

Free drivers for Oculus Rift headsets

- Jan Schmidt <jan@centricular.com>
- [@thaytan](#)

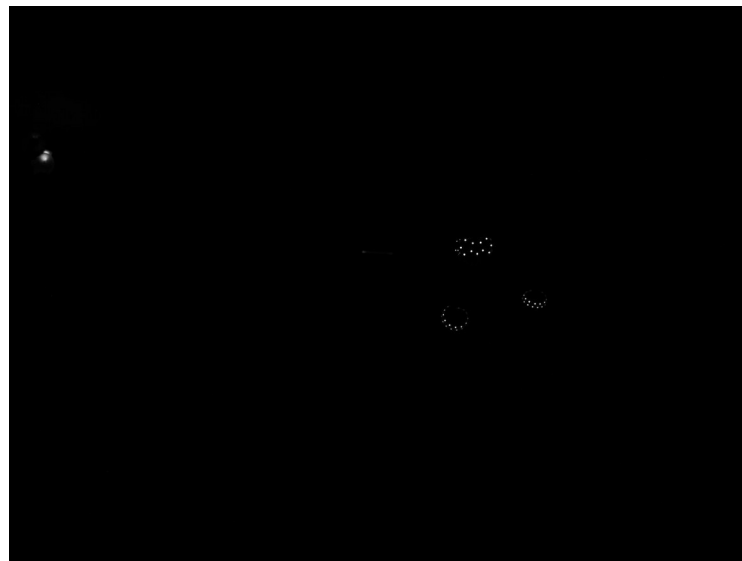
Oculus Headsets

- Rift DK2 / CV1
 - “Outside-In” tracking
- Rift S
 - “Inside-out” tracking
- **Not Quest / Quest 2**



Constellation System (CV1)

- Camera sensors see IR
- LED models from firmware
 - headband adjustments, occlusion mean they don't match
- LEDs are pulsed in sync with the camera
- Track IR blobs... extract poses
- Need to know camera poses



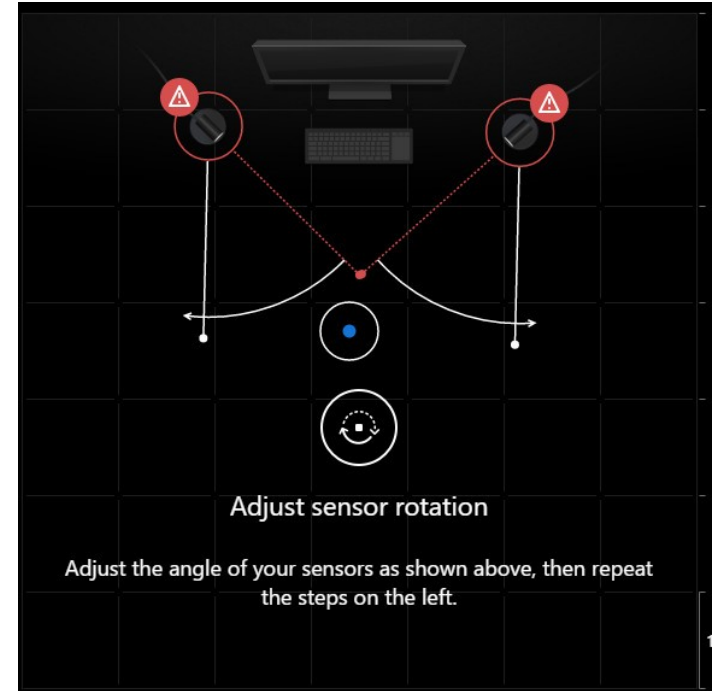
Camera Poses

- Need to know the camera positions
- Work backward from views of the headset
- Run once each time the configuration changes



