

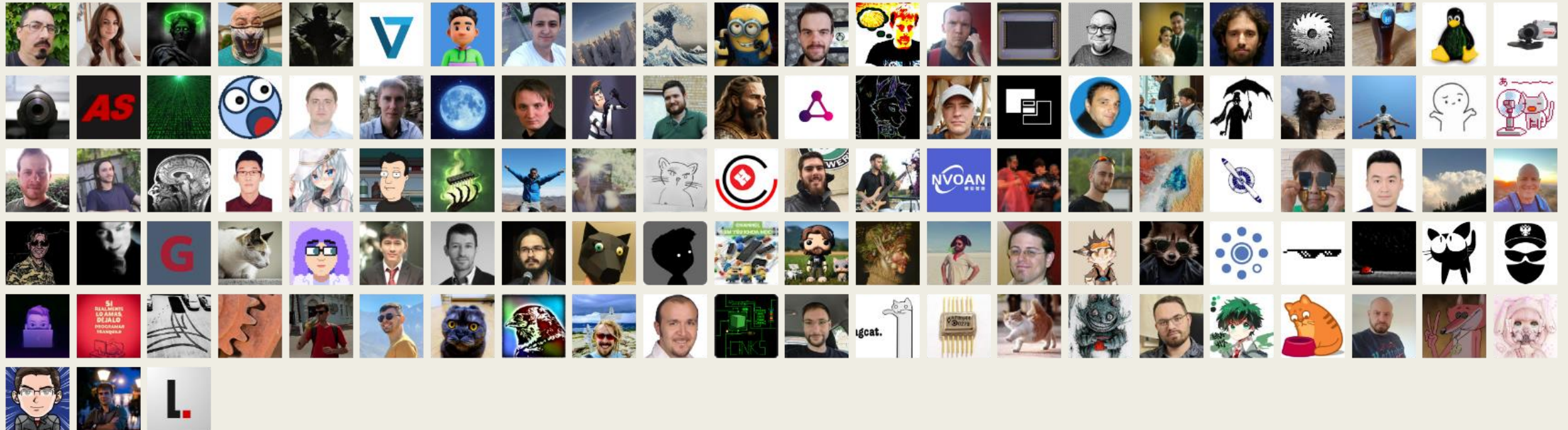


ДИЗАЙН И ЭВОЛЮЦИЯ OPEN SOURCE СИСТЕМ БЕСПРОВОДНОЙ ПЕРЕДАЧИ ВИДЕО

Ильин Дмитрий

By standing upon the shoulders of giants*

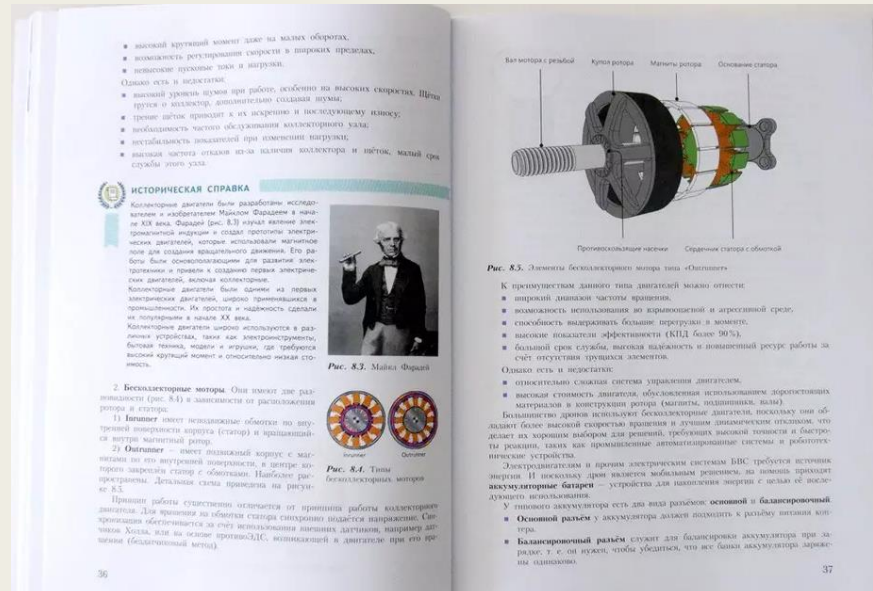
*Isaac Newton



Снова в школу



Наши дни



- высокой крутящий момент даже на малых оборотах,
 - возможность регулирования скорости в широких пределах,
 - высокие крутящие ток и вращение.
- Однако есть и недостатки:
- высокий уровень шума при работе, особенно на высоких скоростях. Шум от коллектора, подшипников, статора и т.д.
 - тряска шфта приводит к их износу и последующему износу,
 - необходимость частого обслуживания коллекторного узла,
 - нестабильность выработки при изменении нагрузки,
 - высокая частота отказов из-за наличия коллектора и шфта, малый срок службы узла шфта.

ИСТОРИЧЕСКАЯ СПРАВКА

Коллекторные двигатели были разработаны исследователем и изобретателем Майклом Фарадеем в начале XIX века. Фарадей (рис. 8.3) создал первую простейшую модель и создал прототипы электрических двигателей, которые использовались частично для создания вычислительного устройства. Его работы были основополагающими для развития электротехники и привели к созданию первых электрических двигателей, включая коллекторные.

Коллекторные двигатели были одним из первых электрических двигателей, широко применявшихся в промышленности. Их простота и надежность сделали их популярными в начале XX века.

Коллекторные двигатели широко используются в различных устройствах, таких как электродвигатели, бытовые техники, машины и инструменты, где требуются высокие крутящие моменты и относительно низкая стоимость.

