We STILL have no idea how to hack a Furby Connect

loss

swarley

What’s all this then?

- Repeat of a talk presented at BSidesCBR!
  - Sorry for those unfortunate enough to have seen it already
- Not sick oday
- Not amazing feats of hackery
- Talk is meant to be about the process and lessons learned
- Some of you are totally going to see immediately where we fucked up
  - There are LOTS of examples
  - Please advice

Who are these jerks?

- loss
  - part time hacker; full time lover
  - reformed grumpy sysadmin
  - works for Asterisk
- swarley
  - reformed academic hacker
  - WAHCKon guy
  - works for Asterisk

Why would you do such a thing?

- ~~Corporate responsibility and think of the children and…~~
- ~~IoT Security~~
- We saw it in an xmas catalogue and thought it would be hilarious to hack
  - We REALLY wanted our Furb to say/do things that weren’t in the spirit of the original specifications
- We hadn’t done much hardware hacking previously, and wanted to give it a shot
  - Hopefully this serves as a primer/warning for anyone who wants to?

WTF is a Furby? - Gen 1 Furby

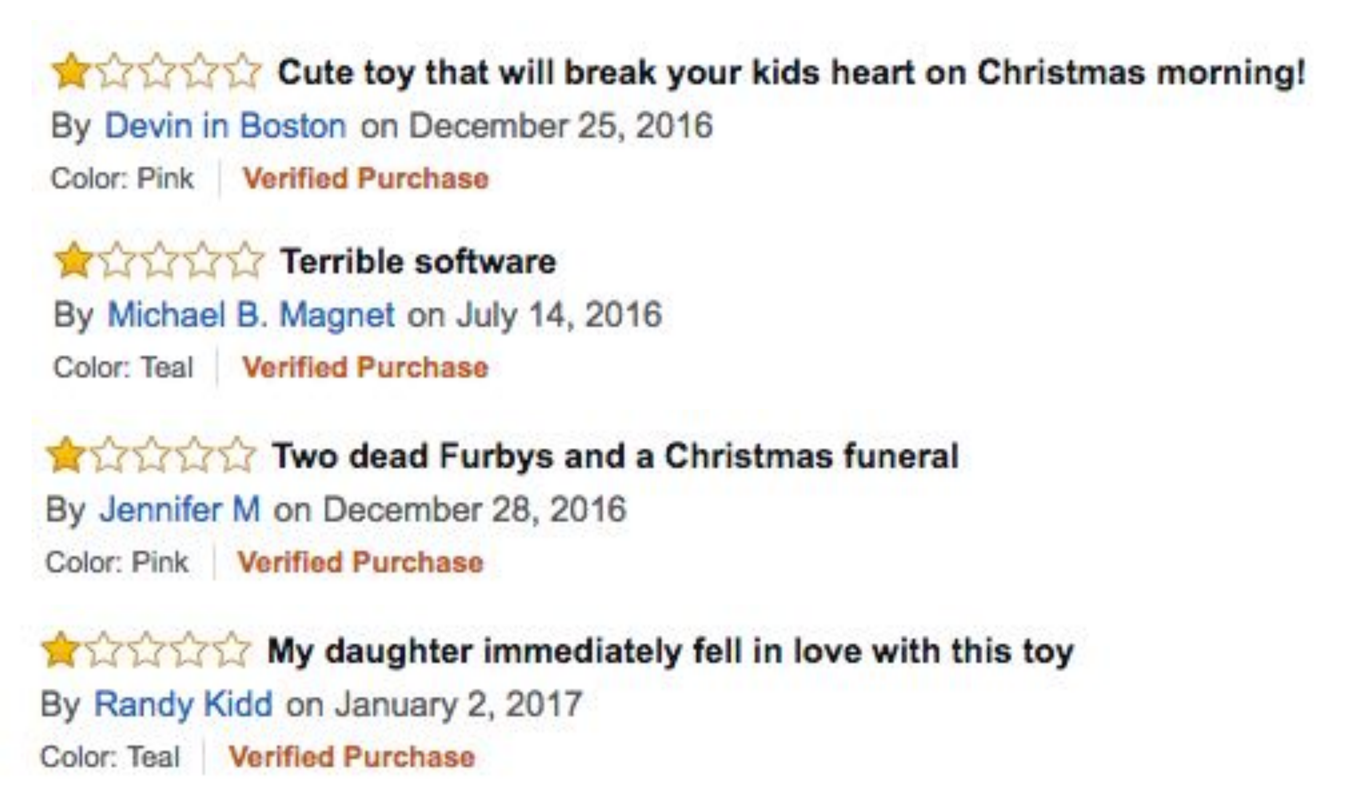
- Microphone
- Motion sensors
- IR Furby-to-Furby (F2F) comms
  - Controllable using universal TV remote
- No app
- No off switch
- Numerous spinoffs
- Banned from NSA facilities
  - Probably because of how f**ken annoying they are

Gen 2 – Furby 2012, Furby Boom

- First version with companion app
- Comms via high freq. audio
- Monochrome eye LCDs
- No OTA updates yet
- Still super annoying

Gen3 – Furby Connect

- **HOTTEST TOY OF 2016! BUY ONE NOW!**



★☆☆☆☆ **Cute toy that will break your kids heart on Christmas morning!**
By Devin in Boston on December 25, 2016
Color: Pink | Verified Purchase

★☆☆☆☆ **Terrible software**
By Michael B. Magnet on July 14, 2016
Color: Teal | Verified Purchase

★☆☆☆☆ **Two dead Furbys and a Christmas funeral**
By Jennifer M on December 28, 2016
Color: Pink | Verified Purchase

★☆☆☆☆ **My daughter immediately fell in love with this toy**
By Randy Kidd on January 2, 2017
Color: Teal | Verified Purchase

Gen3 – Furby connect continued

- BTLE comms with companion app
- OTA updates from internet
- Sleep mask for standby mode
- 3 generations of advancements in irritation
- Stupid cat videos
  - Furby interacts with content on app!
- The subject of our research

