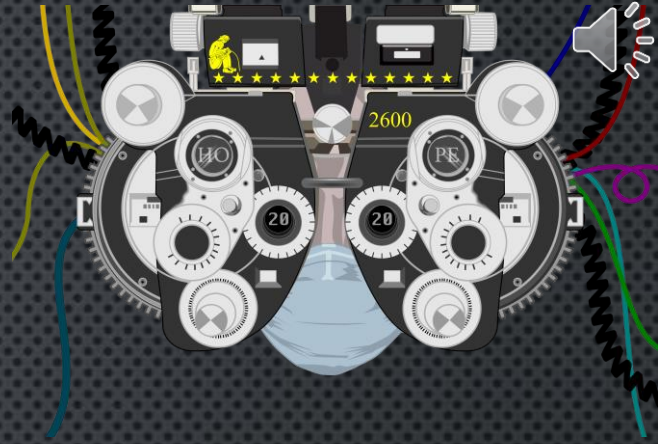
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.
Open source is good for me. Proprietary is evil.

https://www.flickr.com/photos/deia/7942538

Serial Protocols

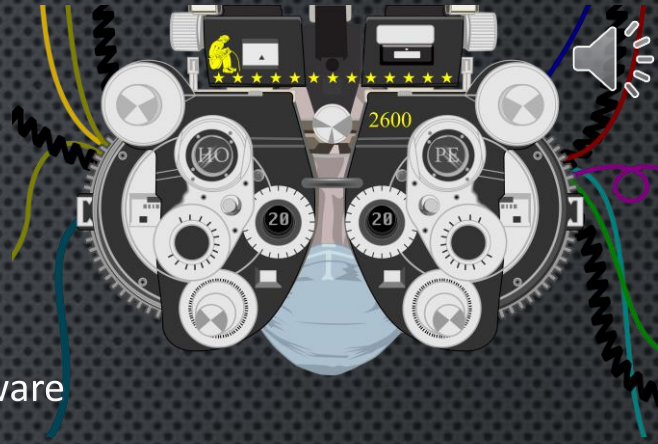


Protocol	Application	Free Tools
UART	Terminal, data logging	PuTTY, screen, minicom, etc.
I^2C	Multiple devices, sensors	python, i2cscan
SPI	Highspeed Display, Memory	avrdude, flashrom, python
SWD	Single Wire Debug, reprogram for CORTEX CPUs	openocd, gdb, urjtag
JTAG	Debug, reprogram	openocd, urjtag
Firmware	Install bootloaders, firmware	avrdude



Hackers vs. proprietary hardware …..

Early AVR programmers



ATMEL/Microchip AVRisp mkII (In-System-Programmer)

- About $200+ in 2000's
- In response, hackers developed open source USB<->SPI programmers to re-flash firmware

2003 - AVRDUDE (AVR Downloader/UploaDEr) – Brian S. Dean

2009

- BusPirate ($40) - Ian Lesnet
- Goodfet ($50/obsolete) - Travis Goodspeed

2010

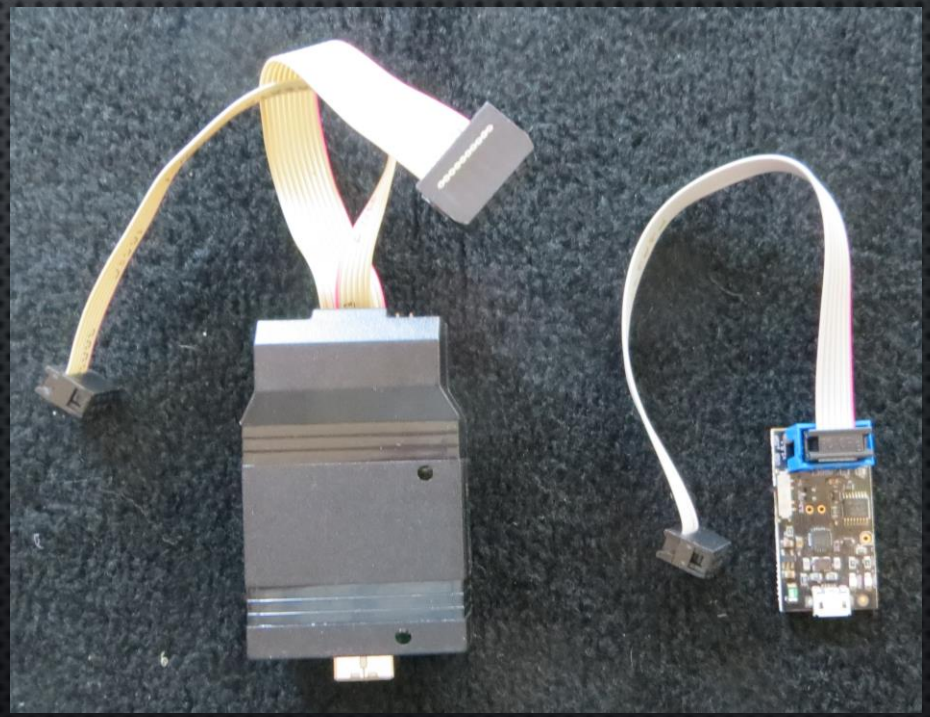
- usbtiny - Dick Streefland

2014

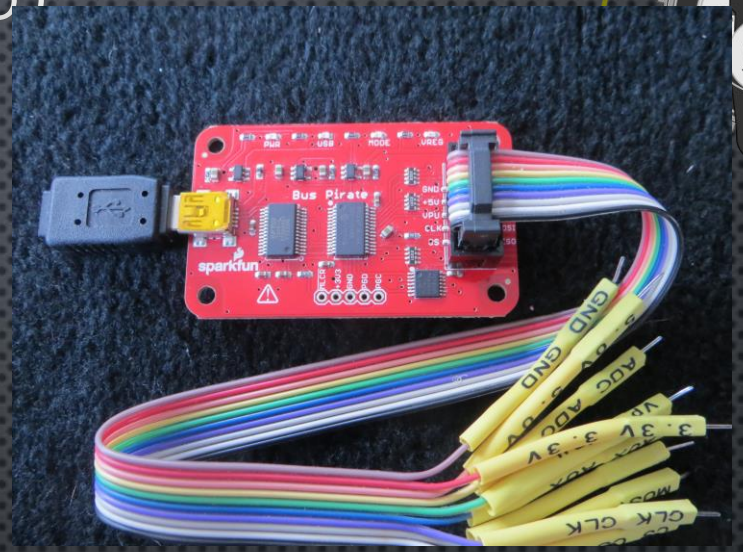
- USBtinyISP kit ($22) –Adafruit

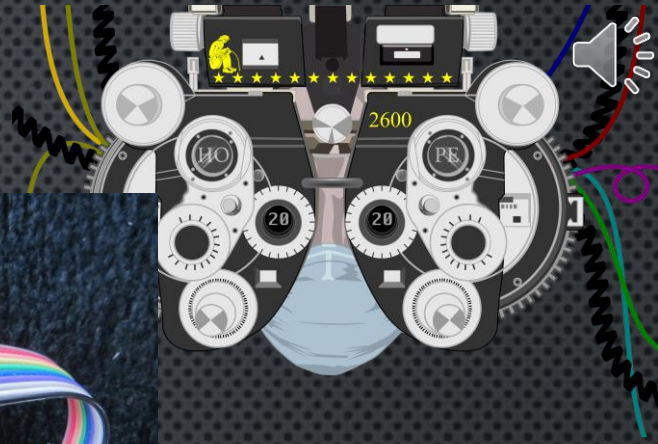
Today

- DIY w/Arduinos
- USBISP AVR Programmer ($2) AliExpress
- USB µISP ($20) @ Tindie by Geppetto Electronics
  - Micro-USB connector
  - Diodes/fuse for protection
  - Works with 3.3V, 5V, self-powered boards



The Noble BusPirate (16-bit PIC CPU)



