Bring VGA to the Game Boy

Retro hardware + DSP + FPGA = ❤️

Bio

- Calle Svensson, 28
- MSc Computer Science, KTH
- Head of Security, KRY/LIVI
- HackingForSoju, CTF
- Twitter: @zetatwo
- Email: calle.svensson@zeta-two.com
- Web: https://zeta-two.com

Last night

- I did a talk about Game Boy hacking
- ~~I did not use my laptop~~
- My slides were quite retro

#GAME BOY ARCH

>Hybrid
-Zlog Z80
-Intel 8080

>8-bit

>Sharp LR35902
-4MHz
-RAM: 8kb
-ROM: 32kb-8mb

>LCD: 160x144px
-60Hz

Rewind: June

- Talk accepted
- Start making slides

What if...

Slides

- Game Boy ROM
- GBDK C library
- Custom Python scripts
- Photoshop

What if...

Present from an actual Game Boy

- Flash ROM on real cartridge
- Connect Game Boy to projector

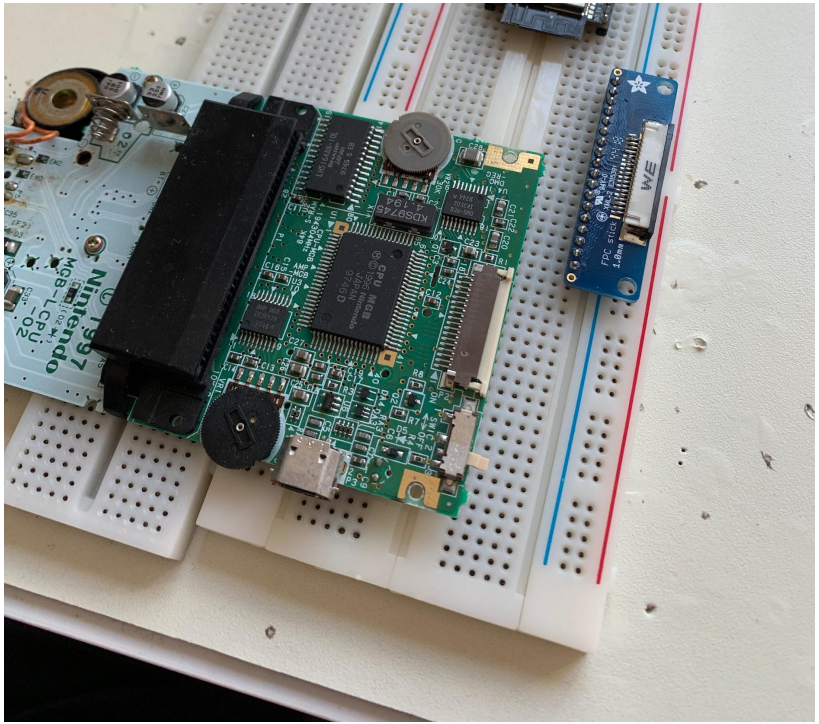
Problem

- Talk about Game Boy
- Slides as a ROM - Nice
- Show on an emulator - Lame (acceptable backup)
- Show on a real Game Boy - Cool
- Game Boy has no VGA/HDMI - Sad