Gives an $[x,y,z]$ position + $[w,x,y,z]$ quaternion for each camera

Can do online estimation

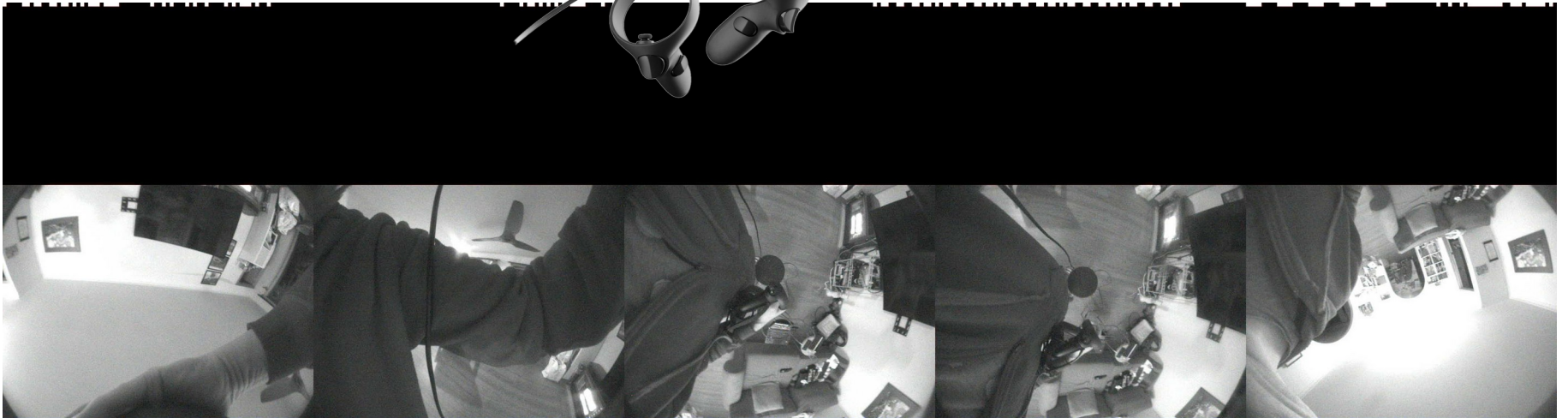


CV1 Details

- Mostly a UVC camera, with quirks
 - 52.0833 FPS (19.2ms / frame)
 - No Linux kernel support for Variable length controls
 - UVC in userspace = scheduling problems
- Frame exposure synchronisation
 - HMD ↔ Controller radio link
- Match up HMD exposures with capture
 - Based on frame arrival times and IMU sample times
 - Frames from different cameras have different arrival times though!

Constellation System (Rift S)

- 5 cameras on the headset
- Fewer LEDs = easier
 - Only the controllers
- But the cameras move
 - The SLAM/VIO is the hard part
 - (later)