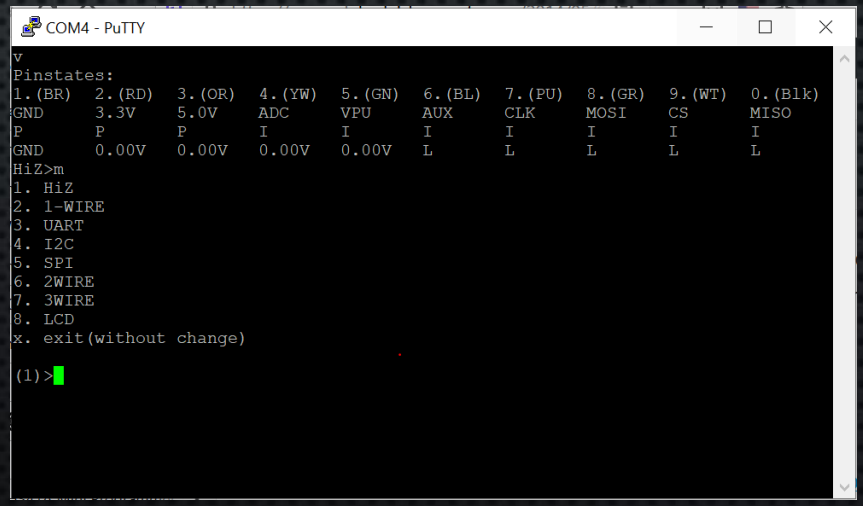


Supports UART, I²C (scan/sniff), SPI (read/write)
- UART-based user interface
- On-board Pull-up resistors
- 3V/5V power pins
- A/D input pin - can be used as a primitive Oscilloscope
- Macros
- BASIC interpreter
- Simple Waveform generator
- Works with most open source tools (avrdude, flashrom, Arduino, openocd)

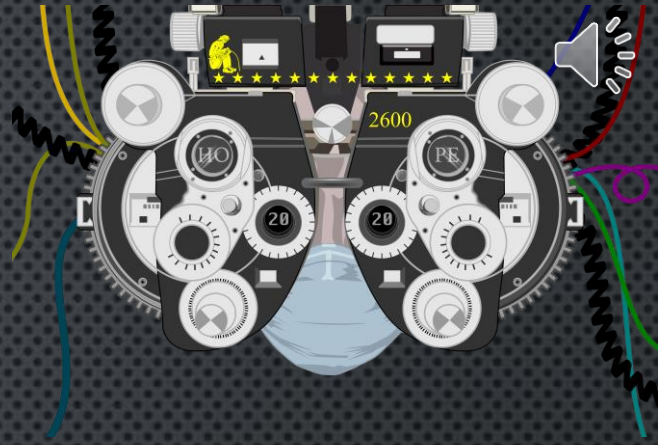
Version 3 - uses standard cable, Memory limited

Version 4 - More memory, no ready-made cables
- “A little flakey”

Problems:
- Old and Slow
  - Flashing takes 30 minutes vs 30 seconds
- No low-level API
- No real JTAG support



Better general purpose boards

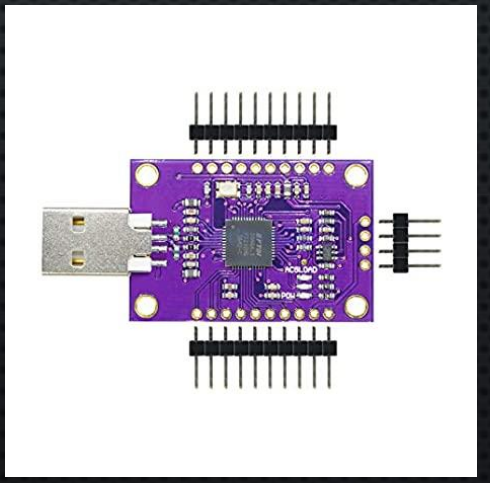


FDTI designed FT232H chip family,

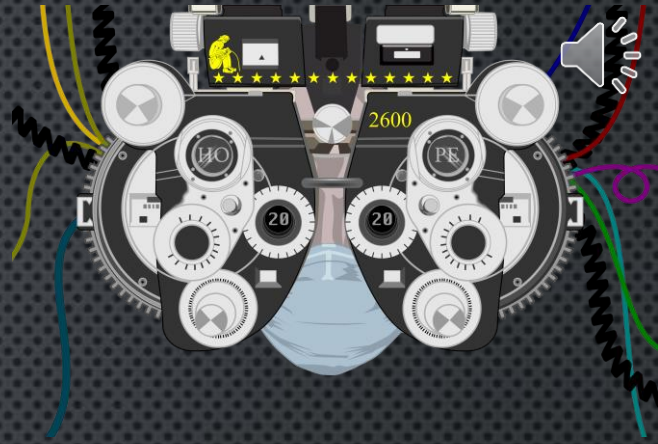
- Multi-Protocol Synchronous Serial Engine (MPSSE)
- Hi-speed UART, I^2C, SPI, JTAG and SWD
- FT2232H boards support 2 connections

Products

- Bus Blaster ($35, $45)
- TUMPA (TIAO USB Multi Protocol Adapter) $40
- Expliot-NANO ($40)
- Shikra ($45)
- Flyswatter2 ($89)
- Adafruit FT232H ($15)
- FT232H CJMCU ($7, $15)