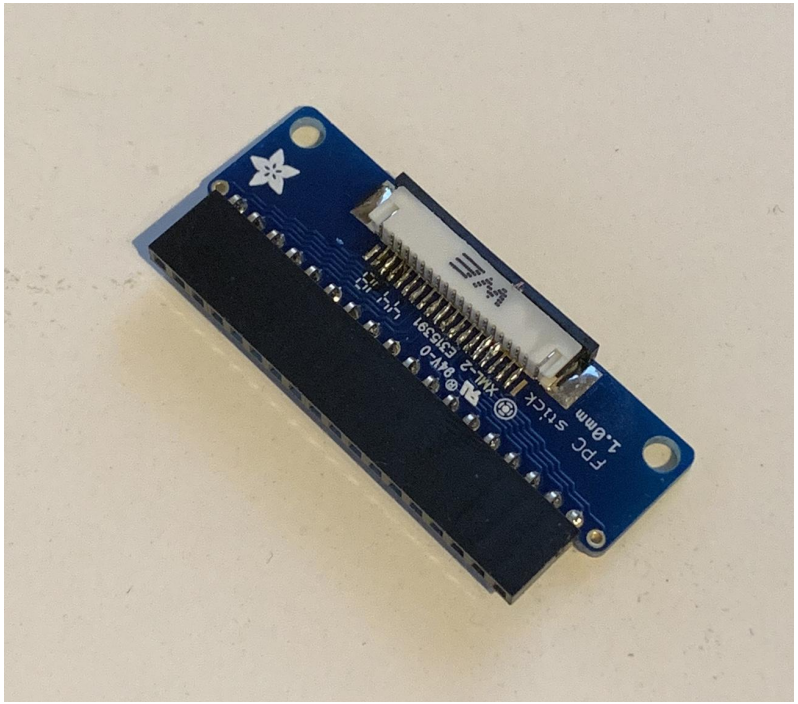
Battle Plan

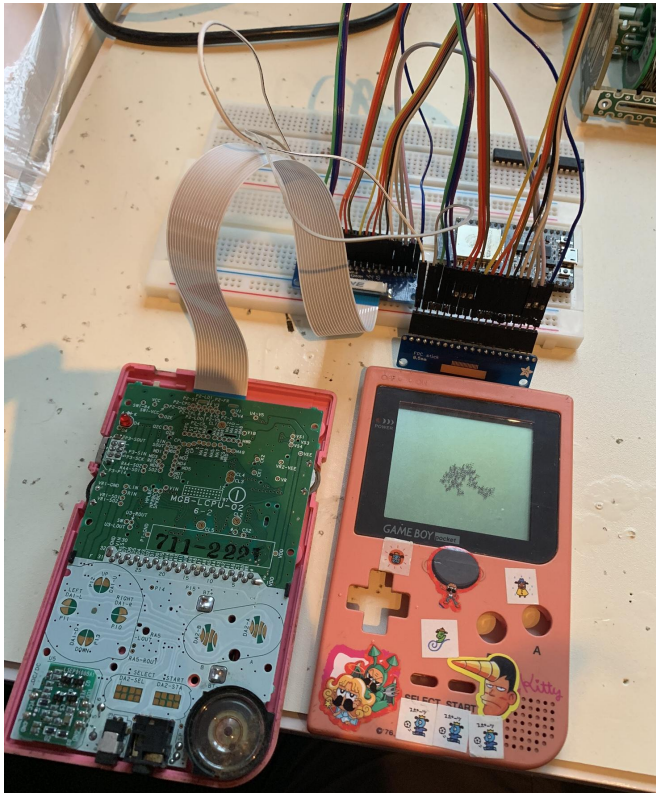
- Disassemble Game Boy
- Splice LCD connector