Бесколлекторные моторы. Они имеют две разновидности (рис. 8.4) и отличаются от расположения ротора в статоре.

1) **Олигопол** имеет несимметричные обмотки по угловой периодичности ротора (биполяр) и радиальной катушки магнитной цепи.

2) **Олигопол** имеет симметричные обмотки с магнитной цепью магнитной цепи, и ротор имеет катушки статора с обмотками. Наиболее рис. 8.5.

Принцип работы существенно отличается от принципа работы коллекторного двигателя. Для вращения на обмотки статора синхронно подается напряжение. Синхронность обеспечивается за счет использования датчиков положения, например датчиков Холла, или на основе принципов ДС, возмущенной и двигателя при его работе (бесколлекторный метод).



Элементы бесколлекторного мотора типа «Олигопол»

К преимуществам данного типа двигателя можно отнести:

- широкий диапазон частоты вращения,
- возможность использования на вращающейся и агрессивной среде,
- способность выдерживать большие перегрузки и моменты,
- высокие показатели эффективности (КПД более 90%),
- большой срок службы, высокая надежность и повышенный ресурс работы за счет отсутствия трущихся элементов.

Однако есть и недостатки:

- относительно сложная система управления двигателями,
- высокая стоимость двигателя, обусловленная использованием дорогостоящих материалов в конструкции ротора (магниты, полупроводники, фольга).

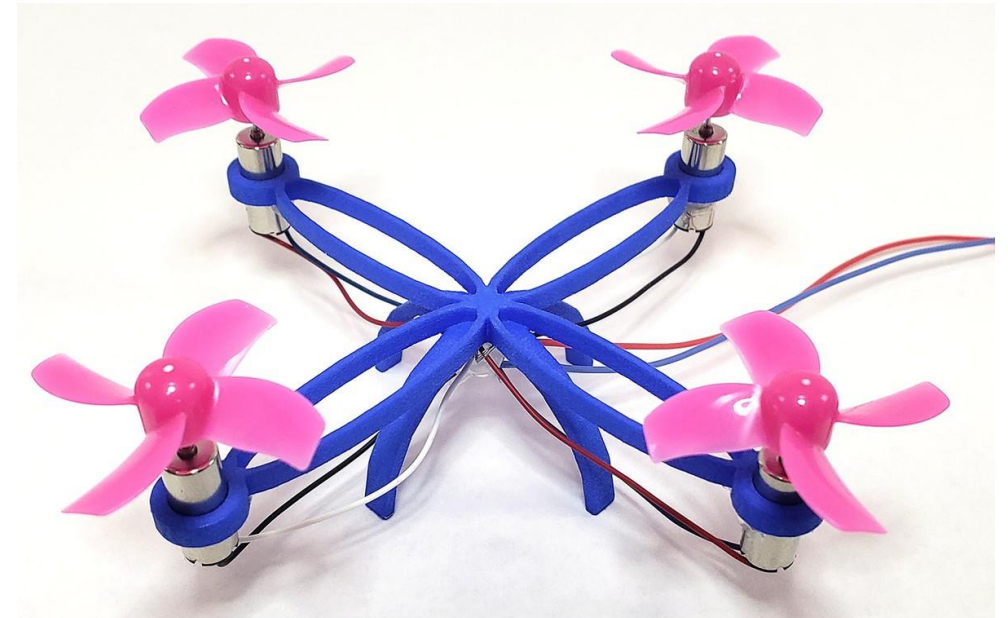
Большинство двигателей используют бесколлекторные двигатели, поскольку они обладают более высокой скоростью вращения и лучшим динамическим откликом, что делает их хорошим выбором для роботизированных устройств, требующих высокой скорости и быстрого реагирования, таких как промышленные автоматизированные системы и робототехнические устройства.

Для управления и другим электрическим системам БИС требуется источник питания. И именно для этого выделены специальные решения, на которых работают **аккумуляторные батареи** — устройства для накопления энергии с целью ее последующего использования.

У любого аккумулятора есть два вида режимов: **основной** и **балансировочный**.

- **Основной режим** — аккумулятор должен работать в режиме питания нагрузки.
- **Балансировочный режим** служит для балансировки аккумулятора при зарядке. Т.е. он нужен, чтобы убедиться, что все ячейки аккумулятора заряжены равномерно.