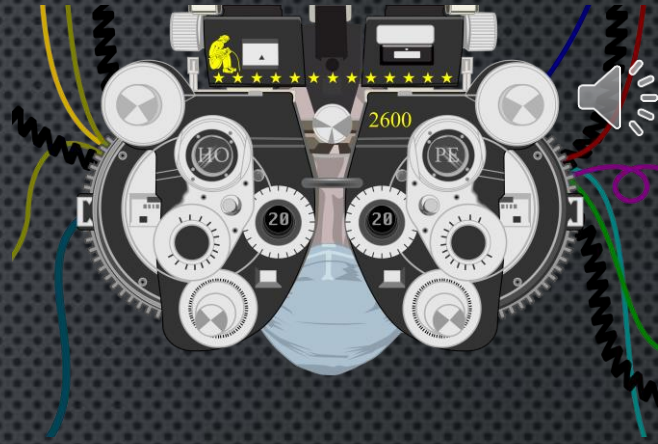
FT232H Boards

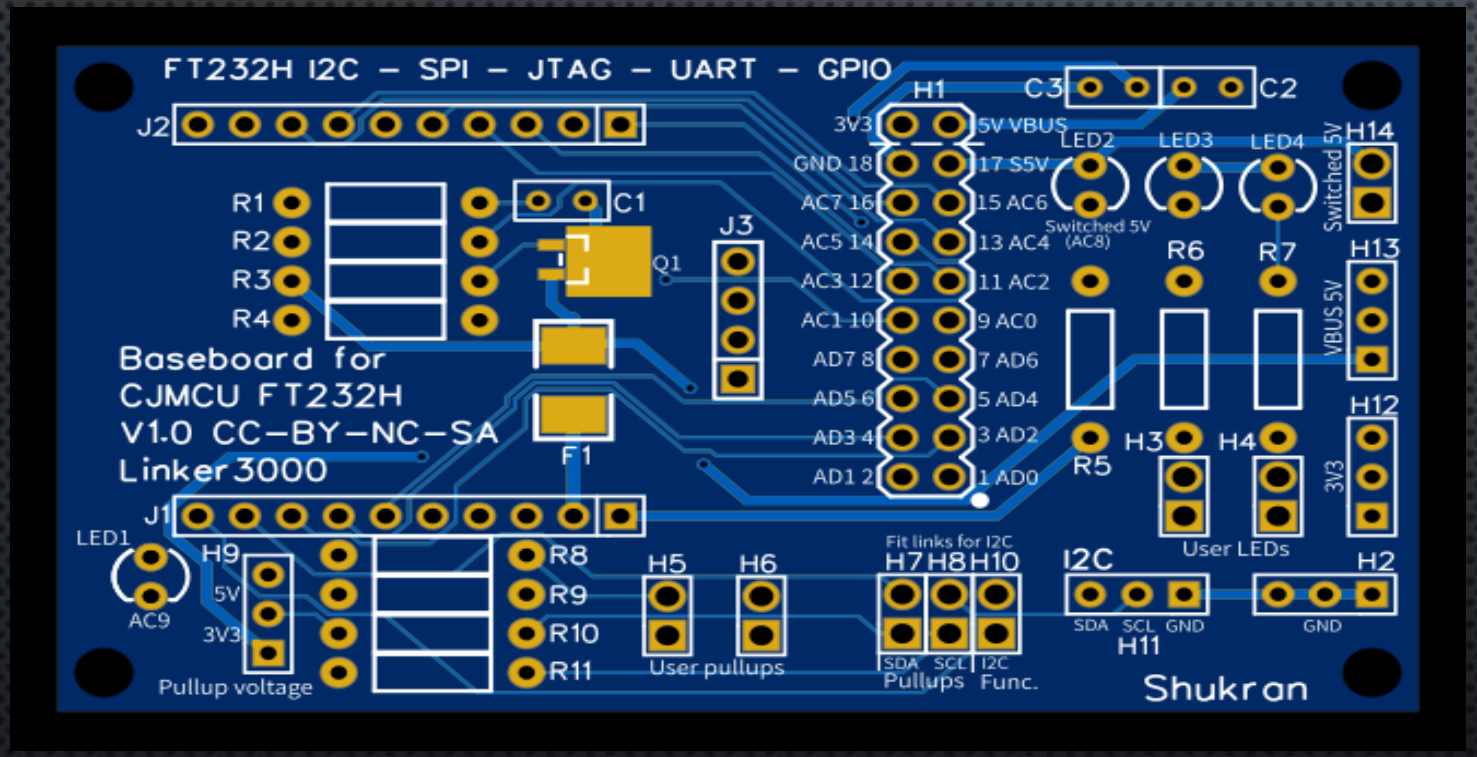


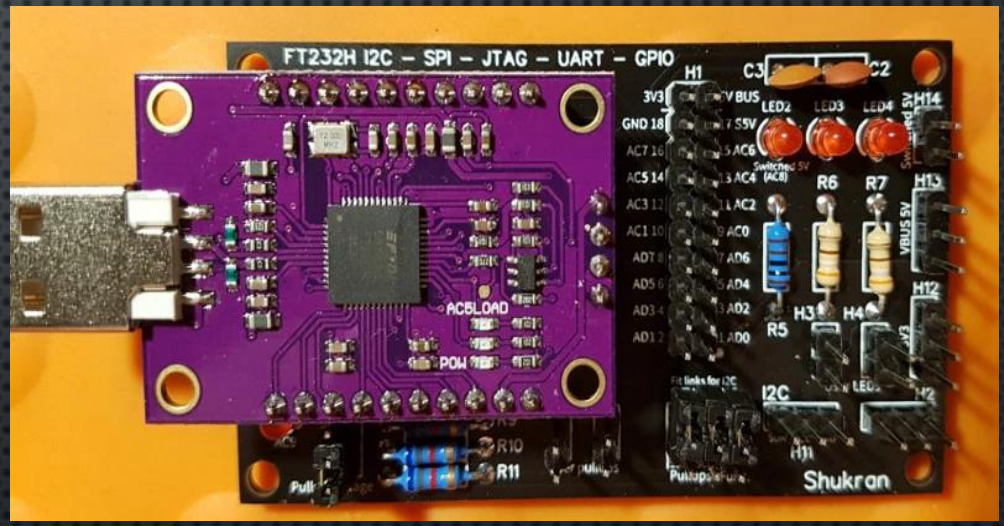
Issues

- No dedicated user interface
  - Serial putty, screen
  - I^2C - Python Libraries
  - SPI - flashprog, avrdude
  - JTAG - openocd, urjag
  - SWD - openocd
- Python libraries for the API
  - pyftdi & libusb
- Individual tutorials are spread all over
- No Pull-ups
- Wiring I^2C, SWD not well documented
- Mis-wiring can affect laptop USB power
  - Shikra board has a polyfuse for protection

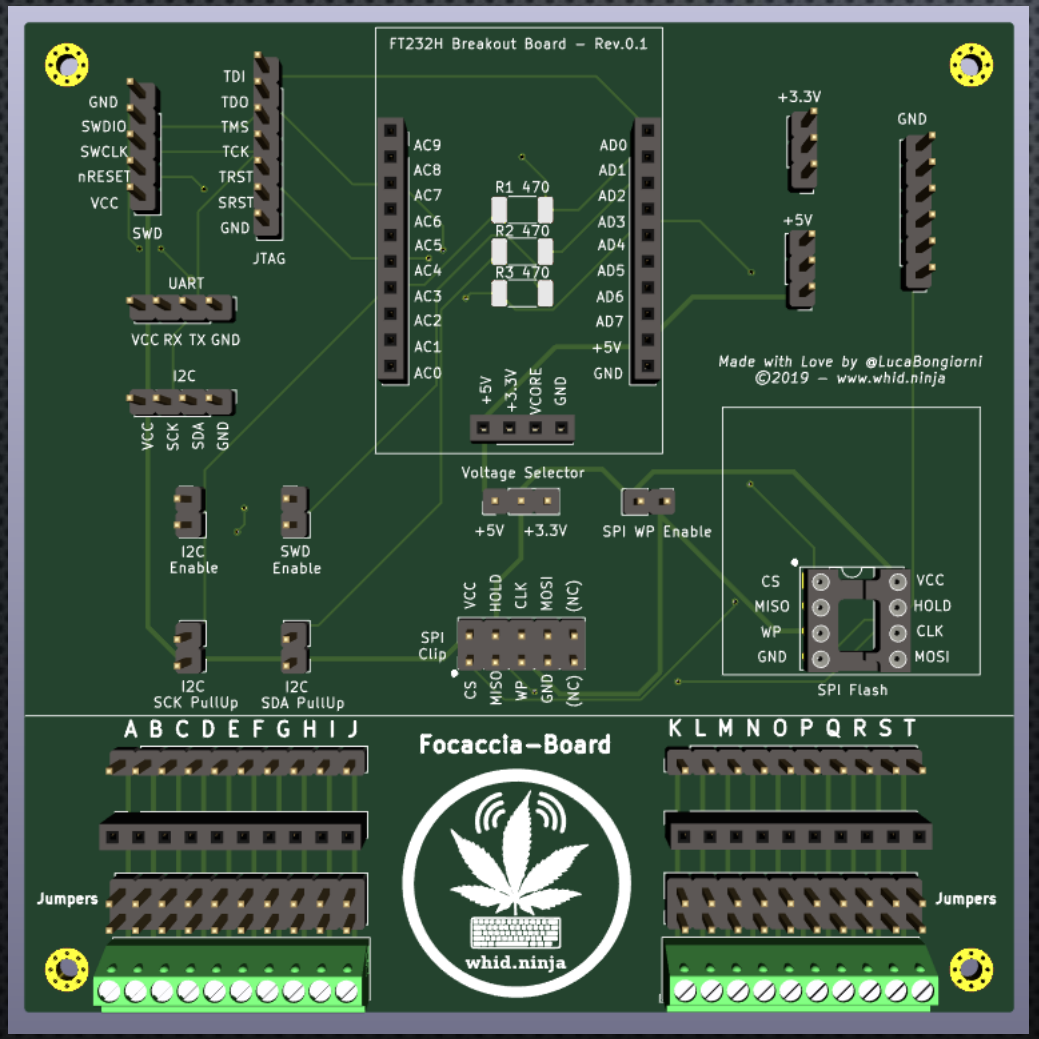
Shukran - FT232H Breakout board for CJMCU