So we purchased two units

INKY

PINKY

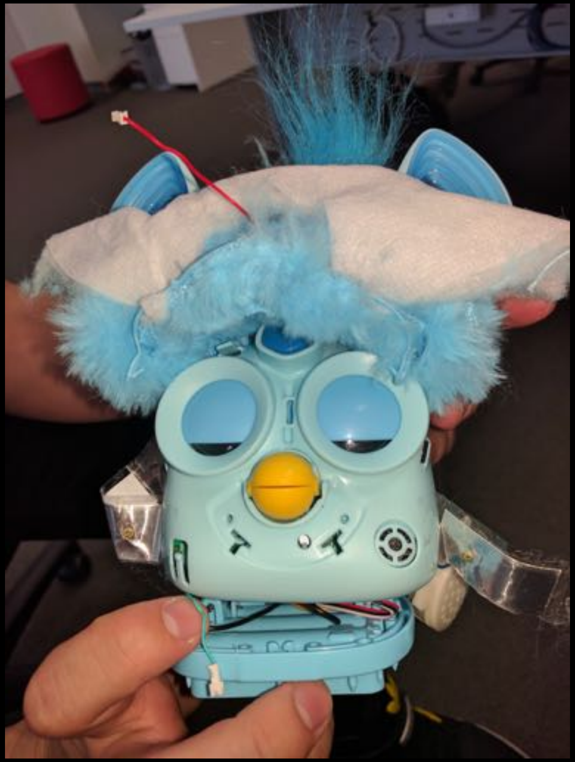
Stupid assumptions

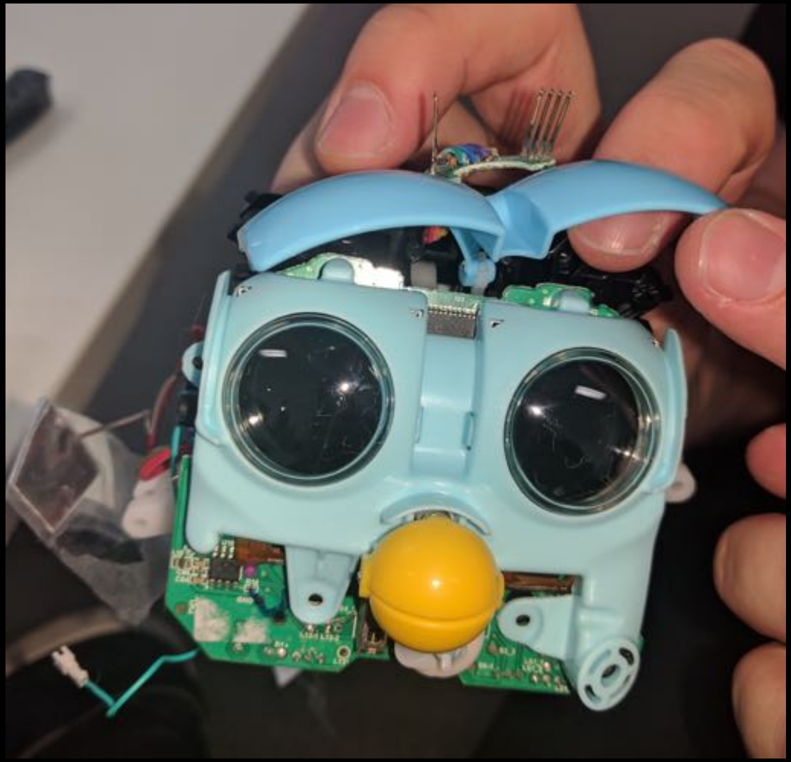
- ARM SoC or worst case something AVR based
- We’re going to have to sniff and crack BTLE
- We’re going to have to break the BTLE pairing process
- UART/JTAG pins easy to find
- Firmware easy to dump
- Firmware easy to RE
- We’re clever hackers

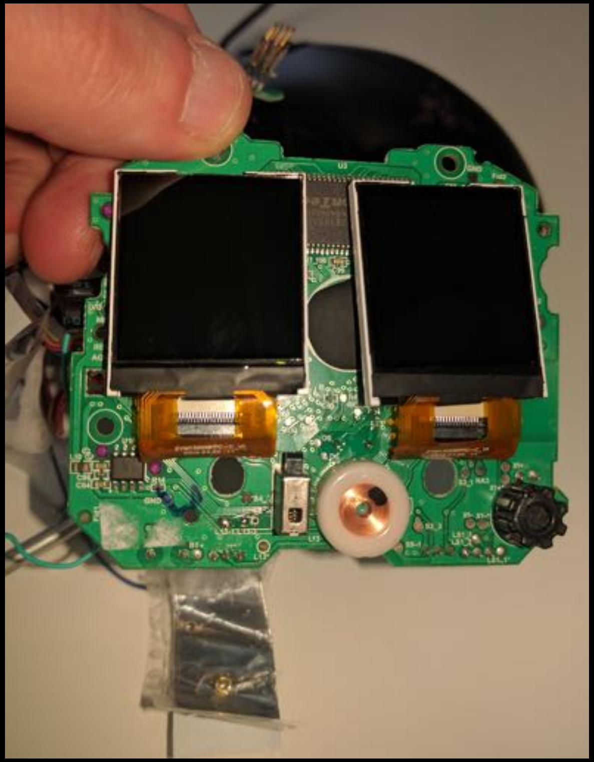
Less stupid assumptions

- Hasbro will have made some bad security decisions
- The app will get updates over HTTP/S and we’ll be able to MITM it

Getting started: JUST TEAR IT APART





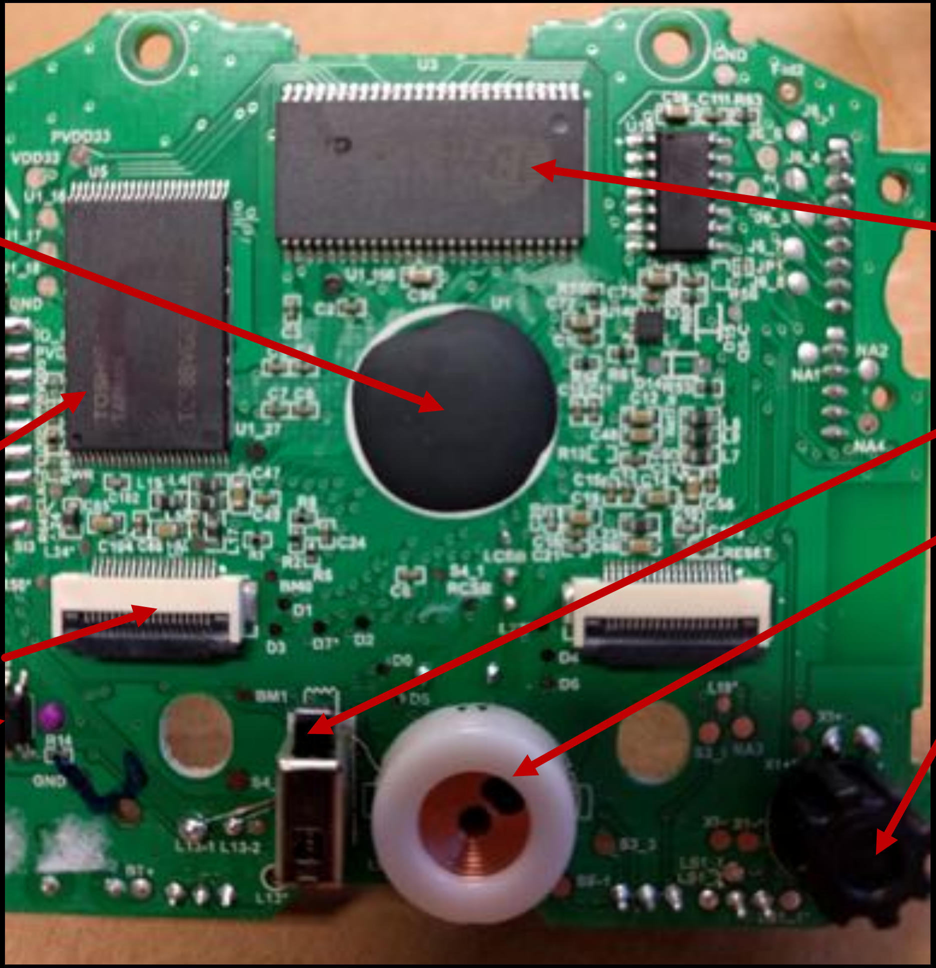


Fortunately Furby isn’t one to be deterred!

“Oooooh, I had a dream about Unai!”