DIY



Попробуем передать видео по Wi-Fi



Что внутри Wi-Fi адаптера

3.1. 11ac Dual-Band 2x2 RF Application

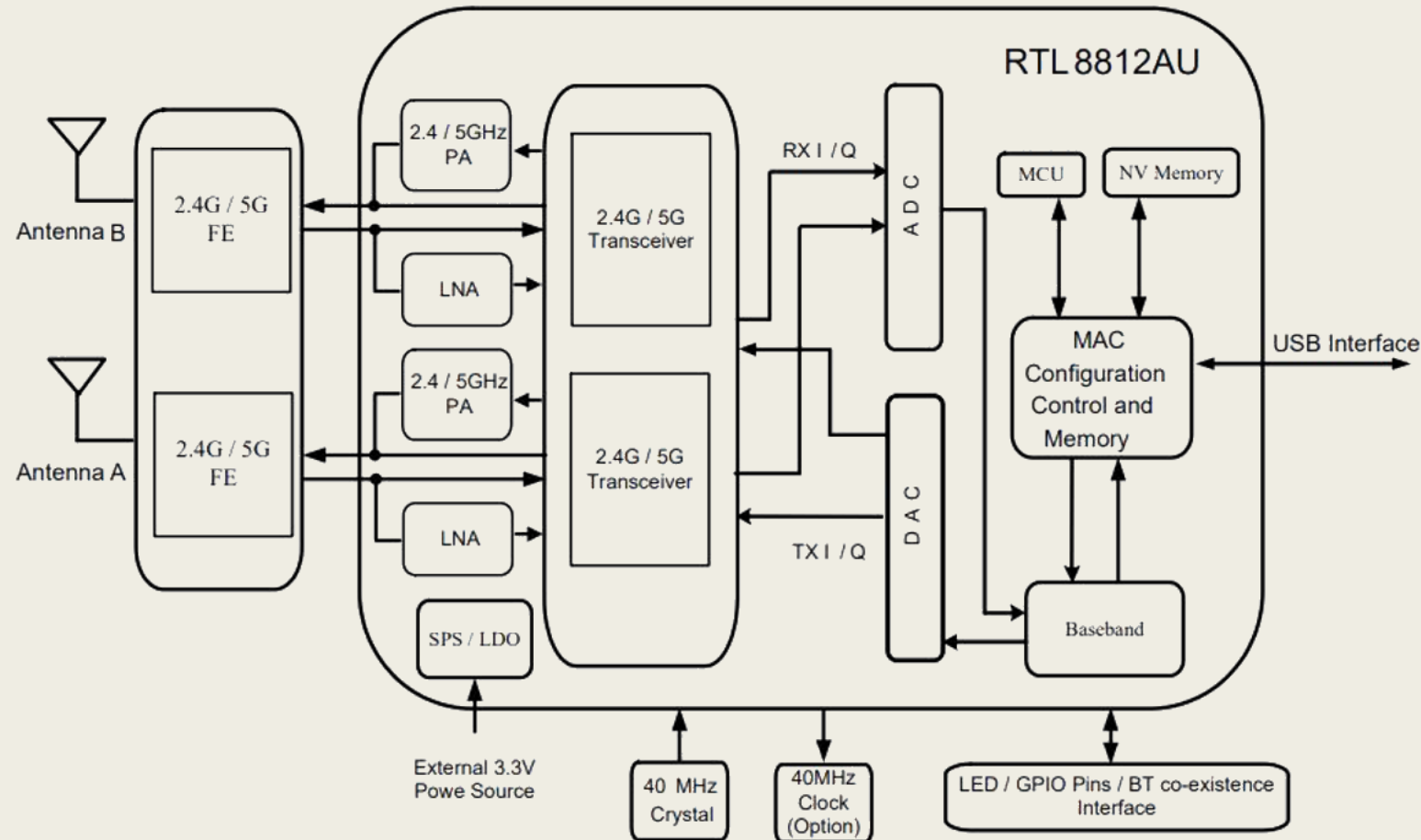


Figure 1. Dual-Band MIMO 2x2 Solution – RTL8812AU-CG (11ac 2x2 MAC/BB/RF + PA)