Disassemble Game Boy



Splice connection with custom splitter





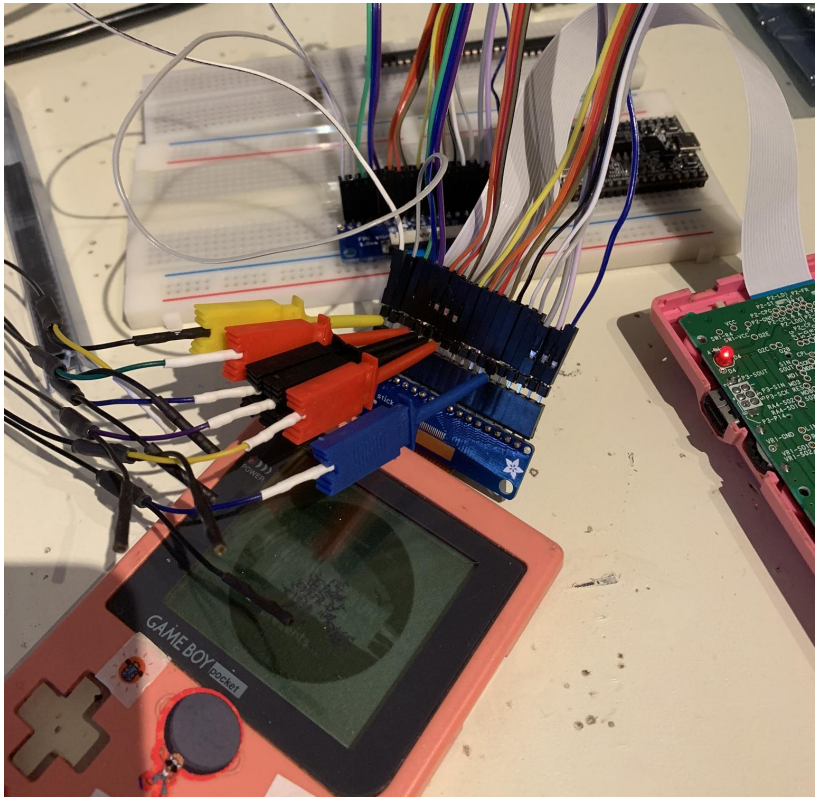
Battle Plan

- Disassemble Game Boy
- Splice LCD connector
- Record LCD traffic

Capture and analyze LCD data

- Hook up logic analyzer
- Start Game Boy
- DMG != MGB
- 18/21 pins
- 5v logic
- Only 6 relevant

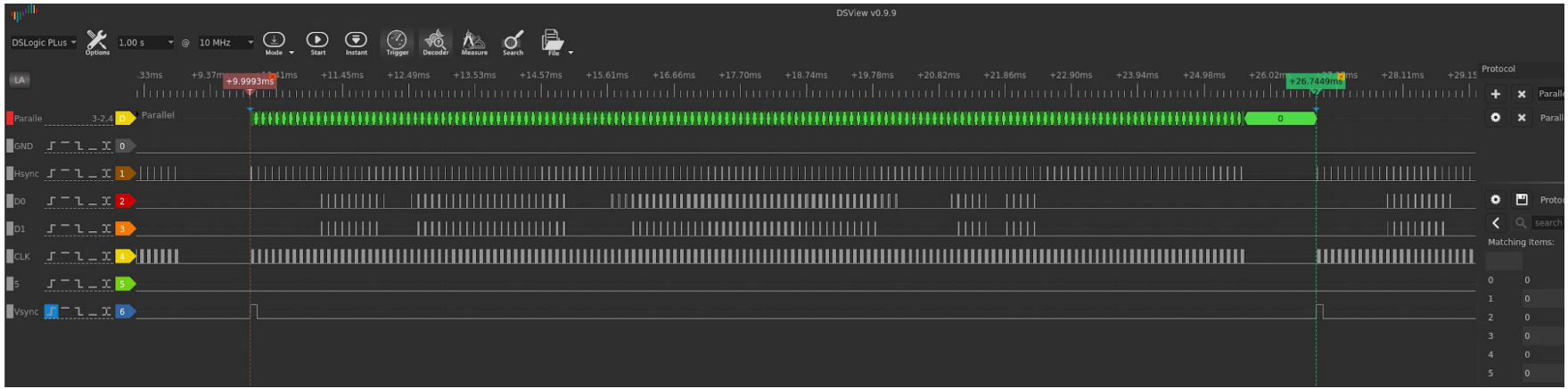
Capture LCD traffic



Battle Plan

- Disassemble Game Boy
- Splice LCD connector
- Record LCD traffic
- Analyze and render offline

Analyze traffic



Render image

- Pins
  - GND
  - H-sync
  - V-sync
  - Data 0
  - Data 1
  - Pixel clock (4MHz)
- V-sync + 144 * (H-sync + 160 * (D0/D1/PxCLK))
  - 60 FPS
  - 160x144x2 = ~2.7 Mbit/s!