– INKY, Dec 2016

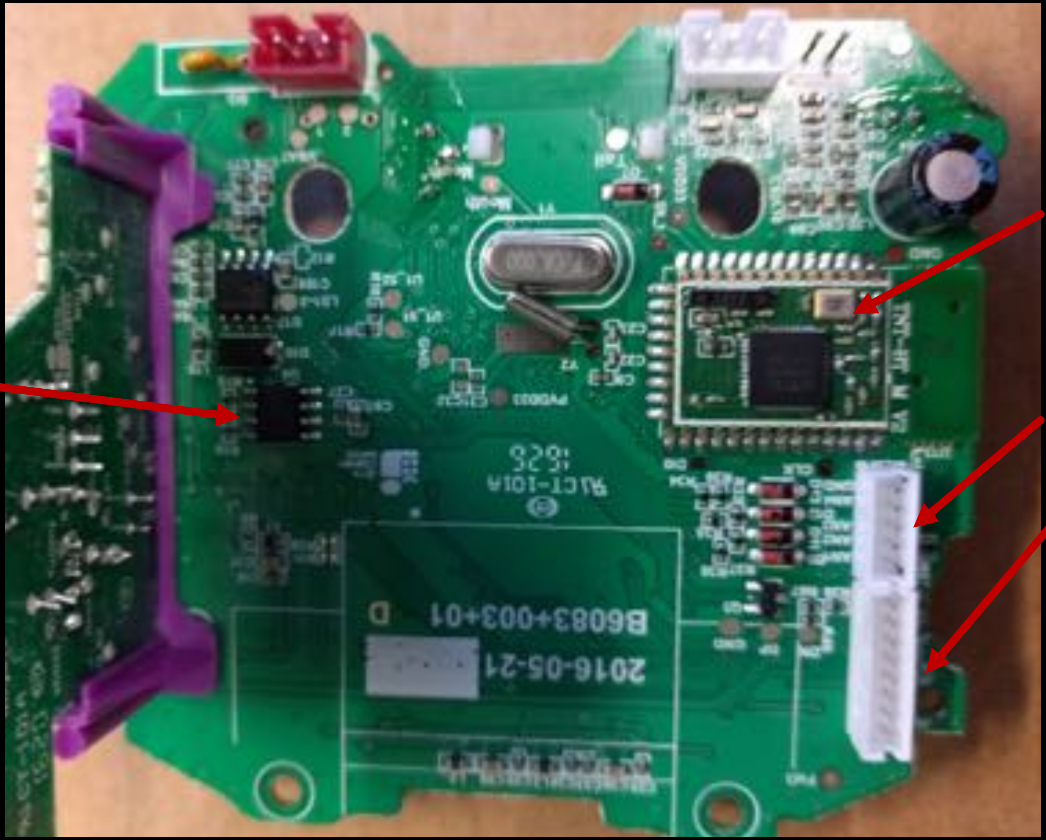
Main PCB - Front



- Generalplus CPU
  - GPL162004 - uses proprietary u'nSP instruction set.
- Toshiba flash – 1Gbit
  - TC58BVG0S3HTA00
- TFT FFC ZIF connectors
- Audio driver

- SDRAM – 128Mbit
  - EtronTech EM639165
- Tongue switch
- Electromagnet for 'beak' movement
- Microphone

Main PCB - Back

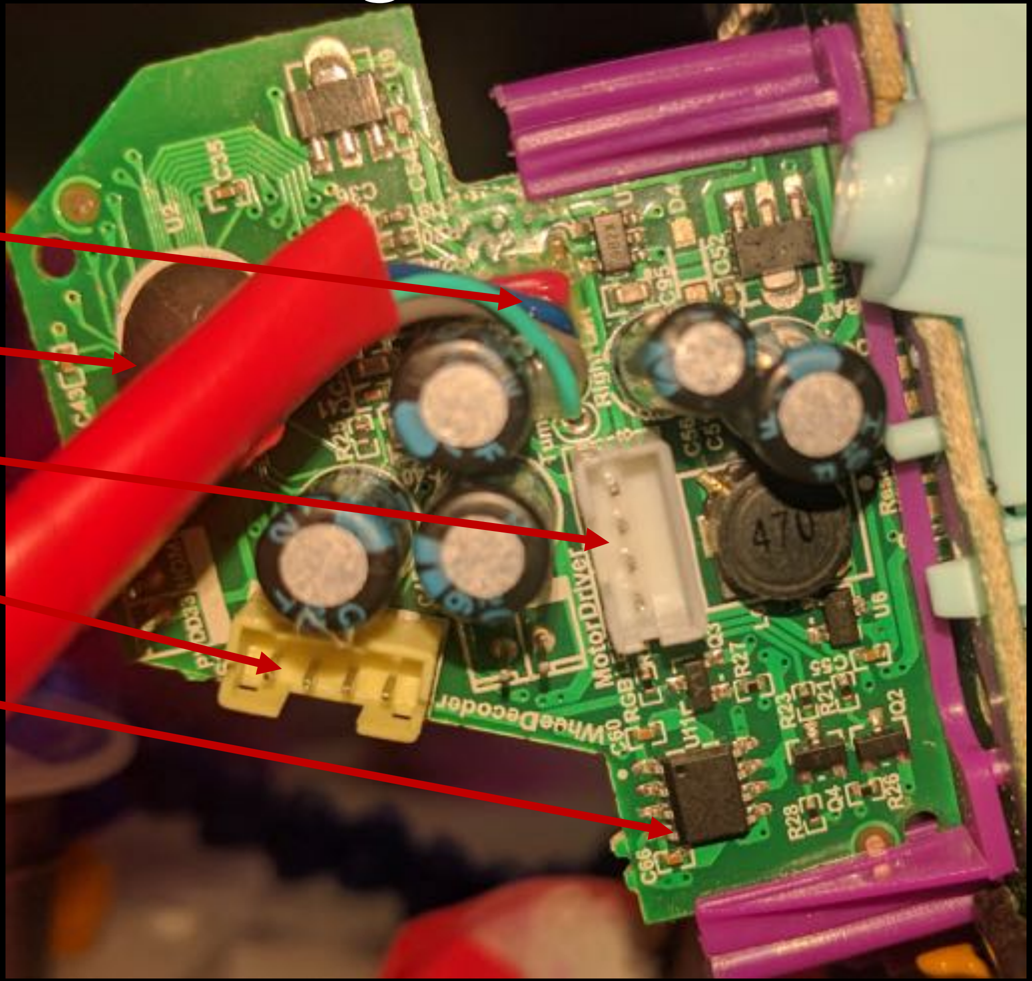


- Gplus SPI NAND
  - 128Mbit

- Nordic NRF51822
  - Off-the-shelf module?
- Antenna buttons
- Forehead socket/mask connector

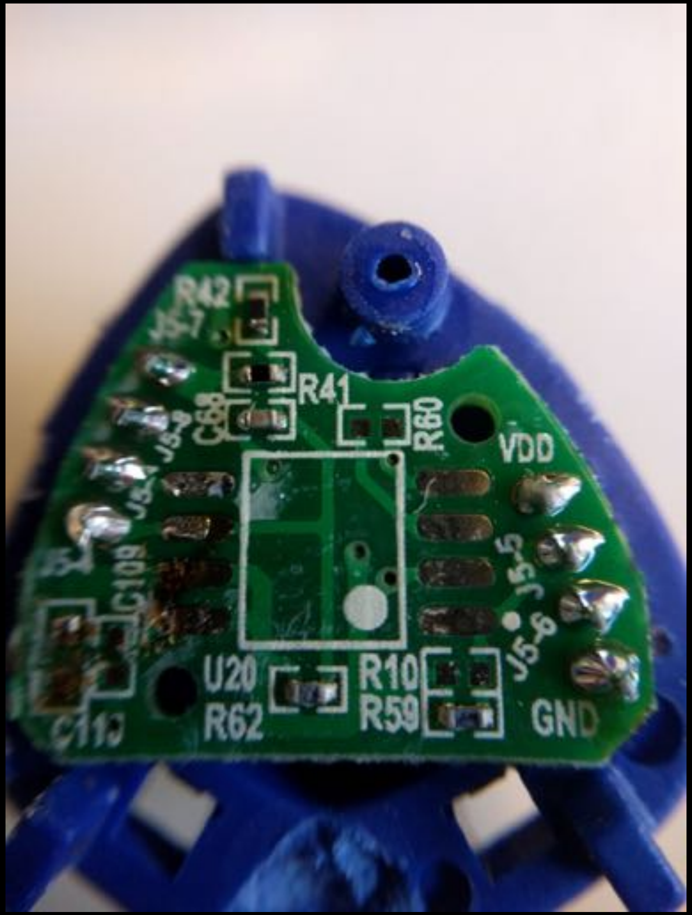
Daughterboard

- ‘Tickle Sensors’
- Accelerometer
- Antenna LEDs
- ‘Whee Decoder’
- Motor Driver