Простейший беспроводной линк

Передатчик



Injection mode



Приемник



Monitor mode

Множественный прием

Передатчик



Приемник

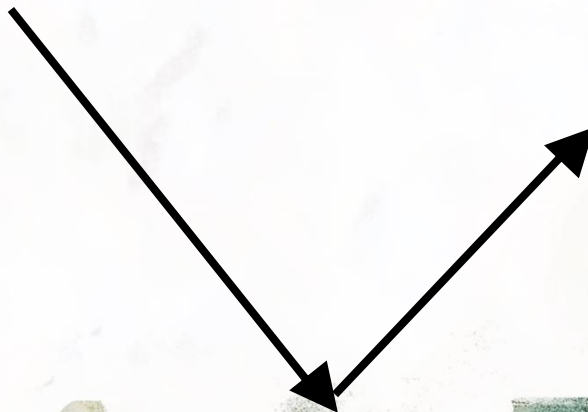


Интерференция

Передатчик

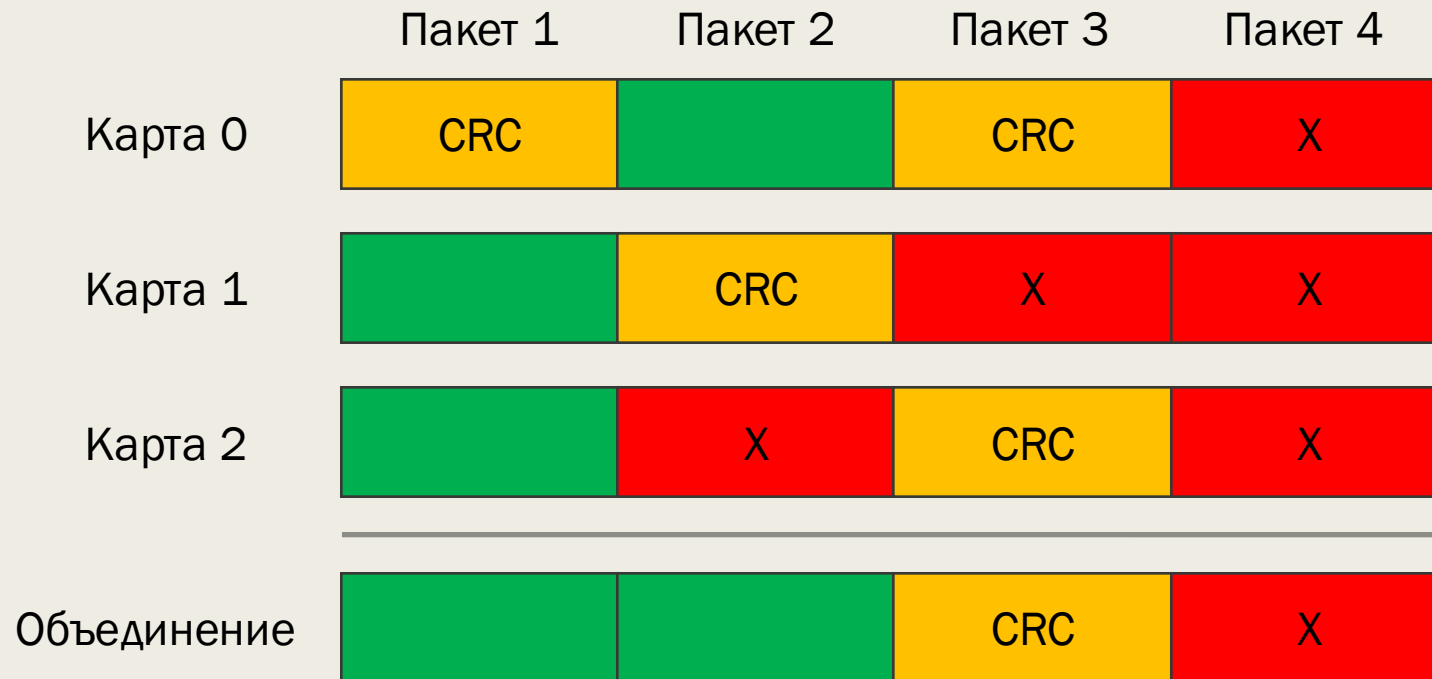


Приемник

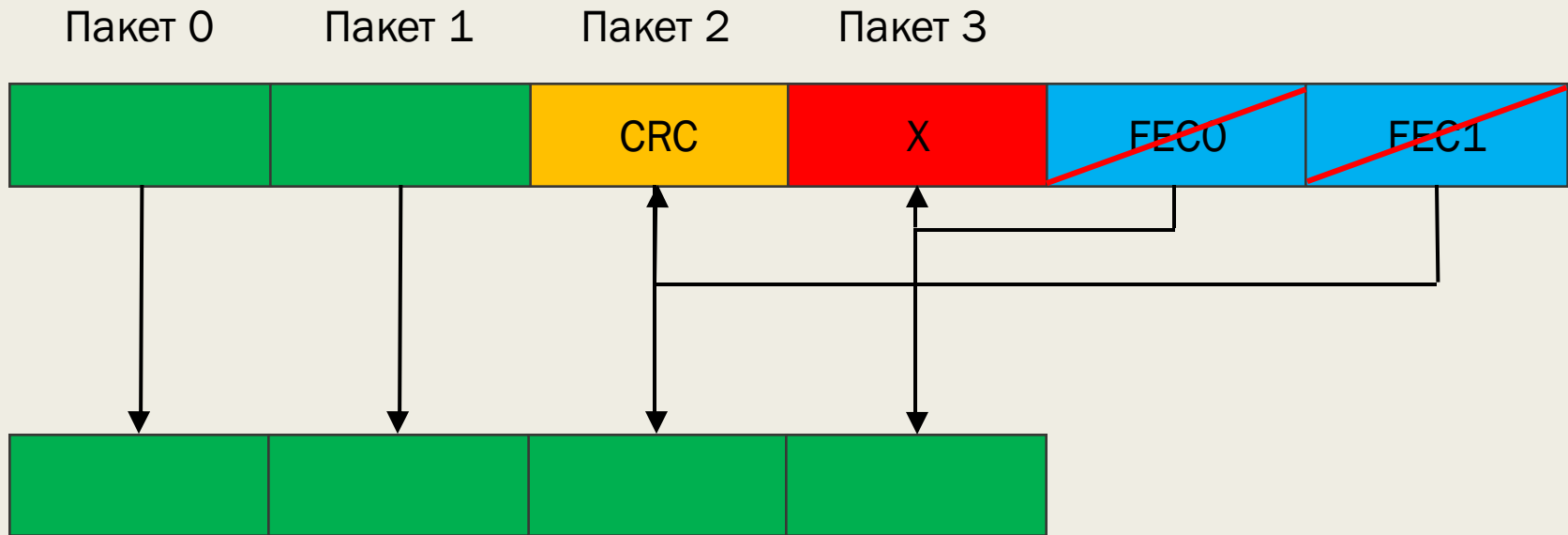




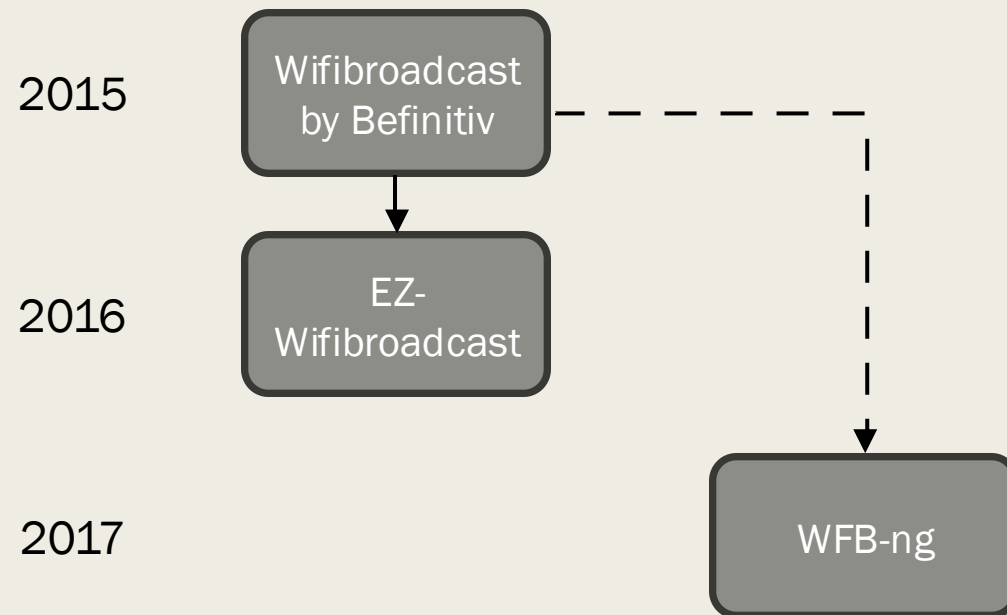
Множественный прием



Предварительная работа над ошибками (FEC)