Render LCD data

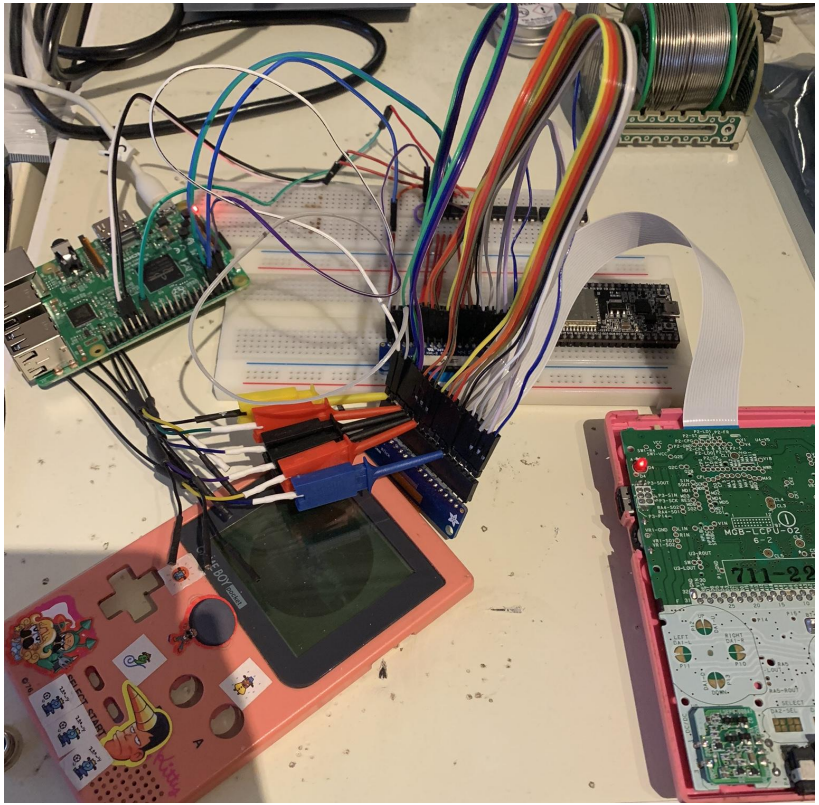
Battle Plan

- Disassemble Game Boy
- Splice LCD connector
- Record LCD traffic
- Analyze and render offline
- Build adapter to render in real time