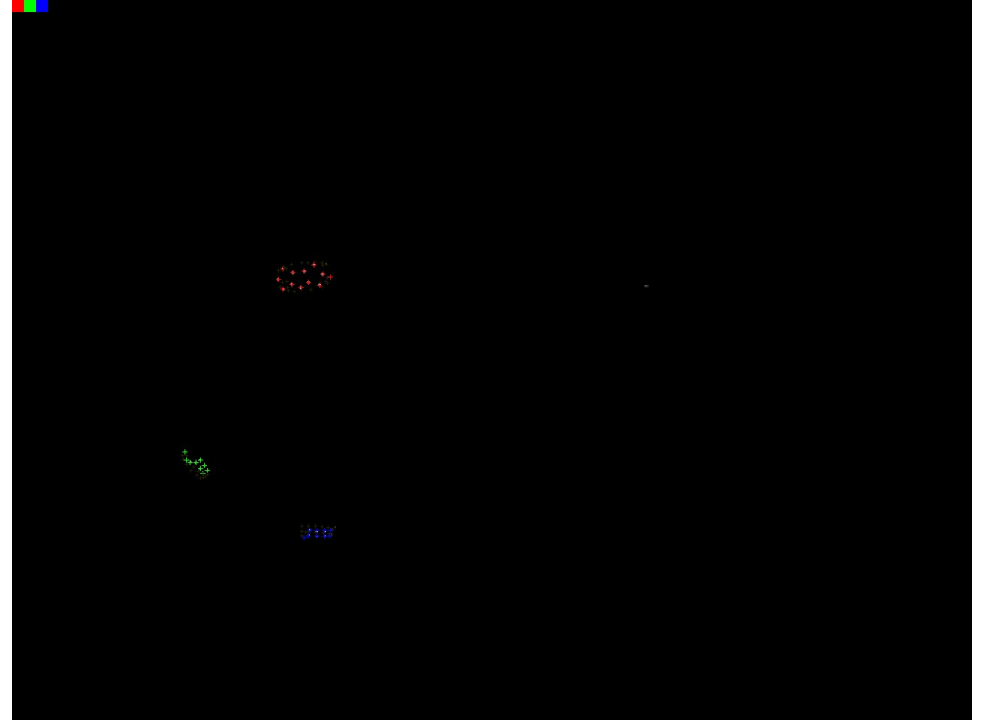


Correspondences

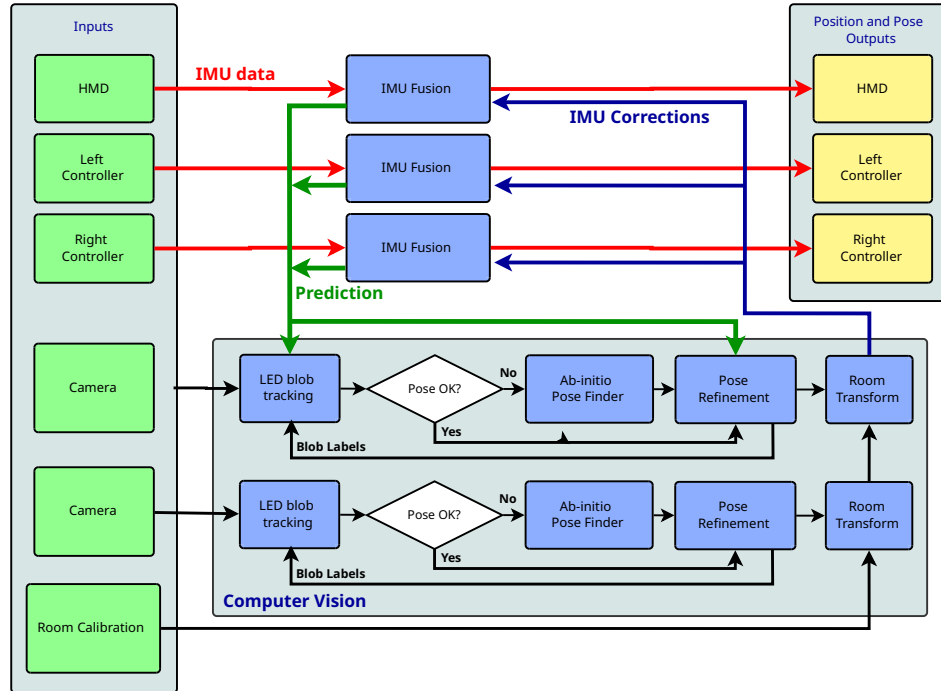
- Matching which blob is which LED. Home-brew depth-first search
- Pre-sort LED model positions based into lists of proximate neighbours
- Sort observed blobs by proximity
- Match groups of 4 LEDs to 4 blobs. Extract pose using LambdaTwist P3P and validate 4th point – then assess ‘pose score’
- Score based on expected matches in the bounding box / visibility of LEDs
- Two-pass strategy for big speed increase (test only nearest LEDs first)

Using the IMU

- 3DOF tracking
 - Let's us align gravity vectors
 - Reduces the viable correspondences
- Can do better – 2-point correspondence + gravity
- Can do even better... 6DOF fusion