Эволюция идеи



WFB-ng



Camera



gstreamer



RTP stream
(UDP)



wfb_tx



Display



gstreamer



RTP stream
(UDP)



wfb_rx





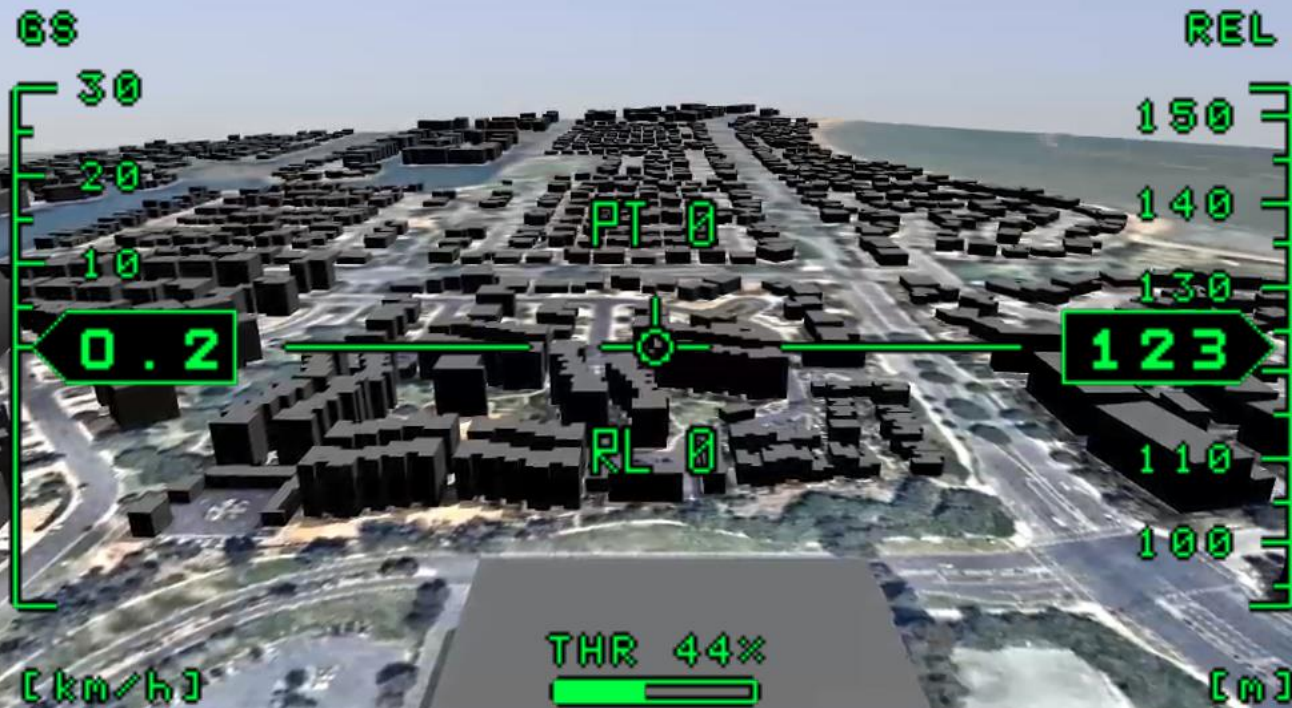
H: 19m

" | N | " 3 042 60 " | E | " H-

01:08:10

ARMED

ALTCTL



THR 44%

60% 30-10
 15.3V 37.523787
 -1.0A -122.255150
 WFB -25 F6 L0

GCS connection regained
 Armed by external command
 Takeoff detected

WFB-ng



[TX: drone video]

```

Flow: 953 kbit/s -> 1.7 mbit/s
sent      pkt/s  pkt
udp       128  60061
fec_t     0  0
drop      0  0
trunc     0  0

[ANT]  pkt/s      Injection [us]
00XX   193      4 < 20 < 89

```

[RX: drone mavlink]

```

Flow: 1.5 kbit/s -> 0.2 kbit/s  FEC: 1/4
recv      pkt/s  pkt
udp       1  35
fec_r     0  3
lost      0  3
d_err     0  26
bad       0  0

Freq  MCS  BW  [ANT]  pkt/s      RSSI [dBm]      SNR [dB]
5805  1  20  0000   4  -22 < -21 < -18  22 < 23 < 25
5805  1  20  0001   4  -30 < -29 < -28  22 < 23 < 25

```

[TX: drone mavlink]

```

Flow: 144 kbit/s -> 295 kbit/s
sent      pkt/s  pkt
udp       13  5546
fec_t     0  0
drop      0  0
trunc     0  0

[ANT]  pkt/s      Injection [us]
00XX   27      7 < 37 < 119

```

[RX: drone tunnel]

```

Flow: 1.1 kbit/s -> 0.0 kbit/s  FEC: 1/2
recv      pkt/s  pkt
udp       1  357
fec_r     0  12
lost      0  31
d_err     0  0
bad       0  0

Freq  MCS  BW  [ANT]  pkt/s      RSSI [dBm]      SNR [dB]
5805  1  20  0000   3  -22 < -19 < -18  19 < 20 < 22
5805  1  20  0001   3  -30 < -29 < -28  19 < 20 < 22

```

[TX: drone tunnel]