Build real time renderer

- Raspberry PI
  - Python x2, lol nope
  - Golang x2, nope
  - C, almost

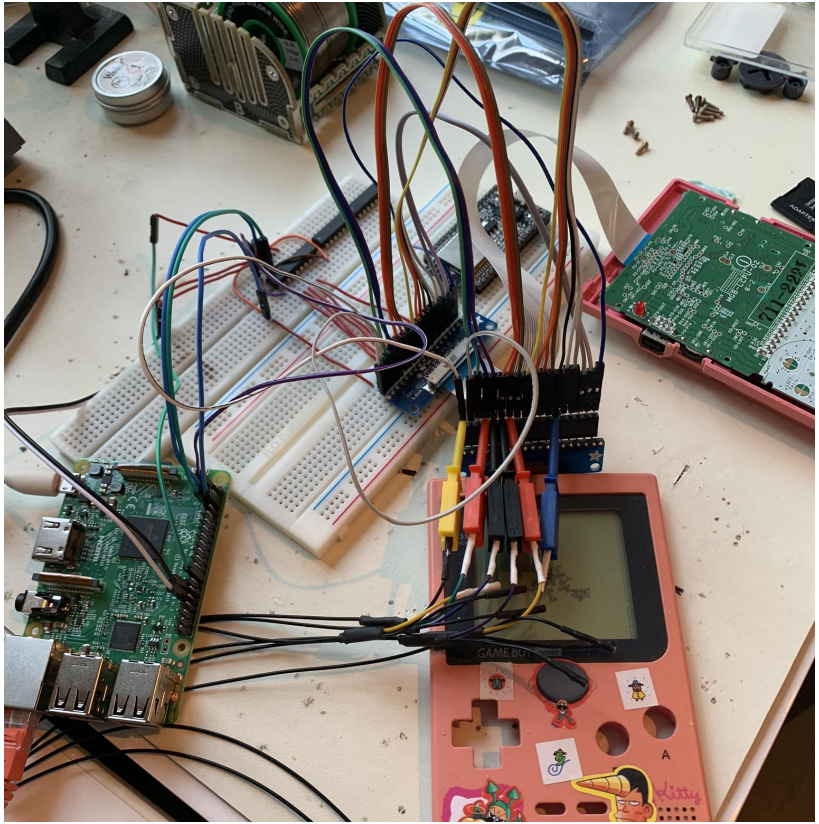
First attempts - Raspberry Pi



Build real time renderer

- Raspberry PI
  - Python x2, lol nope
  - Golang x2, nope
  - C, almost
- ESP32
  - C, fast enough, but…
  - Could halve framerate, lame