IMU+Vision Fusion



Latency

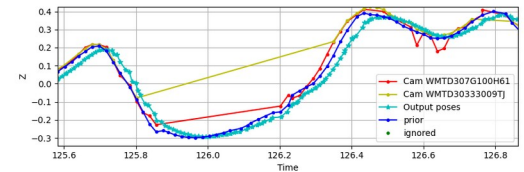
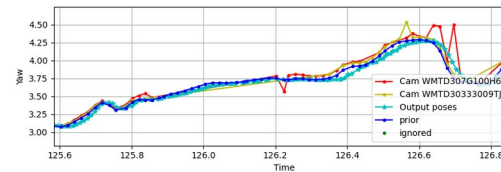
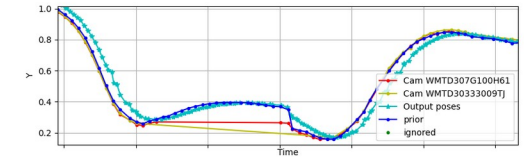
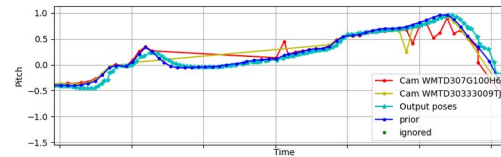
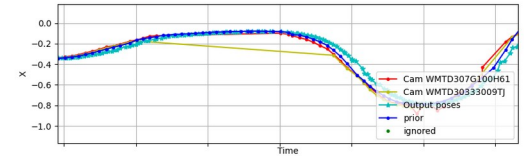
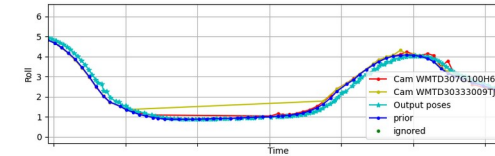
- Frames start arriving every 19.2ms
 - USB transfer time ~17-18ms
- Image processing takes time
 - JPEG decode (for USB 2.0), 2-3ms
 - Blob extraction, RANSAC, 1-10ms
 - Correspondence search – **can be over 100ms**
 - (but more often < 40ms)
- IMU fusion is very quick
 - has to be less than 1ms

Kalman Filtering

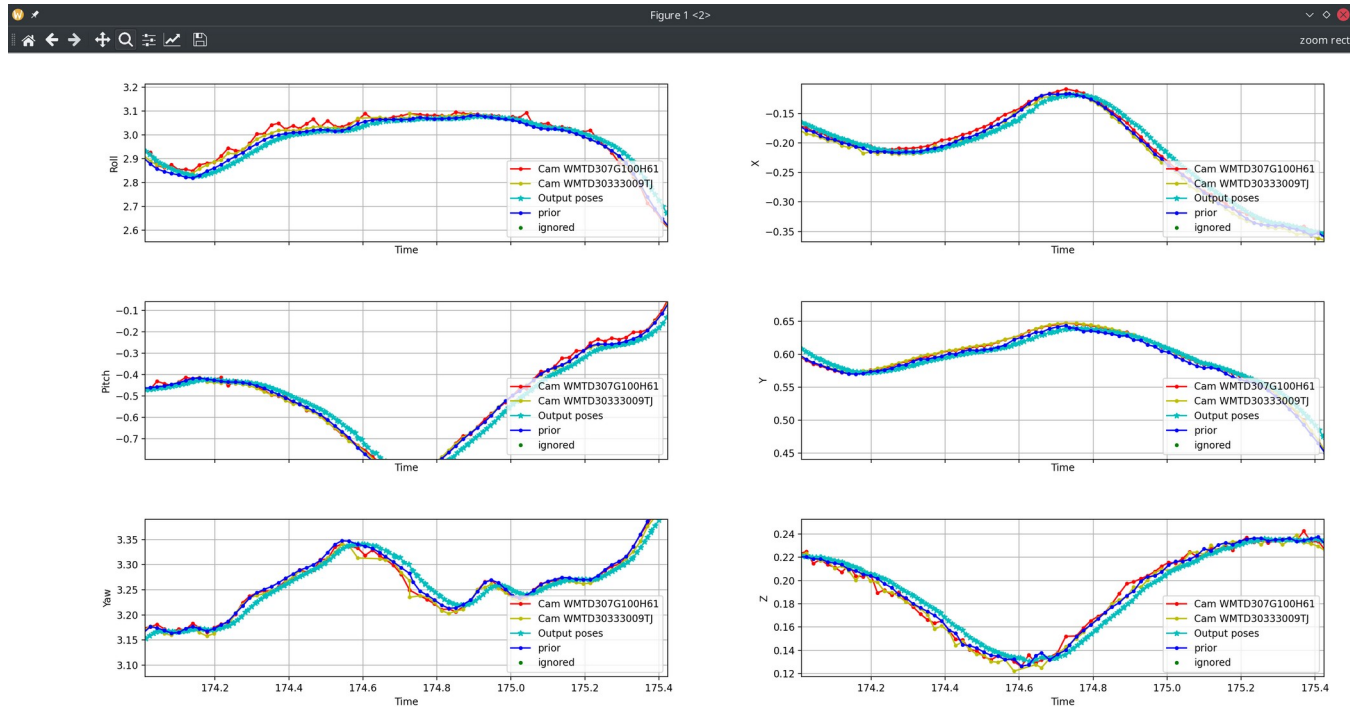
- Improved sensor fusion
 - Unscented Kalman Filter
 - Tracks position, rotation, extracts IMU biases
 - “Slots” for lagged position updating.
- Pretty expensive
 - Runs every 1ms for the headset, 2ms for controllers
 - Could perhaps run at camera rate and predict in between?