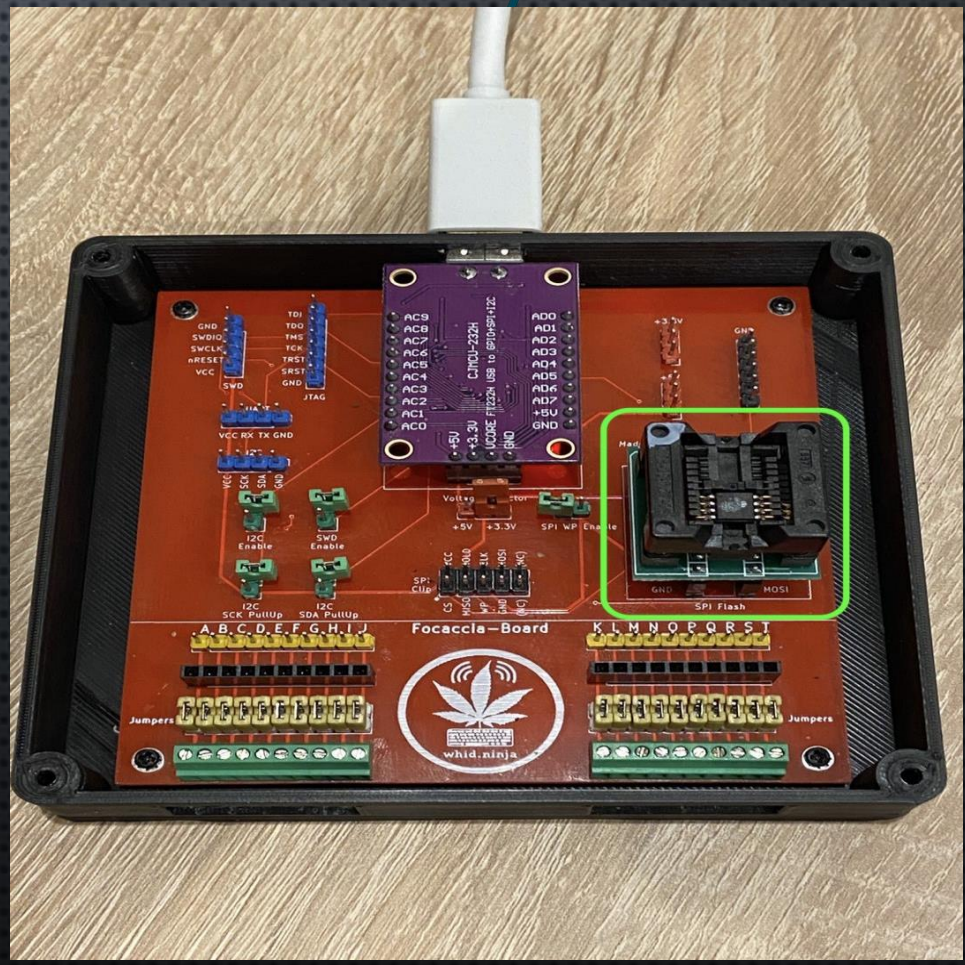




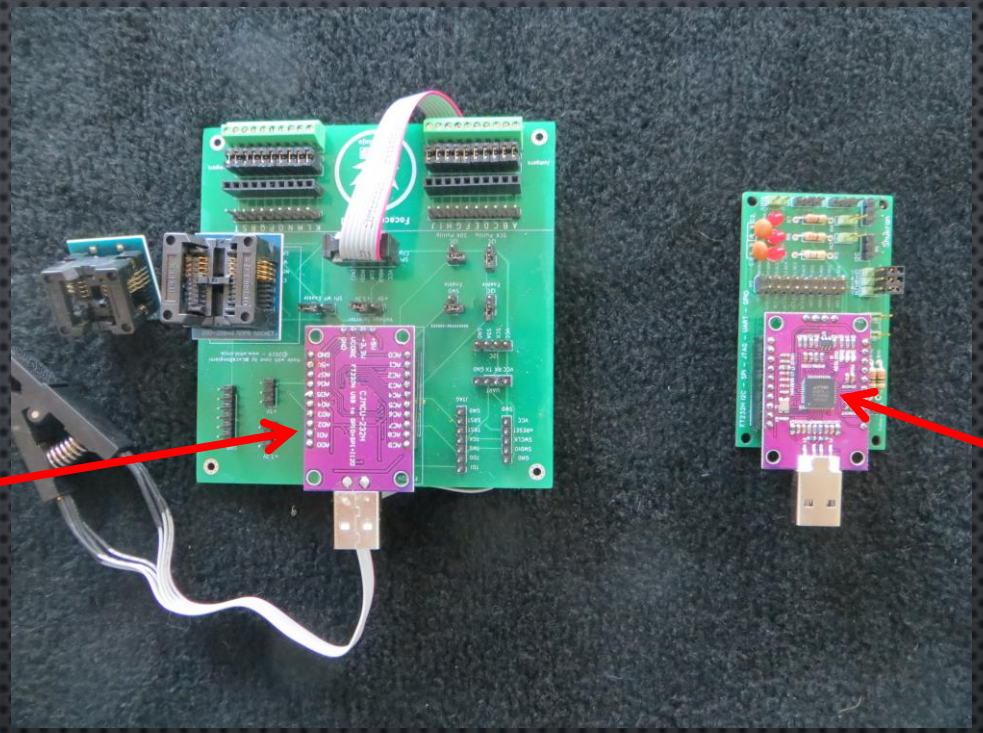


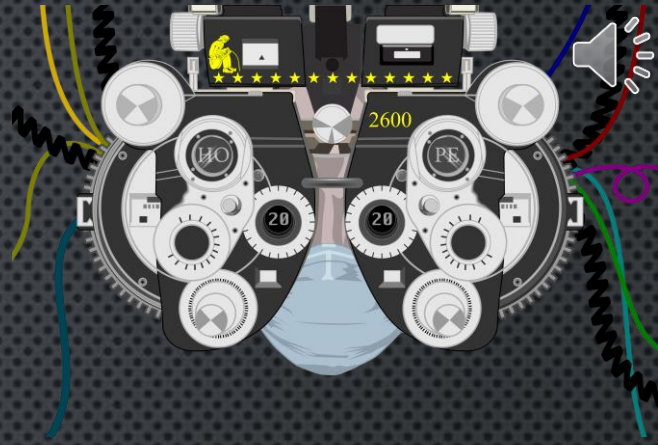
Focaccia - FT232H Breakout board for CJMCU





Focaccia vs Shukran



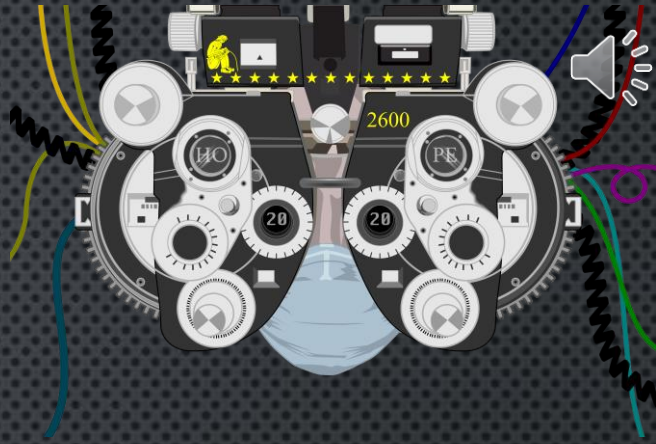


Focaccia

- Commercially available for $10
- Headers for SWD, JTAG, UART, I2C, 3.3V, 5V, GND, SPI test clip, SPI SMT socket.
- Jumpers for I2C, I2C pullups, SWD, SPI WP
- Extra connectors for easy hook-ups
- More readable labels
- FT232H is under CJMCU board

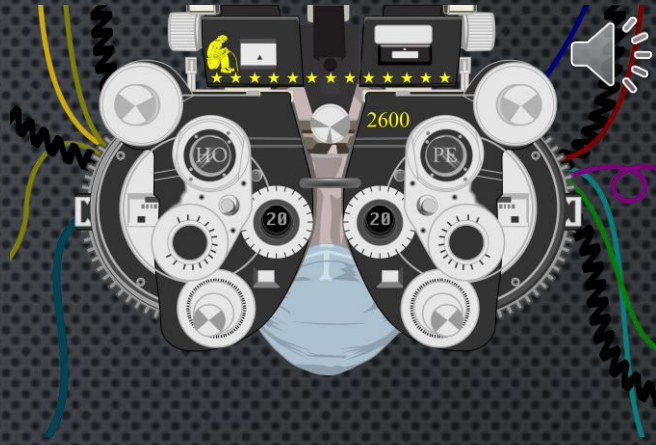
Shukran

- Smaller
- DIY (cost $7.75 for 5 boards or $1.55 each w/JLCPCB and easyeda.com)
- Headers for 3.3V, 5V, GND, I2C
- Jumpers for pull-ups, I2C, 3.3V vs 5V
- Polyfuse to protect laptop from short on I/O
- All FT232H pins available
- FT232H is above CJMCU board