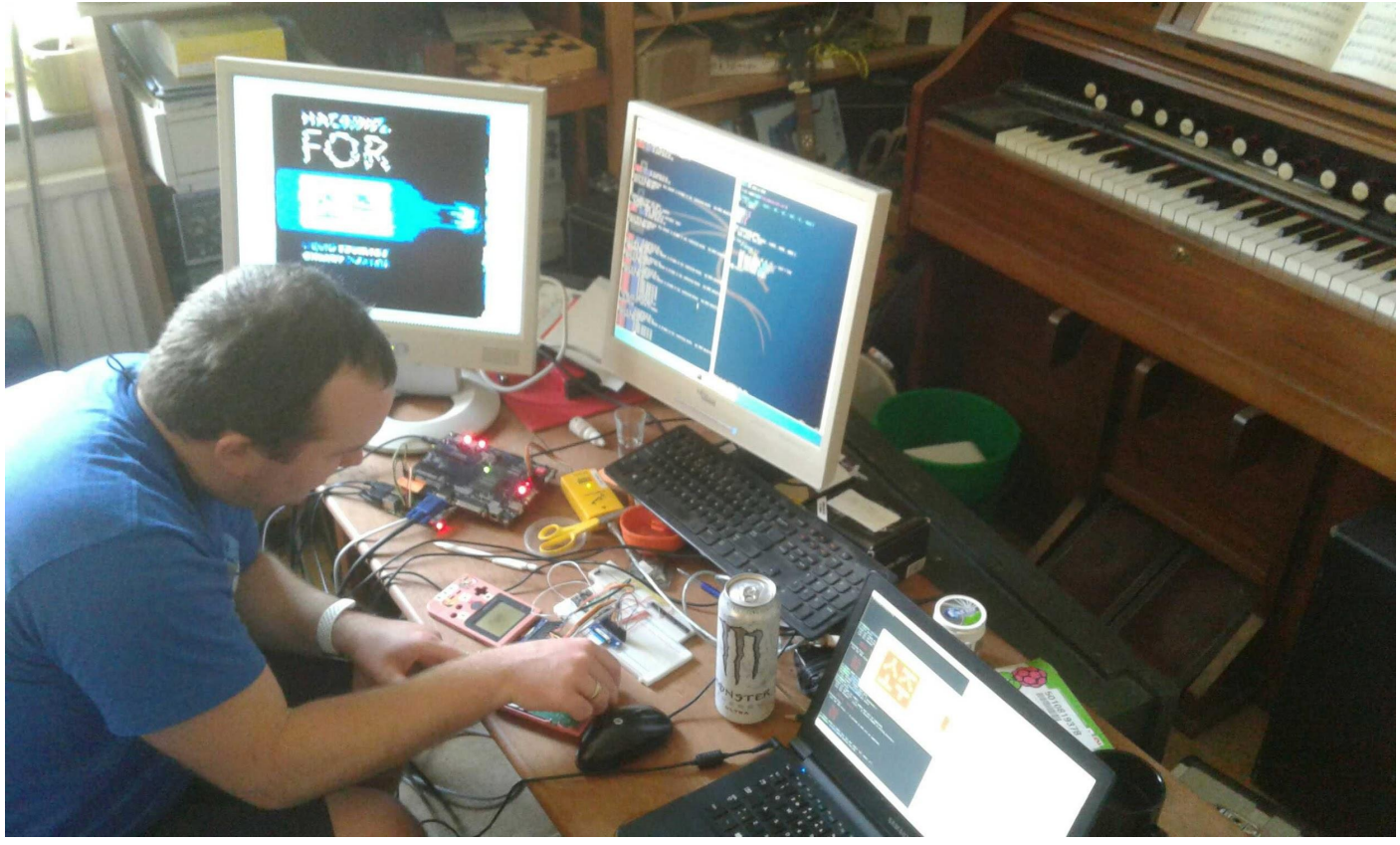
Next attempts - ESP32



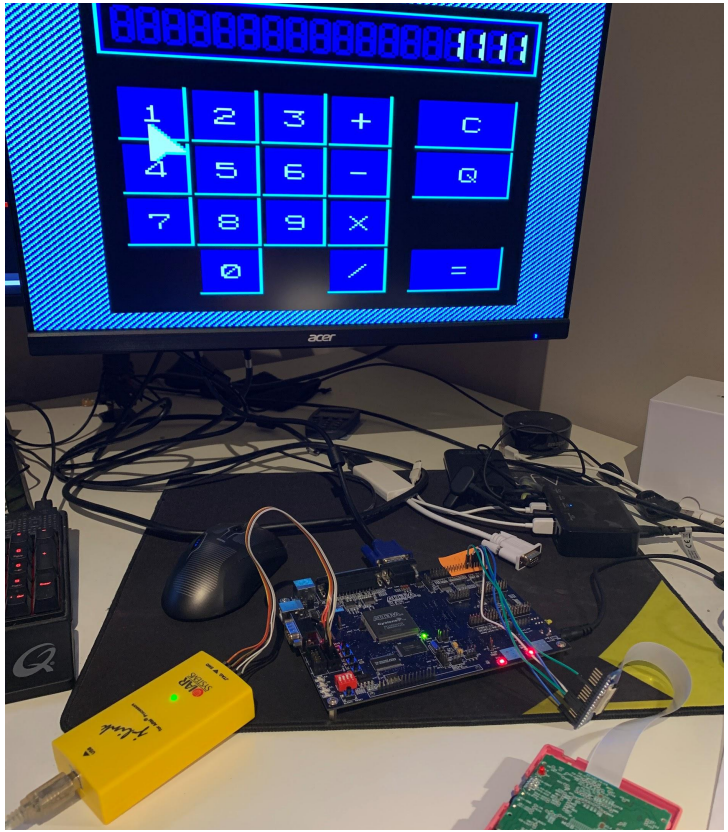
Build real time renderer

- Raspberry PI
  - Python x2, lol nope
  - Golang x2, nope
  - C, almost
- ESP32
  - C, fast enough, but…
  - Could halve framerate, lame
- FPGA
  - The “proper” way

Final attempt - FPGA



Success!



Build real time renderer

- Raspberry PI
  - Python x2, lol nope
  - Golang x2, nope
  - C, almost
- ESP32
  - C, fast enough, but…
  - Could halve framerate, lame
- FPGA
  - The “proper” way

Questions?

Thanks for listening

- Pwny Racing
  - Monthly
  - YouTube
  - https://pwny.racing
- HackingForSoju
  - b0bb
  - LarsH
  - grocid

PWNY RACING