Sleep Mask PCB

- Makes Furby go to sleep!
- OTHERWISE HE'S PERMANENTLY AWAKE
- Will be used for future enhancements.



Comms

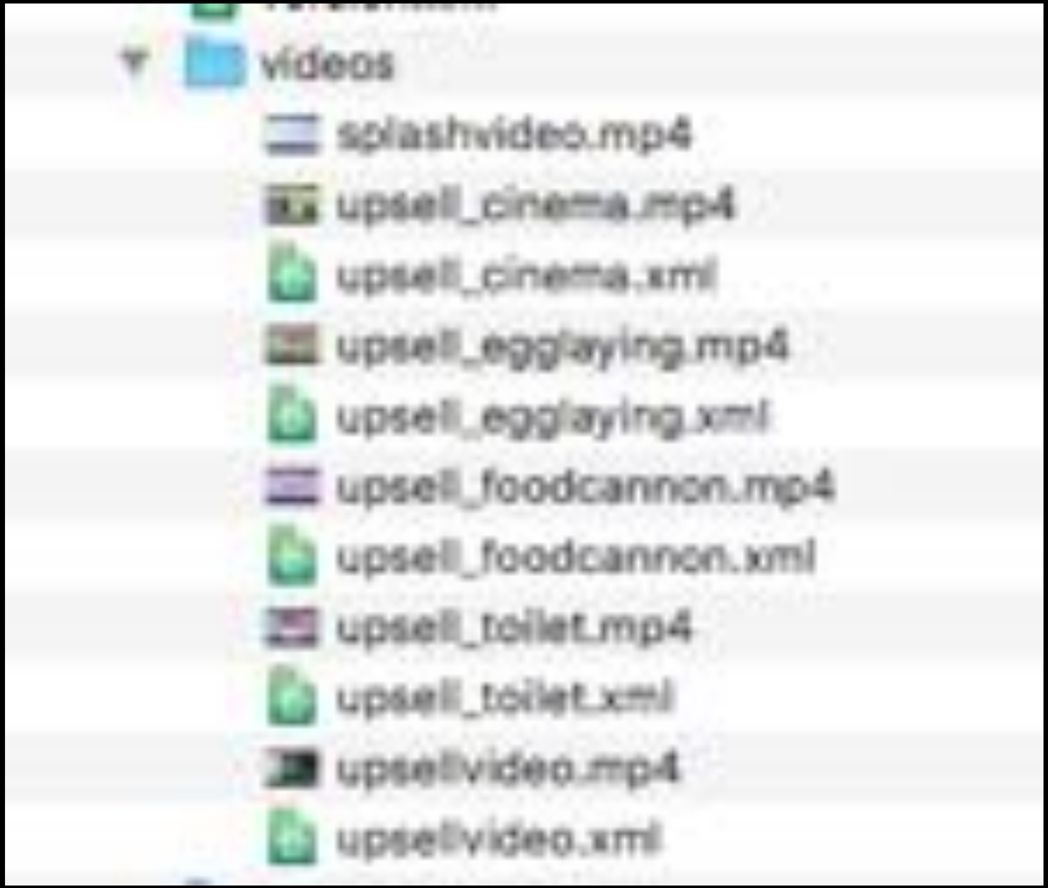
- App talks to remote servers
- Furby only talks to app or other Furbz via BTLE
  - F2F: Furb2Furb
- App provides new content and updates via BTLE

App/HTTP

- App uses HTTP(S) to communicate with update servers
  - Curiously, downloads content from "**chromecastpd**.s3.amazonaws.com"
  - Weird
- MitM was entirely possible thanks to lack of Cert pinning and HSTS in the app
  - Able to feed furby malicious DLC files

Reversing the android app

- Easy to do
  - unzip/dex2jar
- Some interesting files
  - libFluff.so – contains functions for BLE comms – written in C
- Some questionable naming conventions:

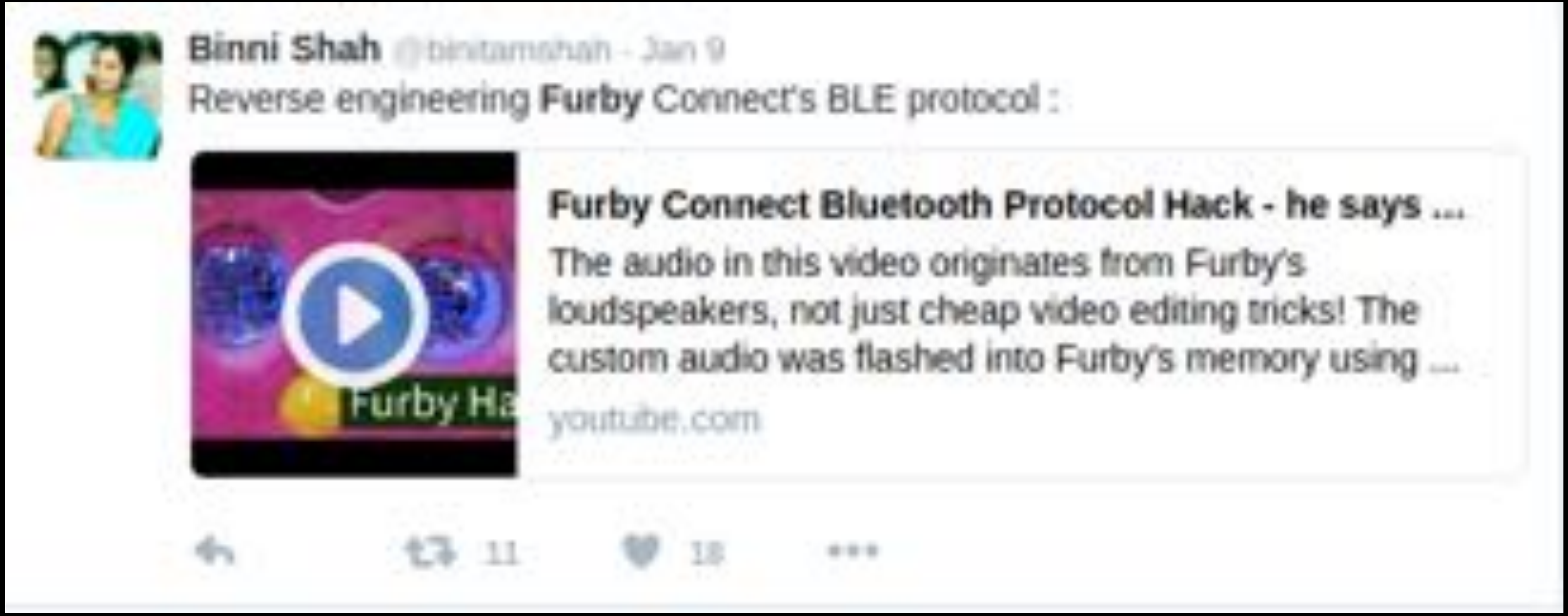


```
videos
    splashvideo.mp4
    upsell_cinema.mp4
    upsell_cinema.xml
    upsell_egglaying.mp4
    upsell_egglaying.xml
    upsell_foodcannon.mp4
    upsell_foodcannon.xml
    upsell_toilet.mp4
    upsell_toilet.xml
    upsellvideo.mp4
    upsellvideo.xml
```