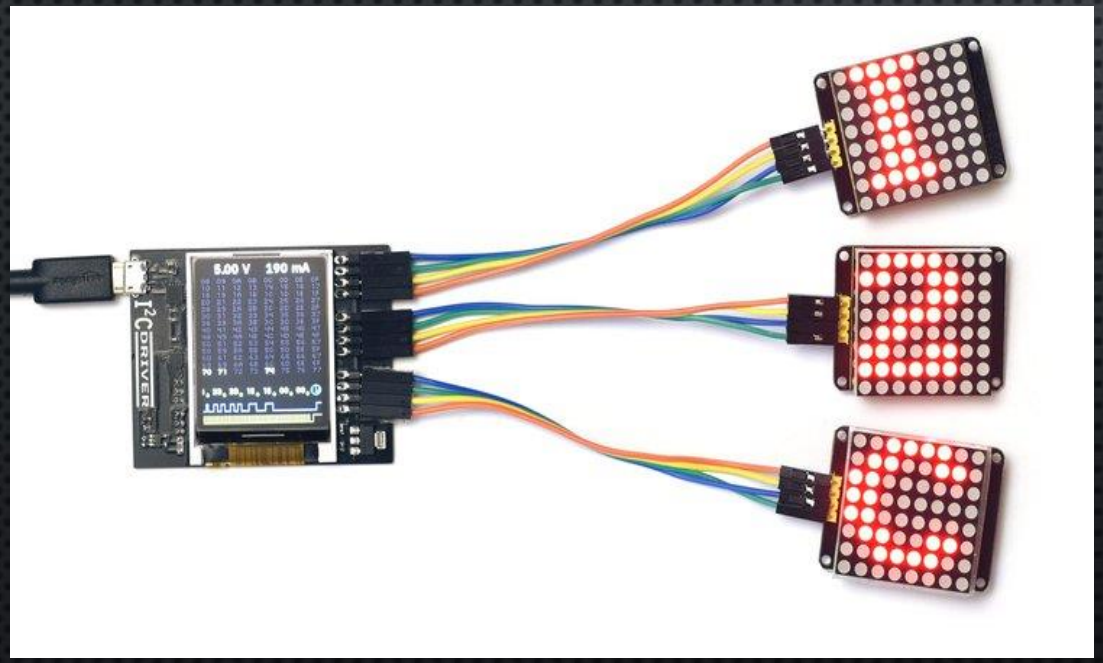


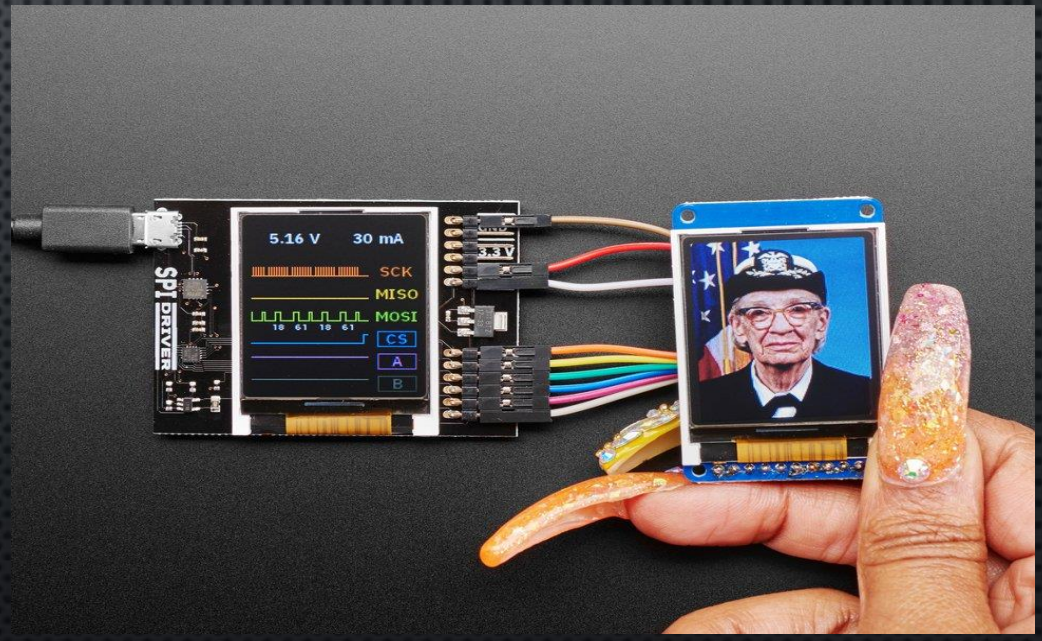
SPECIALIZED BOARDS

EXCAMERA (JAMES BOWMAN)



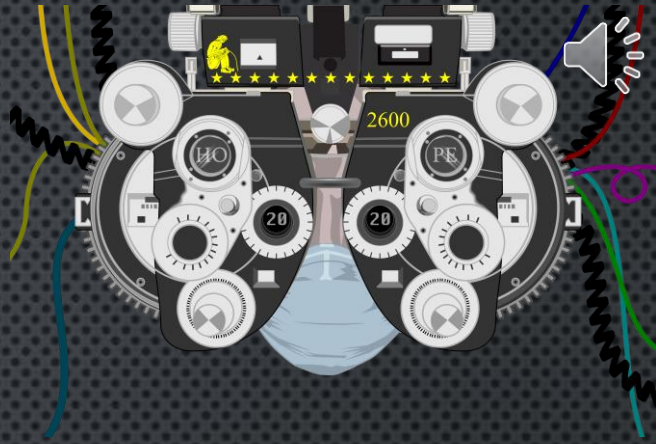
I²CDriver ($30)
& SPIDriver ($30)
Python Interface