Avoiding Glitches

- Extracted poses aren't always right
 - Mis-identified LEDs
 - Room for improvement
 - RANSAC PnP flakiness
- Prediction time limited when tracking is lost
- 1€ exponential filter for smoothing reported pose

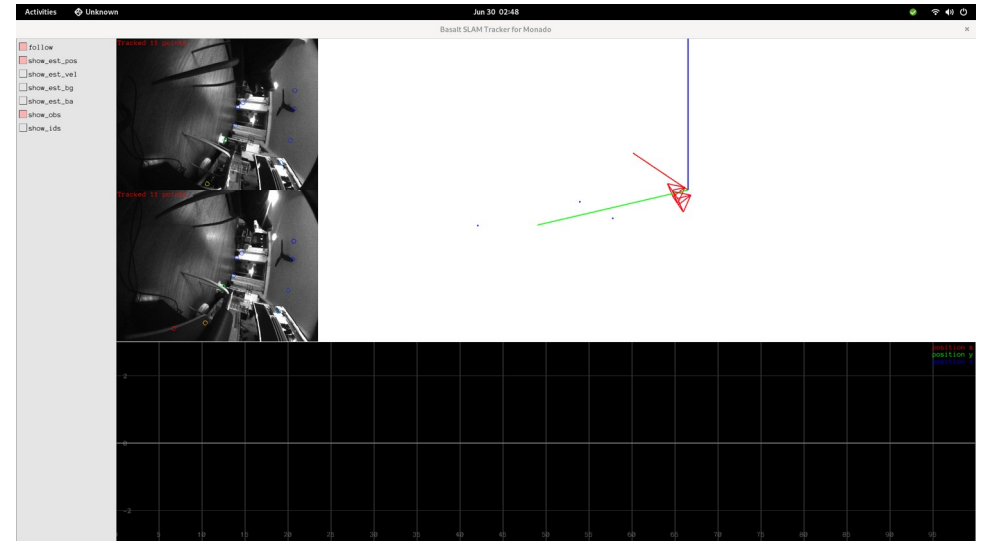


Good Tracking



Rift S Inside-out SLAM/VIO

- Monado, Basalt
- Exposure compensation
- Distortion compensation
 - Native “Fisheye62” model
 - Basalt conversion
- Attach the calculated pose to each frame



Rift S Controllers

- Need the camera pose to predict controller LEDs
 - From the previous interleaved frame + prediction
 - SLAM better keep up (prediction error directly affects controller jitter)
- Controllers might cross view boundaries
- Unlike CV1, camera frames all arrive together



Future Directions

- Fusion performance improvements
 - IMU integration, fusion at camera rates
 - Explore optimisation approaches
- Improve pose extraction
 - Better blob position refinement
 - Figure out OpenCV ransac glitches
 - ML approaches to correspondence?
- Continue simulator / replay work
- Controller tracking for Rift S + WMR