Reversing the firmware/DLC

- Content is served in DLC files:
  - TU003410.DLC (English)
  - TR002790.DLC (Russian)
  - FU001680.DLC (Firmware Patch)
- Binwalk?
  - Nope
- Strings?
  - Nope
- Staring at hexdumps?
  - Apparently not good enough

And then one day on Twitter…



Binni Shah @binitamshah · Jan 9

Reverse engineering **Furby** Connect's BLE protocol :

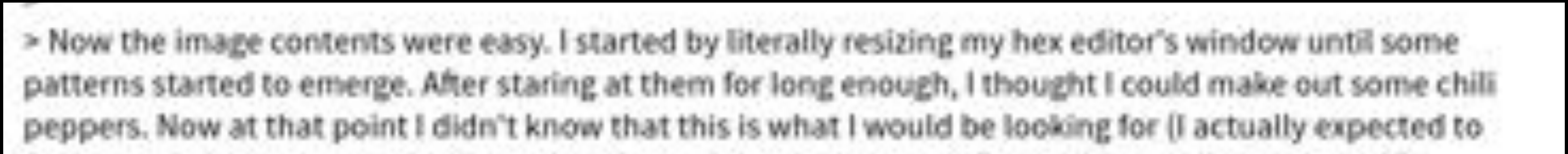
Furby Connect Bluetooth Protocol Hack - he says ...

The audio in this video originates from Furby's loudspeakers, not just cheap video editing tricks! The custom audio was flashed into Furby's memory using ...

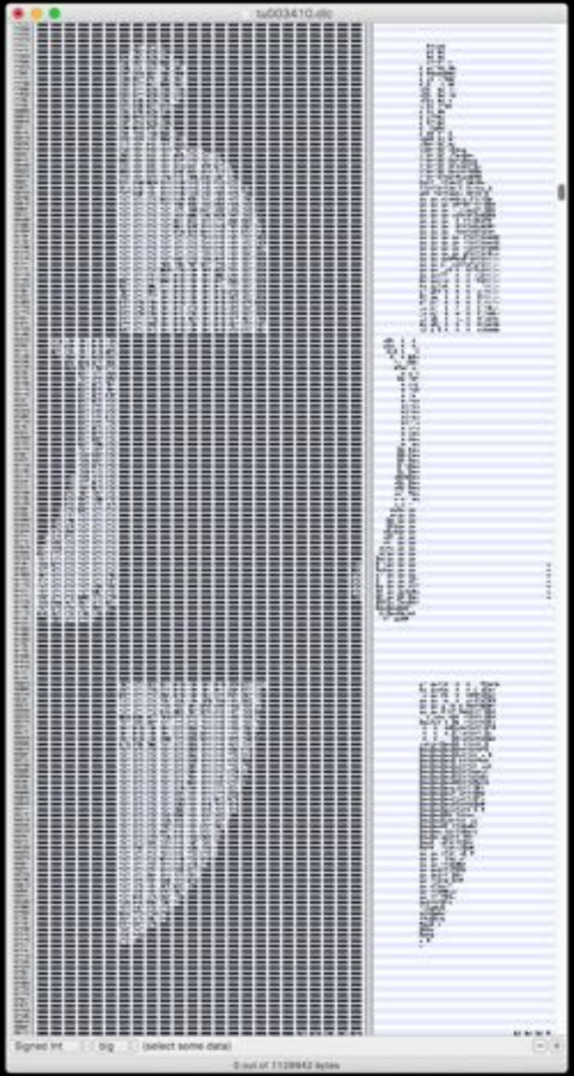
youtube.com

Ze Germans

- As usual someone beat us to it
    - German CS student (Florian Euchner) - bored on summer break
    - Too nice to hate!
    - Smashed out app to control Furb
    - Check out his GitHub: https://github.com/Jeija/bluefluff
        - Seriously, it owns despite being Node.js
- Also as usual, all of our hard efforts trying to methodically smash the furb were bested by a dude eyeballing hexdumps:

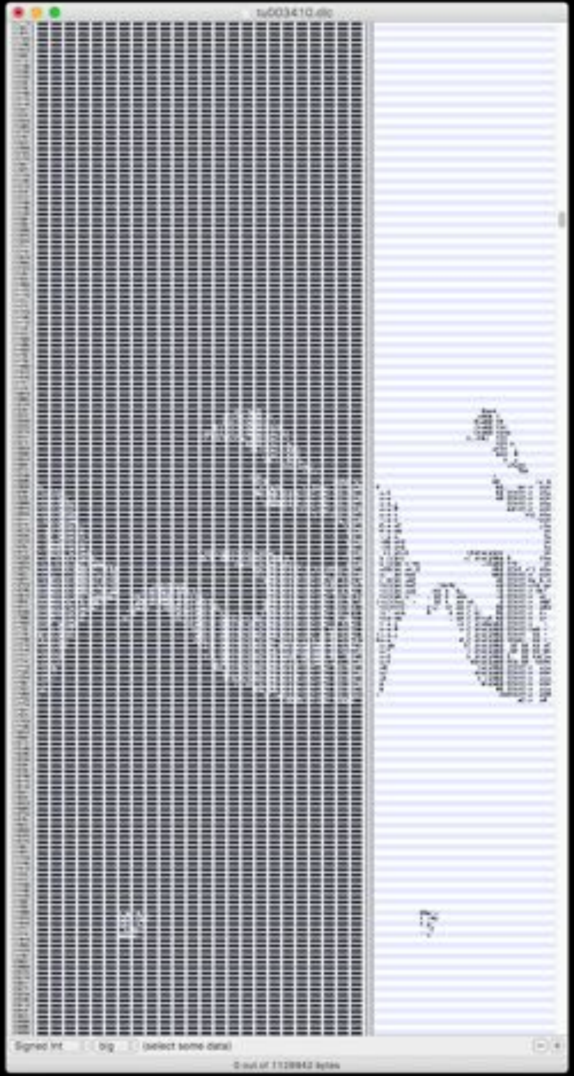


> Now the image contents were easy. I started by literally resizing my hex editor's window until some patterns started to emerge. After staring at them for long enough, I thought I could make out some chili peppers. Now at that point I didn't know that this is what I would be looking for (I actually expected to



← Chillis

Flames →



Tiling algorithm

- The tiling system is… weird
- Tiles are (as previously mentioned) 48*16 byte chunks which converts to 64*16 pixel images
- The tiles are arranged in blocks of 8 with the following format…