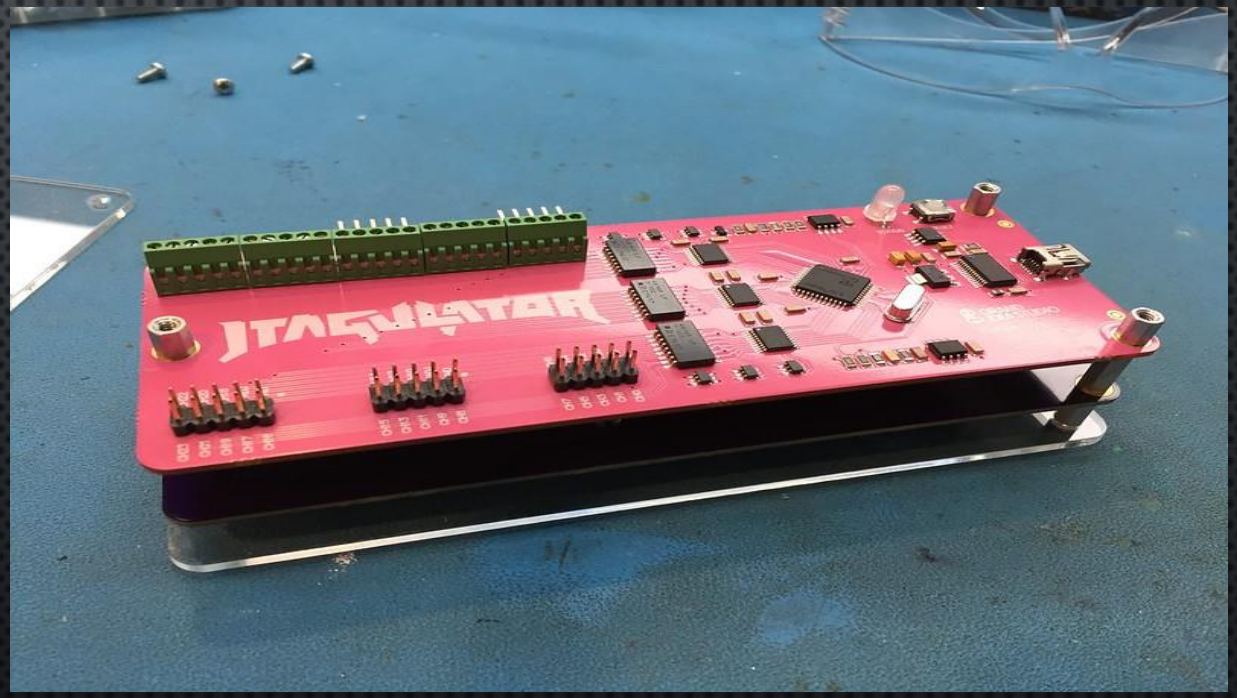




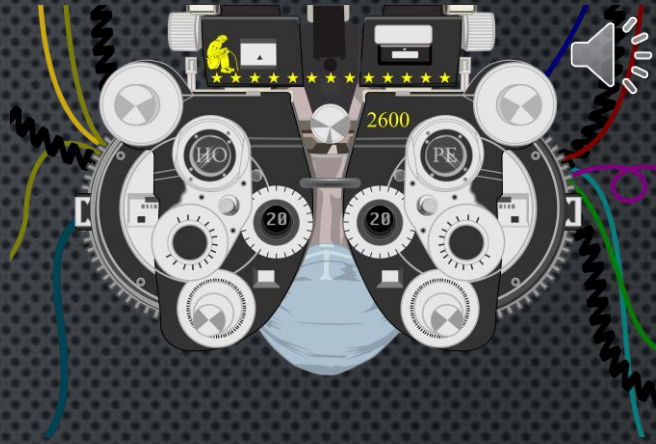
JTAGULATOR ($175)

- 24 1/0 channels
- 1.2V-3.3V
- Automatically identifies serial (UART), JTAG pins
- Time-saver for professionals
- Arduinos can do it, just slower



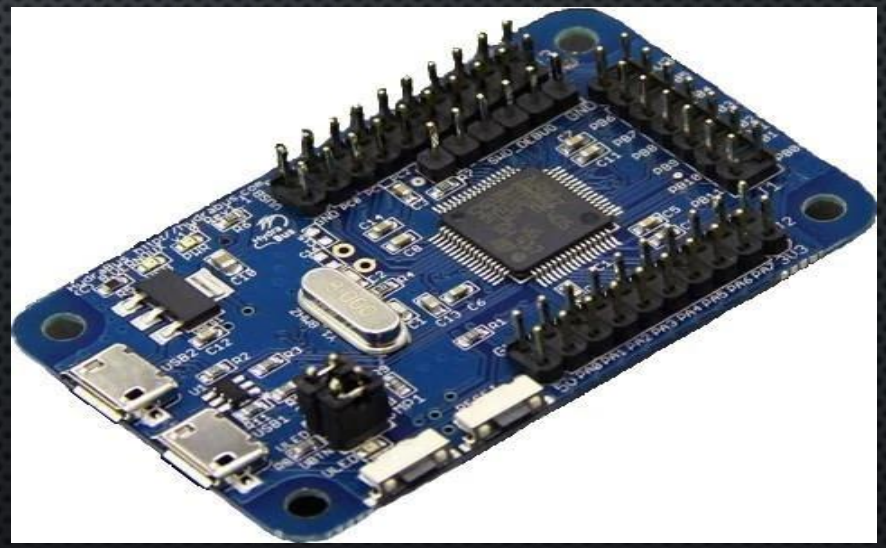


HYDRABUS (2013)

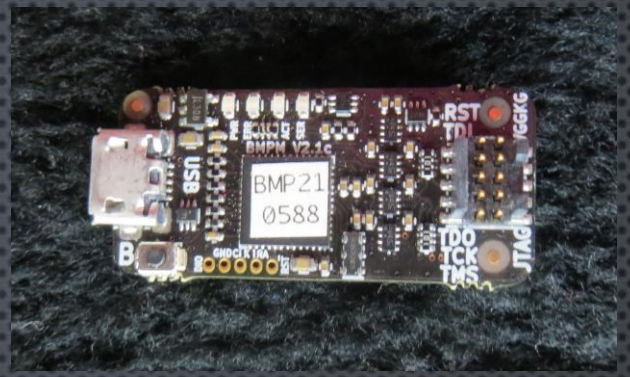


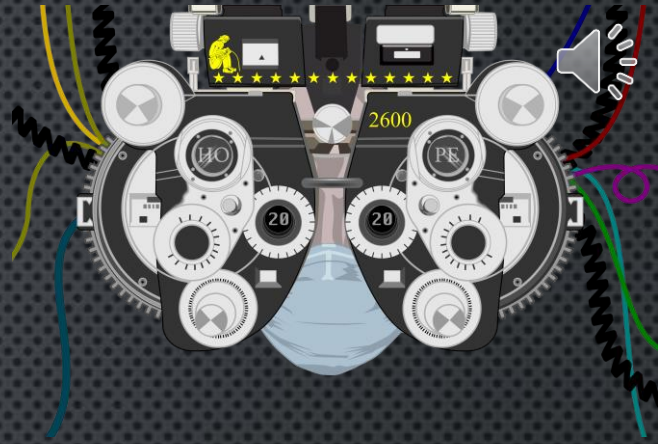
$85
$190 w/NFC Shield
Functions

- JTAG scanner/debugger mode like JTAGulator
- Logic Analyzer mode up to 2MHz 16chan with SUMP support
- CAN1 or 2 (up to 2 Mbit/s)
- SPI1 or 2 (master & slave up to 42MHz)
- I2C (master up to 1MHz)
- UART1 & 2 (up to 10.5Mbps)
- ADC (up to 3.3V, can read internal Temperature, VrefInt, VBAT)
- PWM (1Hz to 42MHz, Duty Cycle 0 to 100%)
- GPIO up to 44 I/O configurable (PA0-15, PB0-11, PC0-15)
- RNG (Random number generator using STM32 hardware RNG)



Black Magic Probe ($60)



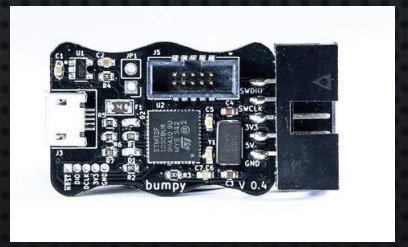


SWD/JTAG Debugger
Version 2.1 (2017)

- Like ST-Link but open source
- Some ARM Cortex-M and Cortex-A CPU's
- Uses GCC toolchain
- Interface: GDB
  - No need to use openocd
- Works with Windows, Linux, MacOS
- Does not support 64-bit devices

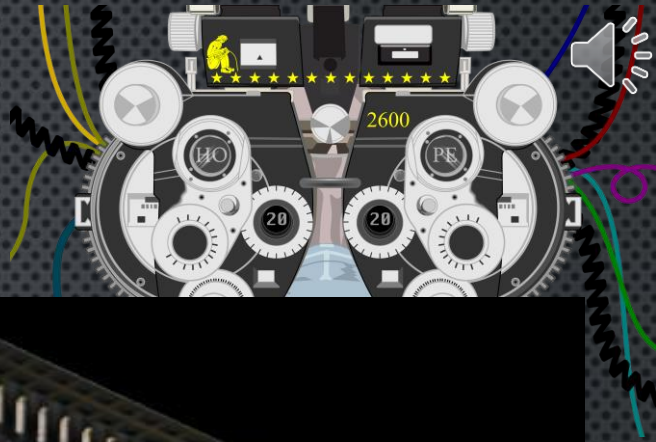
Jeff Probe - $16
(flirc.tv, amazon)

ST Link V2 Clones $2



Bumpy - $12
Tindie -
Electronut Labs

GreatFET ($100)



General Purpose

- FPGA-based
- 2 USB ports (USB hacking)
- 100 IO pins
- SPI, I^2C, UART, and JTAG
- Logic Analyzer
- A/D
- 2 MB Flash
- Neighbor boards planned (RF, IR, SDR, Level Shifter, etc.)
- Use with GNU Radio as DSP