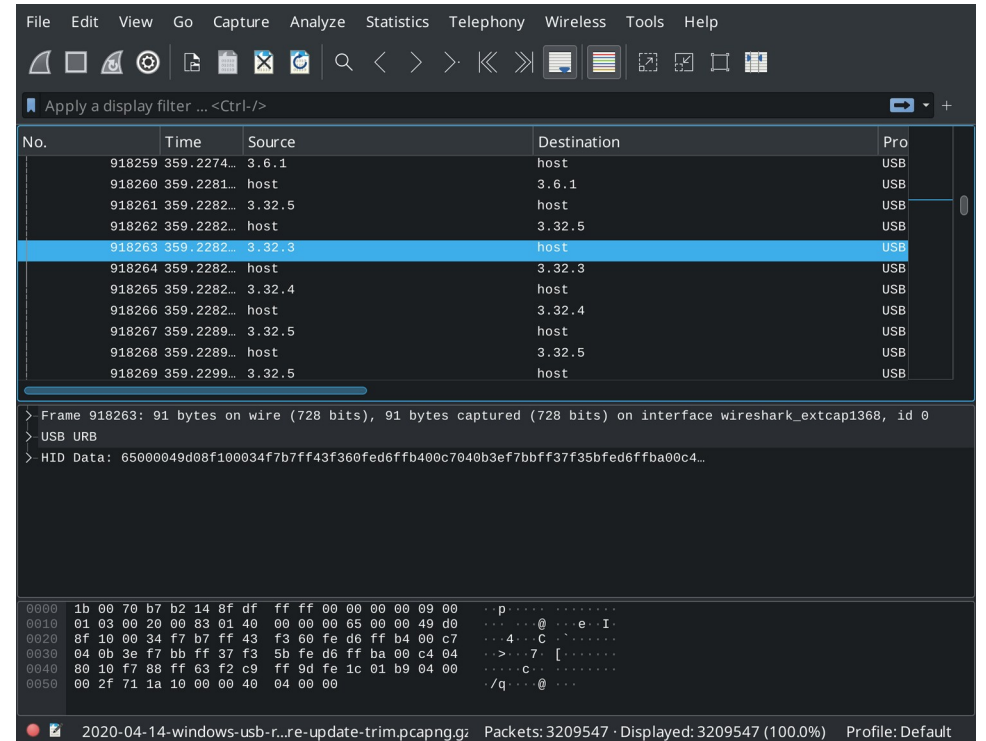
Protocol Reversing

Protocol Reversing

- Sources of information:
 - Code decompiling
 - USB packet captures
- Either could be a breach of the EULA
- But still might be legal in your jurisdiction
- Log files from the official software can be enlightening

USB packet capture

- Wireshark + USBpcap on Windows
- Find which USB root device the port is on first
- Great to capture the first connect
 - Usually a firmware update
 - Capture without fw update too



```
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol
918259 359.2274... 3.6.1 host USB
918260 359.2281... host 3.6.1 USB
918261 359.2282... 3.32.5 host USB
918262 359.2282... host 3.32.5 USB
918263 359.2282... 3.32.3 host USB
918264 359.2282... host 3.32.3 USB
918265 359.2282... 3.32.4 host USB
918266 359.2282... host 3.32.4 USB
918267 359.2289... 3.32.5 host USB
918268 359.2289... host 3.32.5 USB
918269 359.2299... 3.32.5 host USB
> Frame 918263: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface wireshark_extcap1368, id 0
> USB URB
> HID Data: 65000049d08f100034f7b7ff43f360fed6ffb400c7040b3ef7bbff37f35bfd6ffba00c4...
0000 1b 00 70 b7 b2 14 8f df ff ff 00 00 00 00 09 00 . . p . . . . .
0010 01 03 00 20 00 83 01 40 00 00 00 65 00 00 49 d0 . . . @ . . . e . I .
0020 8f 10 00 34 f7 b7 ff 43 f3 60 fe d6 ff b4 00 c7 . . . 4 . . C . . . .
0030 04 0b 3e f7 bb ff 37 f3 5b fe d6 ff ba 00 c4 04 . . > . . 7 . [ . . . . .
0040 80 10 f7 88 ff 63 f2 c9 ff 9d fe 1c 01 b9 04 00 . . . . c . . . . .
0050 00 2f 71 1a 10 00 00 40 04 00 00 . . / q . . . @ . . .
```