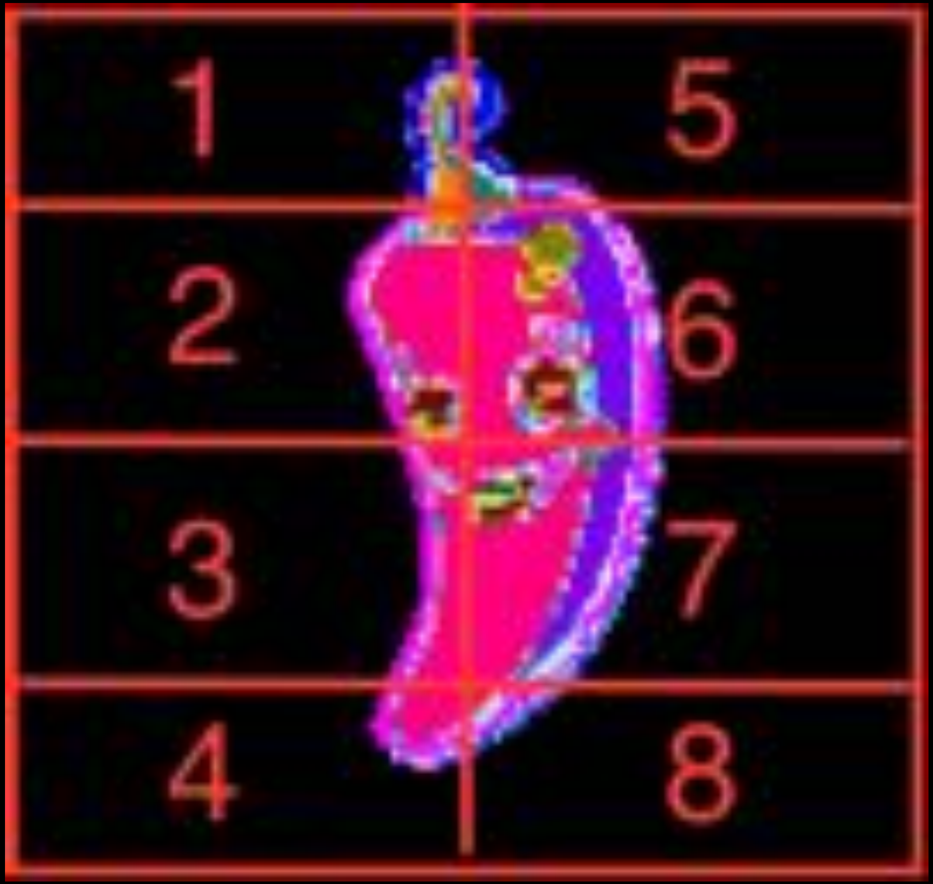
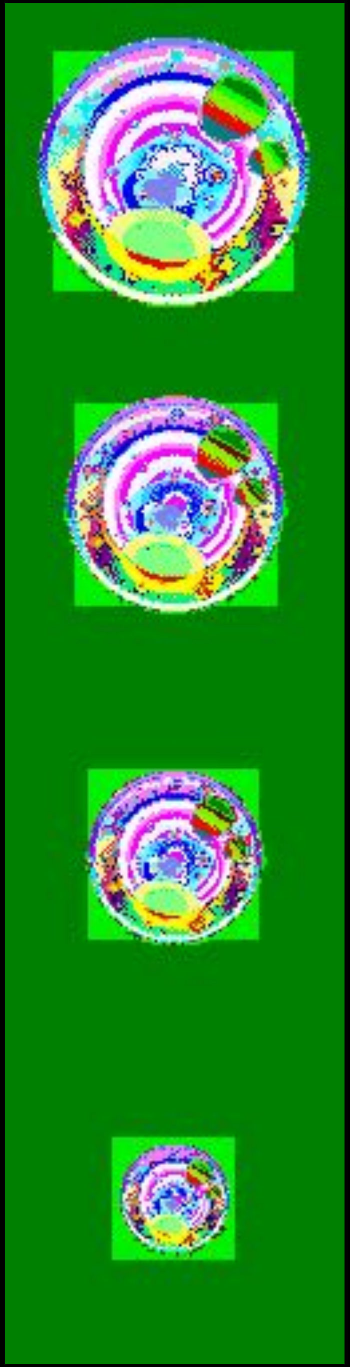


Image format

- Eyes
  - 64*16 pixel arrays (48*16 bytes = 768 bytes)
  - two pixels = 3 bytes/6 nibbles
  - Colour is then RGB of 3 nibbles (1 nibble per colour)
  - Background == 041 (Green?)
- Sprite overlays
  - Background == 000 (Black?)
- Proprietary shit with a custom colour palette?
  - Yet to decipher the palette 😐



<- Eyes

Chillis! ->

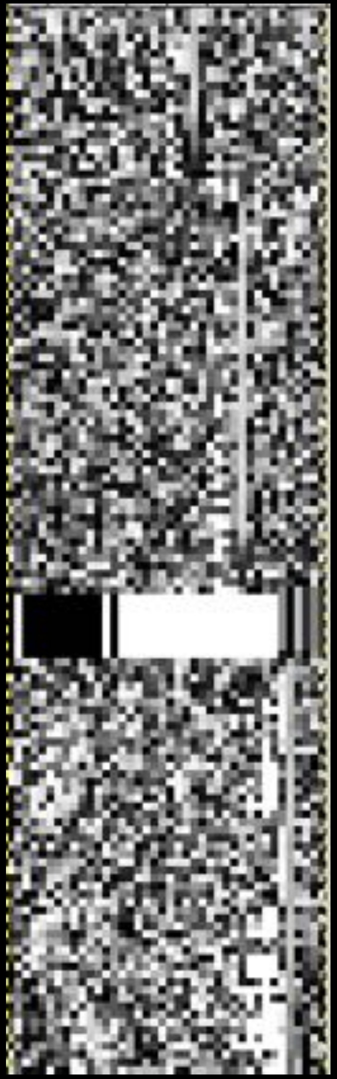
Flamez ->

More about the files

- DLC files
  - Contain images, audio, actions, etc.
  - Reactions to companion app content i.e. cat videos
- Firmware
  - Contains other ‘personalities’
    - Files contain references to ”Pirate”, “DJ”, ”Princess” etc.
    - Appears the plan is to use the mask to enable new personalities at a later date
    - Relevant audio/video already on the Furby from the factory. Day 0 DLC every time…