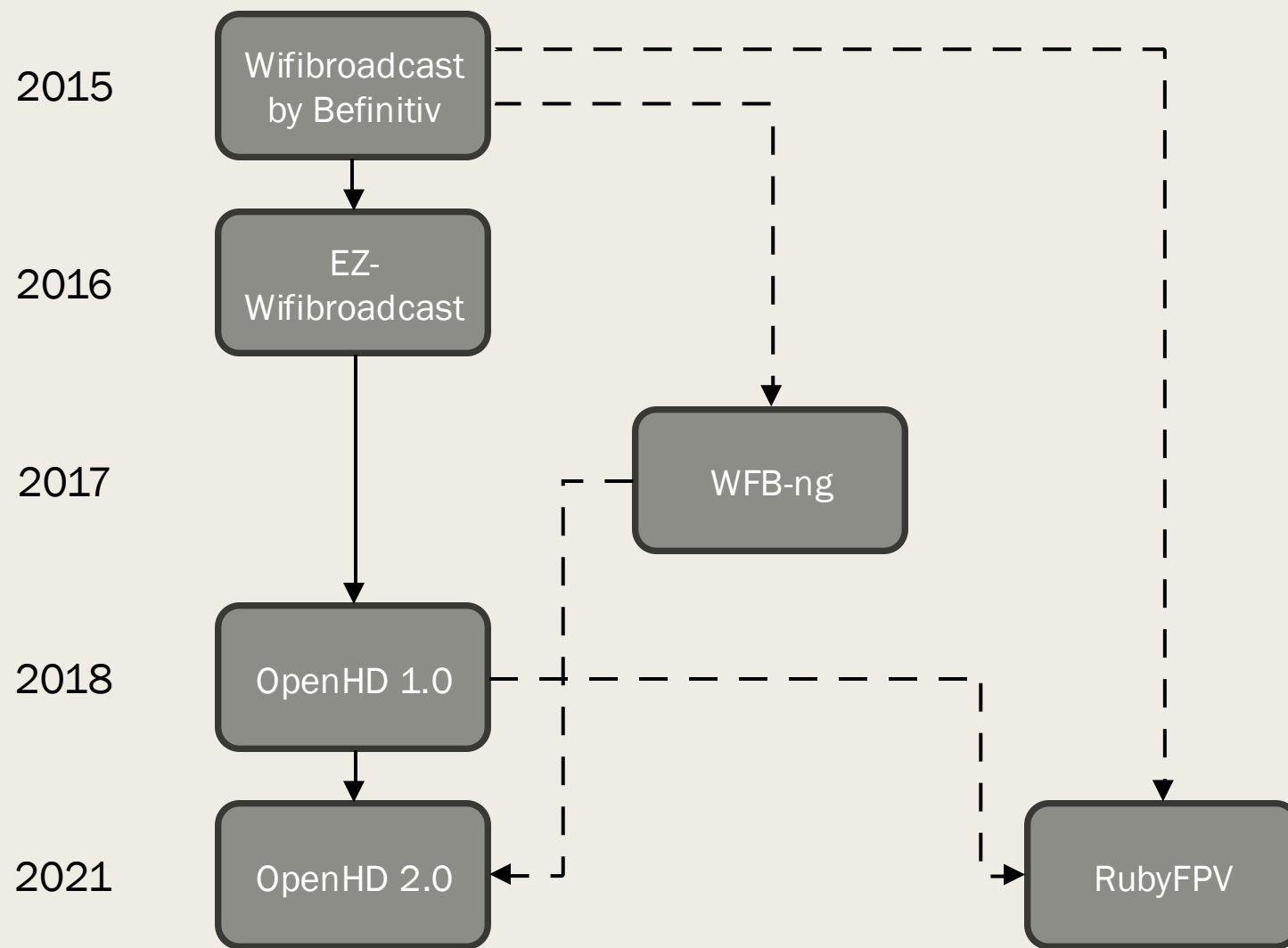
```

Flow: 0.0 kbit/s -> 0.4 kbit/s
sent      pkt/s  pkt
udp       1  523
fec_t     0  0
drop      0  0
trunc     0  0

[ANT]  pkt/s      Injection [us]
00XX   2      16 < 60 < 105

```

Эволюция идеи





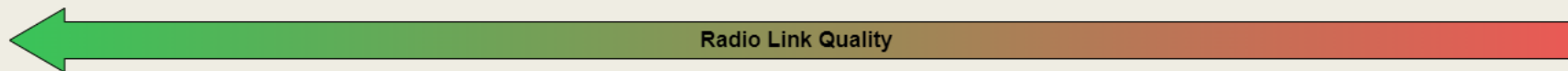
Ruby

Radio Links and Video Links management in Ruby

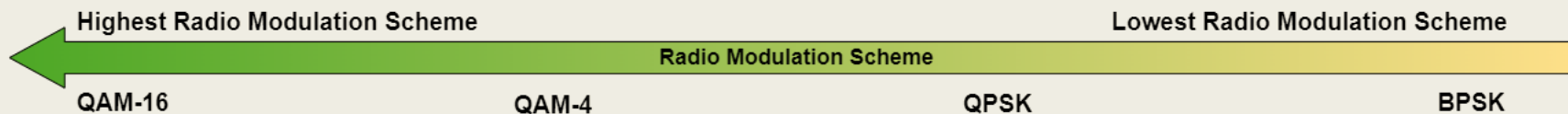
Ruby continuously tracks the radio link quality by measuring dbm signal, radio ping quality, retransmissions, lost packets, interferences in order to compute a global more accurate Radio Link Quality indicator

Best Radio Link Conditions

Worst Radio Link Conditions



Based on radio link conditions, the following parameters are continually adjusted with the single purpose to maintain a consistent video link (albeit with a lower quality or increased latency) even in the worst radio conditions:



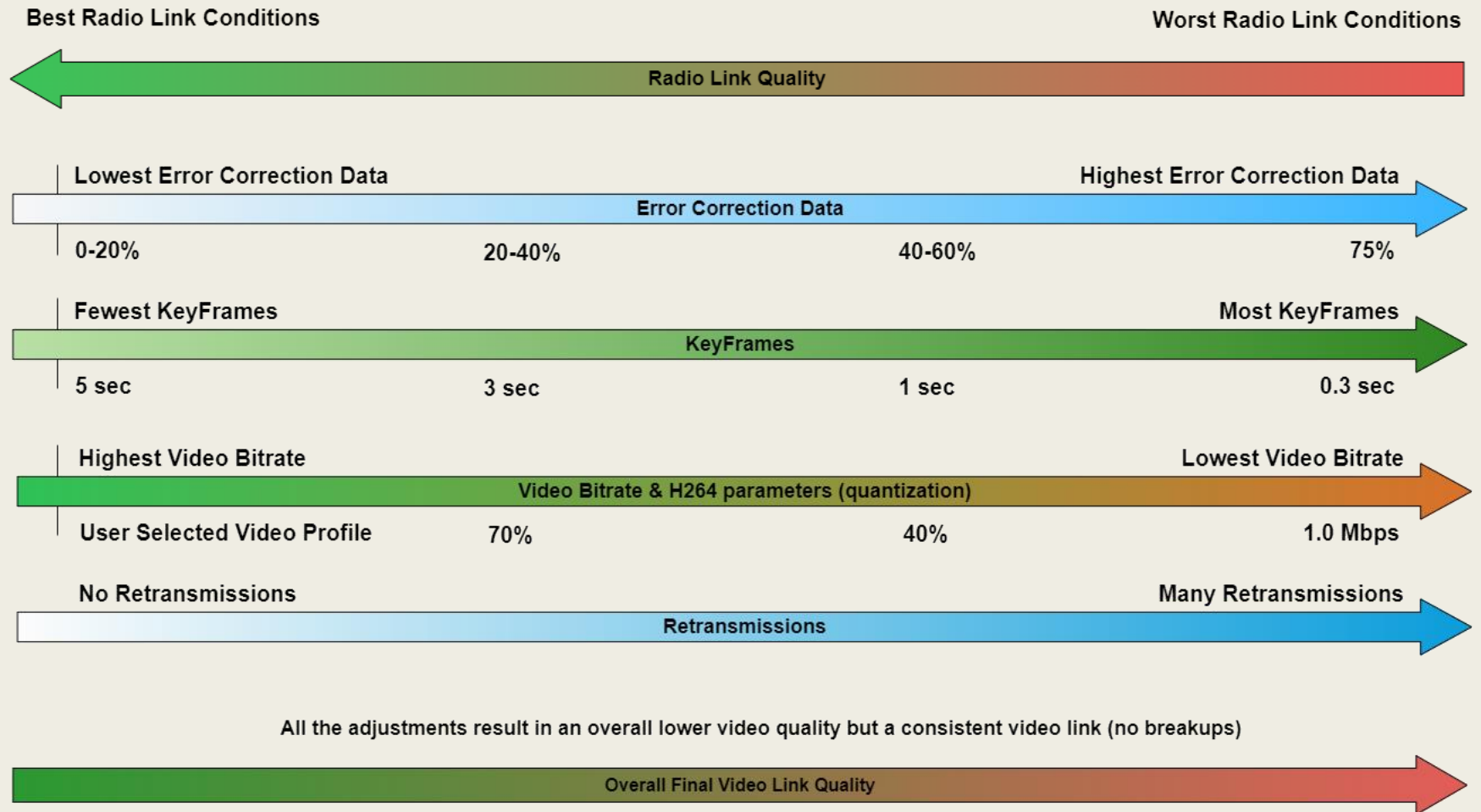
802.11ac - VHT

MCS, SNR and RSSI

VHT MCS	Modulation	Coding	20MHz				40MHz				80MHz				160MHz			
			Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI
			800ns	400ns			800ns	400ns			800ns	400ns			800ns	400ns		
1 Spatial Stream																		
0	BPSK	1/2	6.5	7.2	2	-82	13.5	15	5	-79	29.3	32.5	8	-76	58.5	65	11	-73
1	QPSK	1/2	13	14.4	5	-79	27	30	8	-76	58.5	65	11	-73	117	130	14	-70
2	QPSK	3/4	19.5	21.7	9	-77	40.5	45	12	-74	87.8	97.5	15	-71	175.5	195	18	-68
3	16-QAM	1/2	26	28.9	11	-74	54	60	14	-71	117	130	17	-68	234	260	20	-65
4	16-QAM	3/4	39	43.3	15	-70	81	90	18	-67	175.5	195	21	-64	351	390	24	-61
5	64-QAM	2/3	52	57.8	18	-66	108	120	21	-63	234	260	24	-60	468	520	27	-57
6	64-QAM	3/4	58.5	65	20	-65	121.5	135	23	-62	263.3	292.5	26	-59	526.5	585	29	-56
7	64-QAM	5/6	65	72.2	25	-64	135	150	28	-61	292.5	325	31	-58	585	650	34	-55
8	256-QAM	3/4	78	86.7	29	-59	162	180	32	-56	351	390	35	-53	702	780	38	-50
9	256-QAM	5/6			31	-57	180	200	34	-54	390	433.3	37	-51	780	866.7	40	-48