2020-04-14-windows-usb-r...re-update-trim.pcapng.gz Packets: 3209547 · Displayed: 3209547 (100.0%) Profile: Default

USB packets

- Setup via HID GET/SET reports is normal.
- IMU on USB interrupt
- Controllers / radio traffic
 - Often on another USB interrupt endpoint
- Isochronous in for camera
- Isochronous out for audio

Looking for patterns

- Common operations
 - Turn on the screen
 - Enable IMU
- Known values in hex dumps
 - Screen resolutions, physical dimensions
 - Floating point values
- Take lots of notes

```
6 (0x06) - Get the display configuration
display info:
  06 a0 05 00 0a 01 00 50 fe 03 ef 01 d2 00 5a 00 | .....P.....Z.
  0c 00 01 00 02 00 | .....
Vertical: 1440 (0x5a0) Horiz: 2560 (0xa00) Hz: 80
Unk0: 1 Unk1: 32441342 Unk2: 5898450 Unk3: 65548 Unk4: 2

0x5a0 = 1440      0xa00 = 2560
0x001             0x50 = 80 Hz
03fe = 1022      01ef = 495
00d2 = 210       005a = 90
000c = 12        0001             0002
```

```
9 (0x09) = GET REPORT ID 9 (response len 21)
- imu config

0010                                     09 e8 03 00      ....
0020    00 6f 12 03 42 00 00 80 45 66 66 a3 43 00 00 c8    .o..B...Eff.C...
0030    41

- Sent right after enabling the HMD with 02 01
- 0x000003e8 = 1000 (us between accelerometer readings, or 1000Hz?)
- 0x4203126f = 32.768 (gyro scale)
- 0x45800000 = 4096.0 (accel scale)
  - 0x43a36666 = 326.8 (temperature scale)
- 0x41c80000 = 25.0 (temperature offset)
```

Simple tests

- Replay earliest HID packets
 - This is when it's useful to know which packets might modify firmware
- Omit or reorder packets, see what happens
- Try modifying values in the packets
- Pay attention to inter-packet timing or repetitions
 - Maybe something is polled until a completion value
 - Some operations take time

Rift S Radio Report

High bit set = new log line

Up to 3 chars of log

TS in μS

If ① & 0x20, 2nd IMU entry is valid?