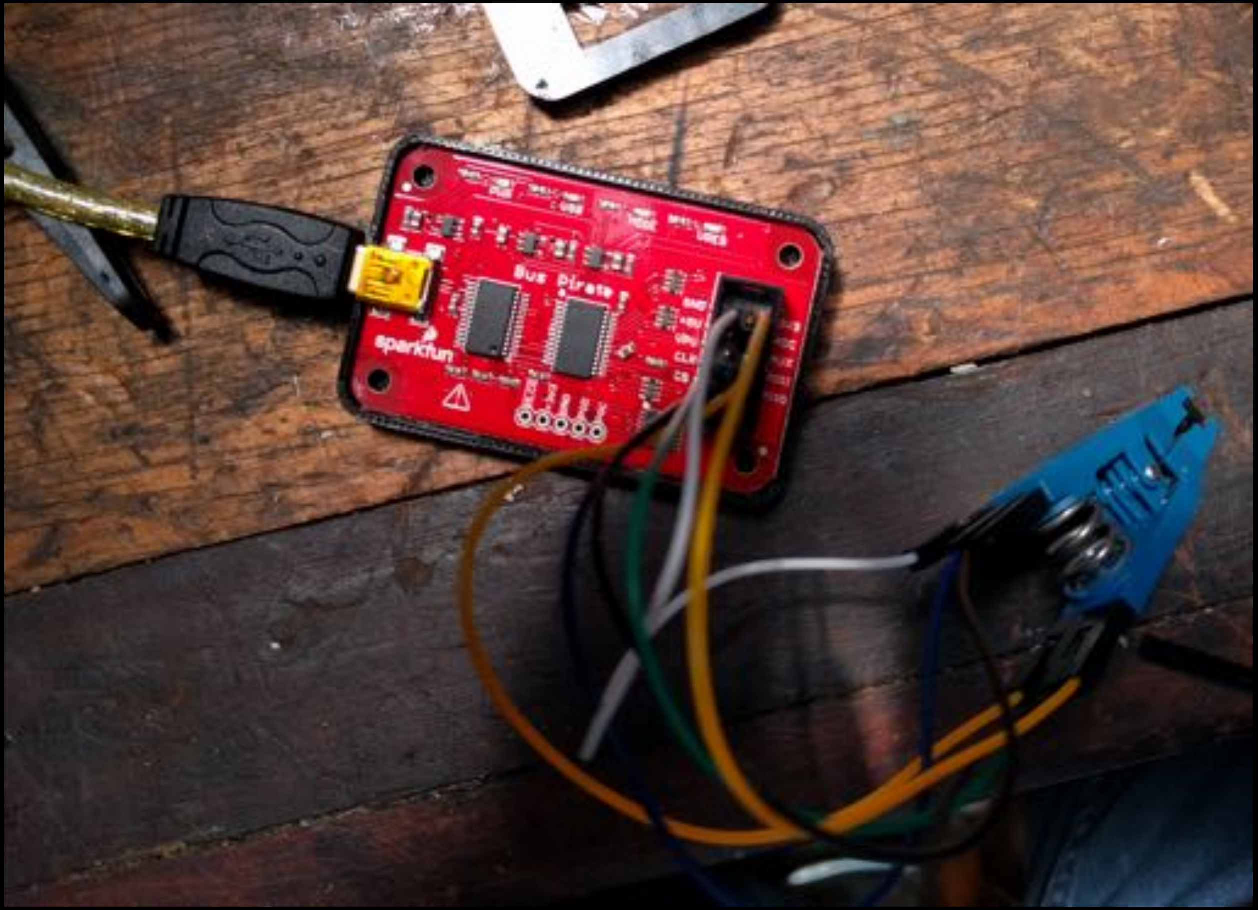
Audio format

- A1800 audio codec
- Basically a terrible, proprietary FFT-style thing
- Again, has a distinctive visual pattern if you know the right way to stare at the hexdumps…



~ Dumping flash with loss and swarley ~

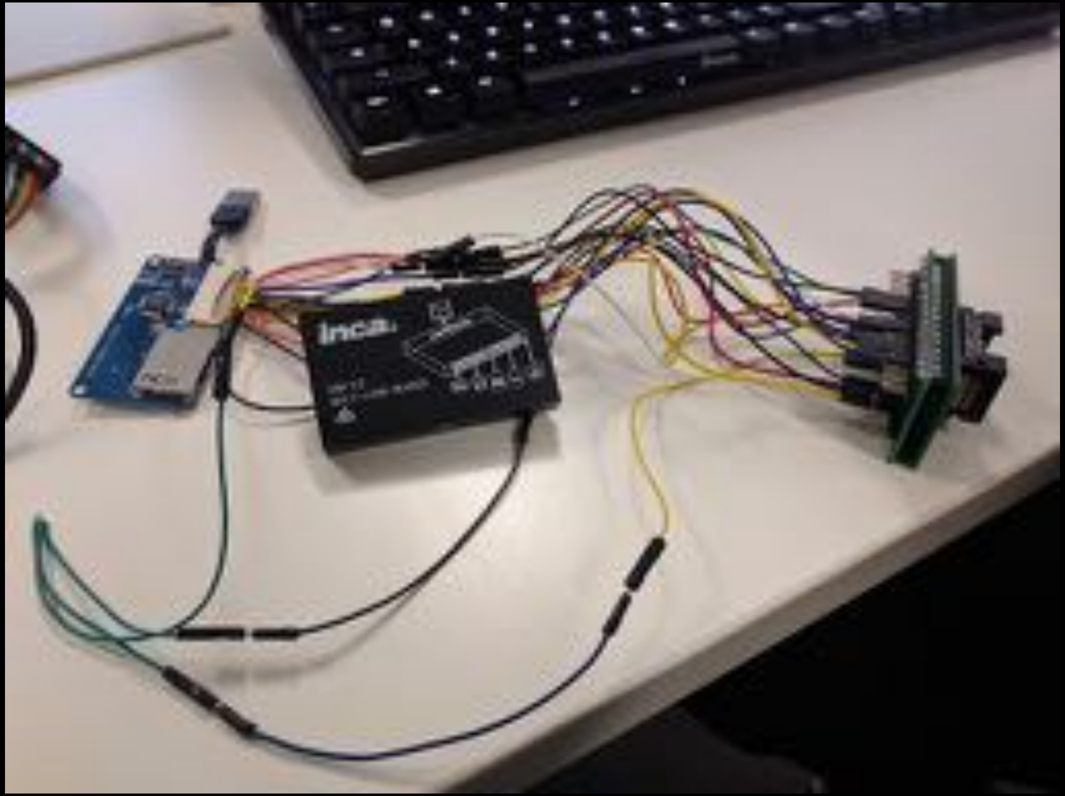
“JUST USE A BUS PIRATE”
– Everyone ever



- This actually worked for the SPI flash chip with the bootloader on it.
- u’nSP bytecode not suuper readable
- Helped ID main SoC
- Small comfort – no good for TSOP48 NAND.
  - Insufficient I/O.

"It's easy. An XD Card reader will do it!"

- Some asshole on a console mod forum



- Rationale: XD cards are basically raw NAND which the XD card reader takes care of interfacing with the host
- ...yeah of course that one didn't fucking work

"JUST JTAG IT"