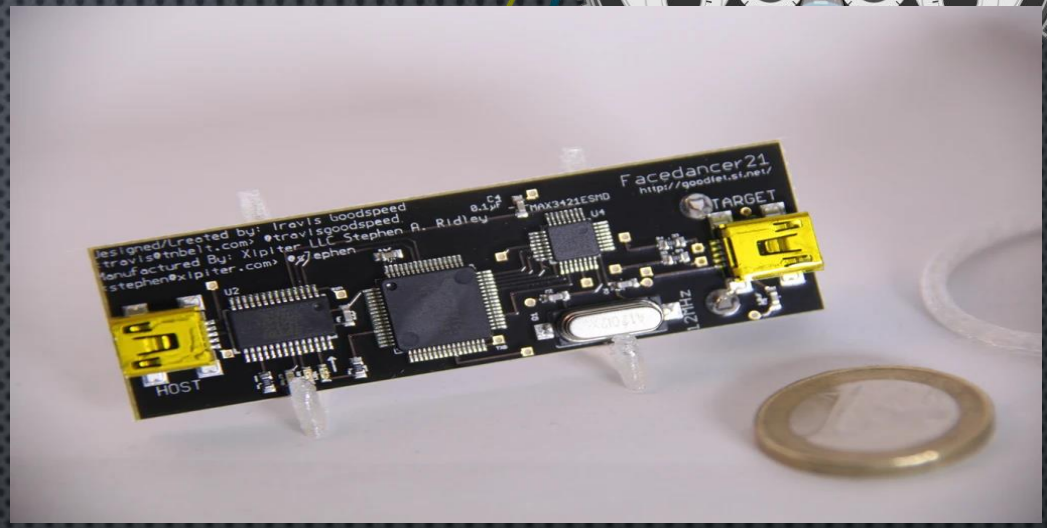
Watch “Software Defined Everything” talk

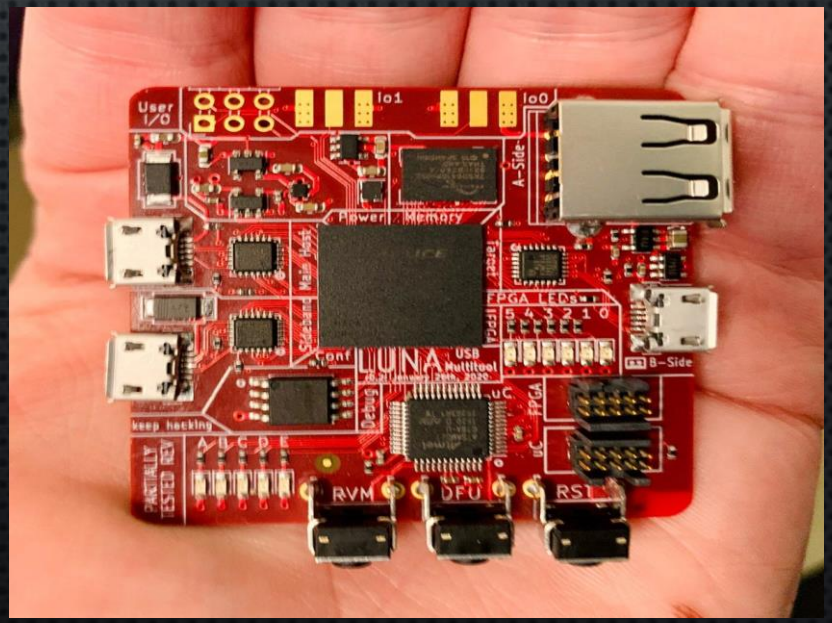
The Future ...