Button mask, changes when pressing controller buttons more button masks

```

Controller device f52ff39ffaf9afe v1 17 00 log [in IMU ts 2127256968 v2 0 accel 886 -17 481 gyro 30 -16 0 Unknown 22 00 00 00 00 1b ff ff ff ff f7 01 00 00 00 00 00 00 end id 0c 10 1b ff ff ff 08 0a 00 00
Controller device f52ff39ffaf9afe v1 19 02 log fo IMU ts 2127258965 v2 0 accel 893 -18 474 gyro 33 -15 -2 Buttons mask 10 unknown2 22 00 00 00 00 1b ff ff ff 21 02 00 00 00 00 00 end id 00 00 00 00 00 00 00 00 00 00 00 00
Controller device f52ff39ffaf9afe v1 33 02 log fo IMU ts 2127258965 v2 0 accel 893 -18 474 gyro 33 -15 -2 IMU ts 2127260962 v2 0 accel 885 -18 476 gyro 32 -16 -1 end id 0c 10 1d 00 27 00 00 00 00 00 00 00
Controller device f52ff39ffaf9afe v1 2a 00 log j IMU ts 2127262960 v2 0 accel 883 -16 480 gyro 31 -15 -1 IMU ts 2127264957 v2 0 accel 882 -17 483 gyro 32 -14 -1 end id 0c 10 22 00 00 00 00 0e 0e 00 00
Controller device f52ff39ffaf9afe v1 29 02 log l IMU ts 2127266954 v2 0 accel 882 -15 481 gyro 32 -15 -2 Buttons mask 10 unknown2 22 00 00 00 00 1b ff ff ff 0d 00 27 00 00 00 00 00 end id 0c 10 1b ff ff ff 08 0a 00 00
Controller device f52ff39ffaf9afe v1 17 00 log OG IMU ts 2127268951 v2 0 accel 885 -17 479 gyro 33 -15 -1 Buttons mask 10 unknown2 22 00 00 00 00 1b ff ff ff 21 02 00 00 00 00 00 00 end id 0c 00 00 00 00 00 00 00 00 00 00 00
Controller device f52ff39ffaf9afe v1 20 02 log j: IMU ts 2127270947 v2 0 accel 889 -18 488 gyro 32 -14 -1 Buttons mask 10 unknown2 0d 00 27 00 00 00 00 ee ff dc 01 20 00 f0 ff ff ff end id 0c 10 0d 00 27 00 00 00 00 00 00
Controller device f52ff39ffaf9afe v1 17 00 log FW IMU ts 2127272945 v2 0 accel 885 -19 480 gyro 32 -15 -1 IMU ts 2127264957 v2 0 accel 882 -17 483 gyro 32 -14 -1 end id 0c 10 22 00 00 00 00 0e 0e 00 00
Controller device f52ff39ffaf9afe v1 1e 02 log Ver IMU ts 2127274943 v2 0 accel 885 -20 484 gyro 33 -15 -1 Unknown 0d 00 27 00 00 00 00 1b ff ff ff 0d 00 27 00 00 00 00 00 end id 0c 10 1b ff ff ff 08 0a 00 00
Controller device f52ff39ffaf9afe v1 33 00 log sio IMU ts 2127276940 v2 0 accel 887 -19 474 gyro 33 -14 -1 IMU ts 2127278937 v2 0 accel 889 -19 477 gyro 32 -14 -2 end id 0c 10 22 00 00 00 00 0d 00 00
Controller device f52ff39ffaf9afe v1 24 02 log n: IMU ts 2127280934 v2 0 accel 890 -18 483 gyro 32 -16 -1 Buttons mask 10 unknown2 1b ff ff ff 0d 00 27 00 00 00 20 00 f0 ff ff ff end id 0c 10 0d 00 27 00 00 00 00 00 00
Controller device f52ff39ffaf9afe v1 17 00 log 1.1 IMU ts 2127282930 v2 0 accel 886 -15 483 gyro 32 -14 -1 IMU ts 2127264957 v2 0 accel 882 -17 483 gyro 32 -14 -1 end id 0c 10 22 00 00 00 00 0e 0e 00 00
Controller device f52ff39ffaf9afe v1 20 02 log 4.2 IMU ts 2127284928 v2 0 accel 889 -18 482 gyro 32 -15 -1 Buttons mask 10 unknown2 0d 00 27 00 00 00 00 ff ff 0d 00 27 00 00 00 00 00 end id 0c 10 1b ff ff ff 08 0a 00 00
Controller device f52ff39ffaf9afe v1 1e 00 log (d IMU ts 2127286925 v2 0 accel 885 -18 479 gyro 33 -15 0 Unknown 0d 00 27 00 00 00 79 03 ed ff dd 01 20 00 f2 ff fe ff end id 0c 10 22 00 00 00 00 0d 00 00 00
Controller device f52ff39ffaf9afe v1 19 02 log 69e IMU ts 2127288923 v2 0 accel 885 -18 479 gyro 32 -16 -1 Buttons mask 10 unknown2 1b ff ff ff 0d 00 27 00 00 00 20 00 f0 ff ff ff end id 0c 10 0d 00 27 00 00 00 00 00
    
```

Radio ID of sender

seems related to how many IMU/other reports are valid

Zeros

Resting accel $|\bar{a}| \sim 1024 = 1g$

Not sure about gyro units yet

These react to cap sense

Invalid/repeated 2nd IMU block

No idea yet

Questions:

- Jan Schmidt <jan@centricular.com>
- @thaytan