– Yahoo Answers

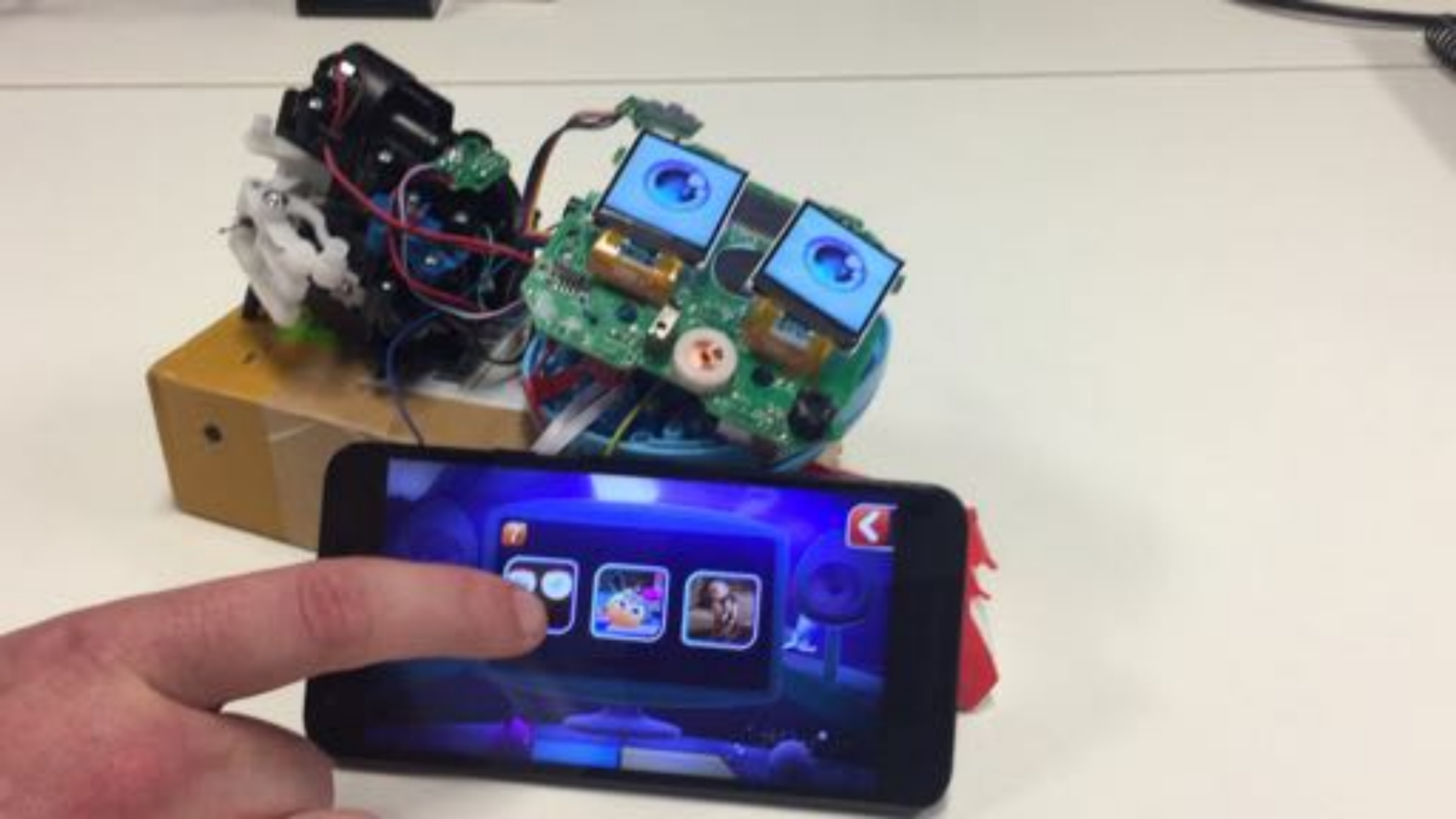
Solution: **Throw money at the problem!**

- AGE OF ENTITLEMENT IS OVER, OK
- Chinese Universal Programmer did the trick (RT809H)

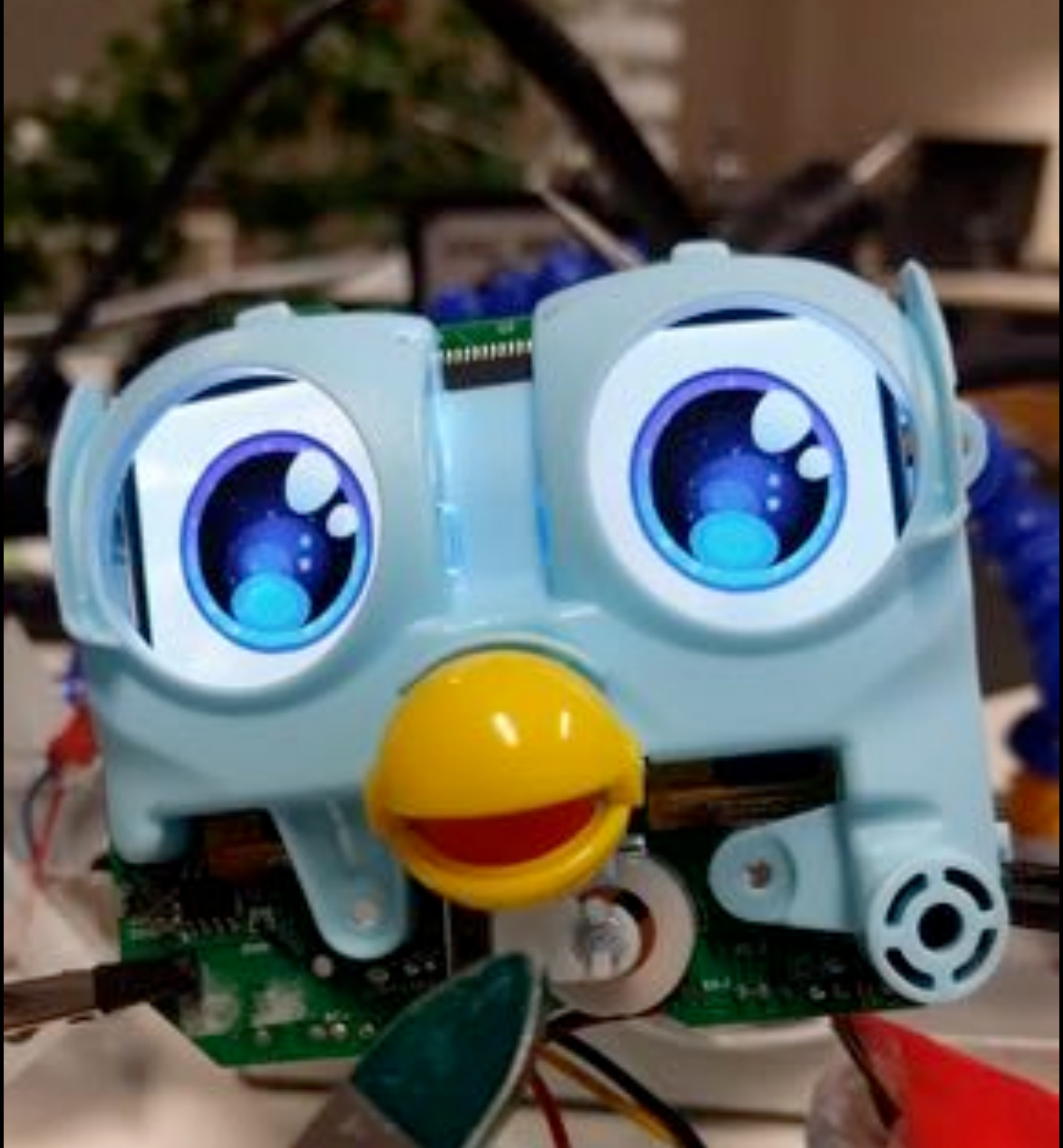
YOU’LL NEVER BELIEVE WHAT WE CAN DO WITH THIS

[WARRANTY RETURNS]

[PETRIFY CHILDREN]



[MINIMUM VIABLE DEMO]



Implications

- There's no security about the pairing process
- In fact there is no pairing process
  - You just need to know a Furby's UUID to send it data
  - Furbies broadcast their UUID constantly
- Can broadcast content to any and all Furbs within range
- We actually managed to destroy Pinky's brain during the process
  - Suspect DLC file wrote out of bounds of allocated DLC memory slot and overwrote something important…
  - May point to the possibility of writing a DLC file long enough to overflow into actual code.
- Remotely brick every Furby within range after showing naughty videos?

Lessons Learned

- Visualising binary data in simple ways is super effective.
- If there’s a correct tool for the job (and you can afford it) just buy it.
- Not everything ‘IoT’ is *nix on ARM or AVR

Conclusions

- Hardware hacking is HARD
- Furbies are SUPER ANNOYING
- God bless the Germans
- We STILL have no idea how to hack a Furby Connect.

ACK

- Dave Taylor – The Boss
- Florian Euchner – Ze Germans
- snare
- liam
- eon
- kronicd
- anyone unfortunate enough to have sat through hours of Furby audio during this process.

Questions?

- RIP INKY… ☹