https://www.flickr.com/photos/andertoons-cartoons/2969559163

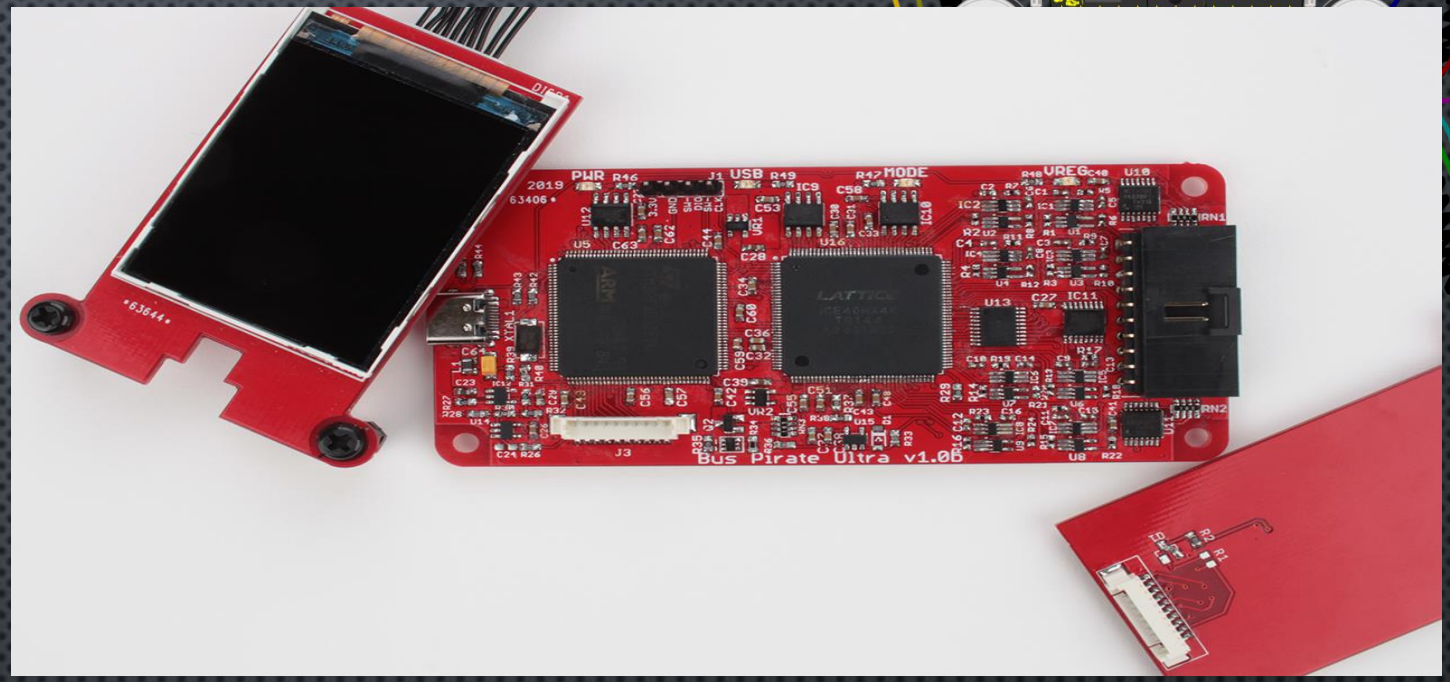
USB Protocol Hacking

- Facedancer ($85) - Deprecated
- GreatFET ($100)
- Luna ($?) - Kate Temkin



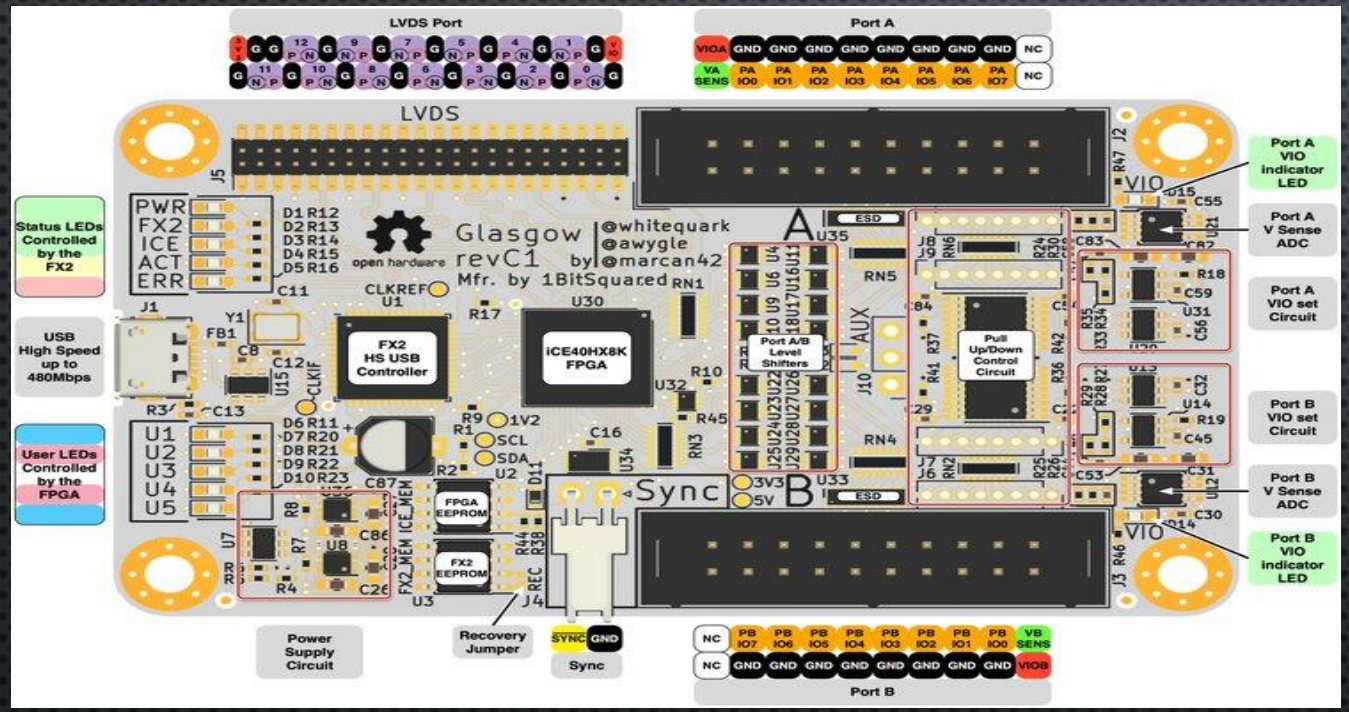


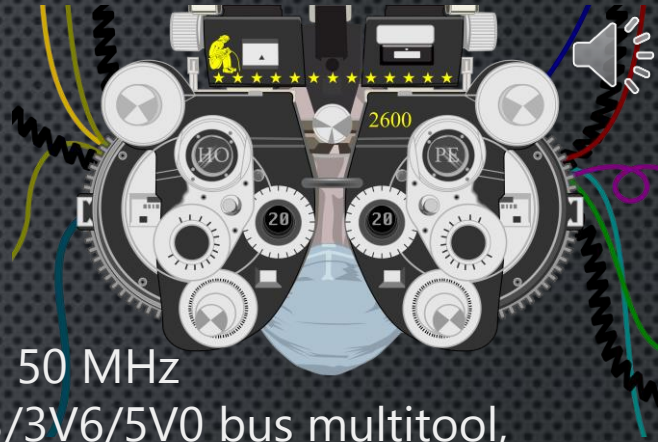
Bus Pirate Ultra (beta)



- ICE40 FPGA + STM32F103 MCU
- Display Connector
- 0.8-5.0V programmable power supply
- 8 I/O pins w/voltage measurement
- Pull-up resistors on all pins
- Built-in Logic Analyzer (2nd USB for data collection)

Glasgow Board (Crowd Supply pre-announcement) - @whitequark





"Glasgow is a 50 MHz 1V8/2V5/3V3/3V6/5V0 bus multitool, think Bus Pirate + Bus Blaster + Logic Sniffer all in one reconfigurable package you have 16 pins. put any of {JTAG, SWD, SPI, I2C, UART,...} on any of them, or even use your own protocol core on the FPGA!"

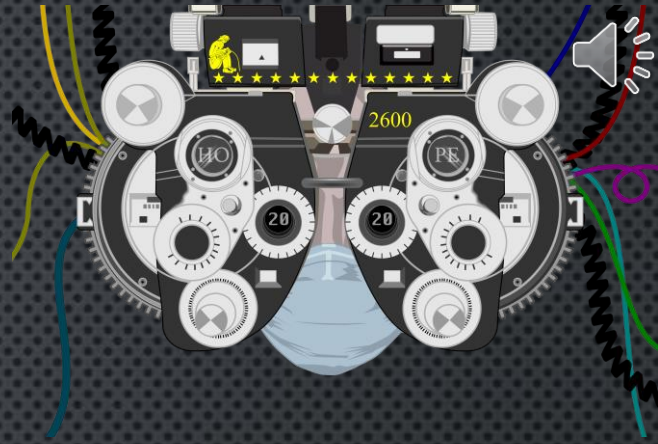
Cost: perhaps ~ $100 early bird

Edinbugh: lower-cost variation by @FauthNiklas

From @LucaBongiorni [Focaccia Board, etc.]

- The Burtleina Board ??
  - **NodeMCU ESP8266**
  - **NRF24L01?**

Resources



https://www.youtube.com/channel/UC0W5_eccqXJ7BZooCRzcy3Q
VOIDING WARRANTIES with @jilles_com

- Start with #0 - Finding uart data pins on WRT54G

https://github.com/unprovable/PentestHardware/blob/master/DRAFT.pdf

- Mark Carney's book on PenTesting hardware

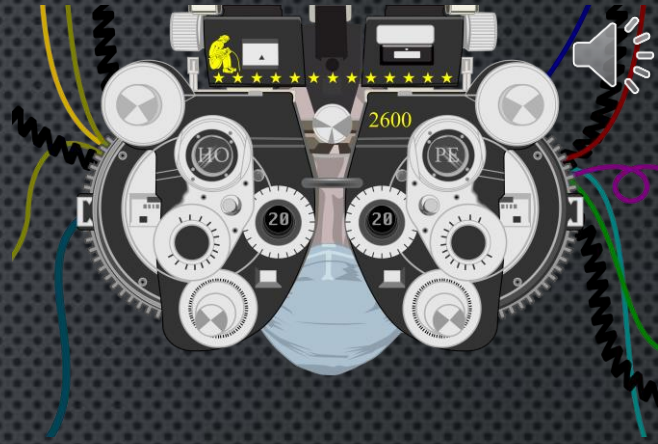
https://github.com/fkie-cad/awesome-embedded-and-iot-security

- Curated list of hacking resources

UNRE - Unnamed Reverse Engineering Podcast

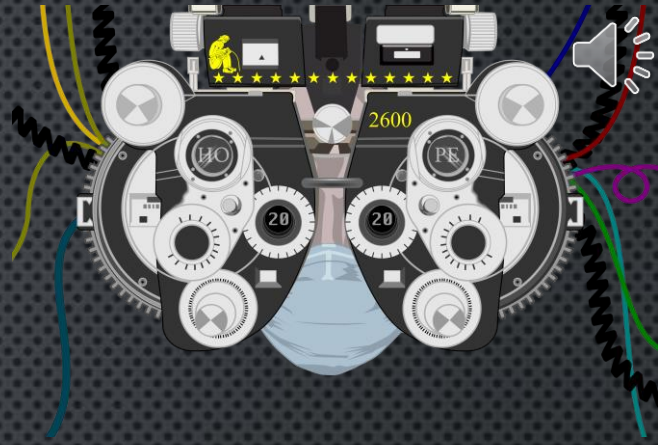
Me: @grymoire, http://www.grymoire.com, http://grymoire.wordpress.com/

What's your next step?



- Get your tools
- Find a device to hack
- Find ground on board
- Measure voltages with your multimeter
- Discover interfaces
- Learn to use the software
- Tell your friends what you discover

Thanks



General Help

- @cybergibbons - Andrew Tierney
- @LargeCardinal - Mark C
- @jilles_com
- @dcuthbert - Daniel Cuthbert

Boards:

- @travisgoodspeed - Goodfet
- @dangerousproto : Ian Lesnet - BusPirate
- @joegrand - JTagulator
- @1bitsquared - Black Magic Probe
- @michaelossmann - GreatFet
- @gameduino : James Bowman - I2Cdriver, SPIDriver
- @LucaBongiorni - Focaccia Board
- @linker3000 : Nigel Kendrick - Shukran board
- @whitequark - Glasgow Board

Questions?