Ruby



Highest Video Quality



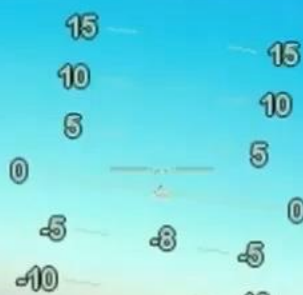
Lowest Video Quality



30 40 50 60 70 80 90 100 110 120 130
NE E
080

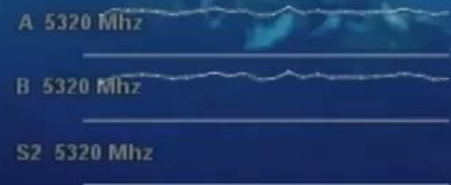
(km/h) SPD(A) 61.7

GS: 95.0 km/h
D: 21614 m
080



1980
1967 (m) ALT

Radio Link 1960 99%
Link delay: 10 ms
Data: 8 kbps
Video: 12204 kbps
A: RX -54 dbm 99% 12.2 Mbps
B: RX/TX -56 dbm 1.8% 11.9 Mbps
S2: RX -40 dbm 0% 0 kbps

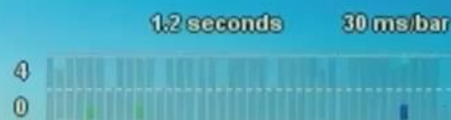


Efficiency Stats

mA/km: 164
mA/h: 11023

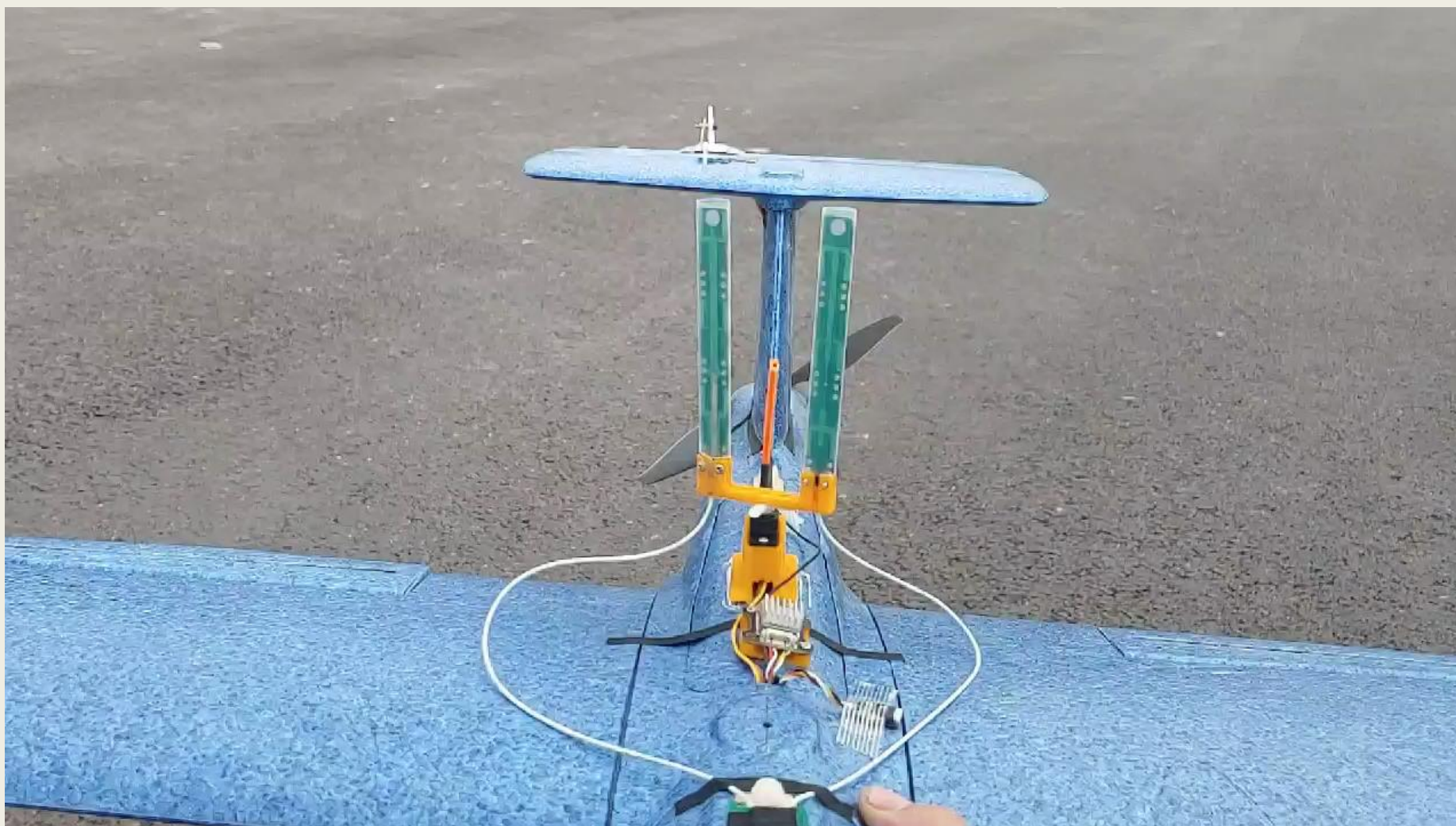
Video Decoder 12.2 Mbps

Stream: 1920x1080, 30 fps 10 key
Params: HQ / R1/[250-26]
EC: 5/1/1300 / 10/2 ms



Max lost gap packets: 3
Max lost gap time: 6 ms
Packets/sec out: 1090
Blocks/sec out: 181
Received retrans: 11540
Bad blocks (discarded): 153
Unrecoverable packets: 381
S2: RX, -40 dbm
B: RX/TX, -56 dbm
A: RX, -54 dbm

Направленные антенны



RX1: 1 %
-88dBm

14478 mAh

-88dBm [155751]
-90dBm [81712]

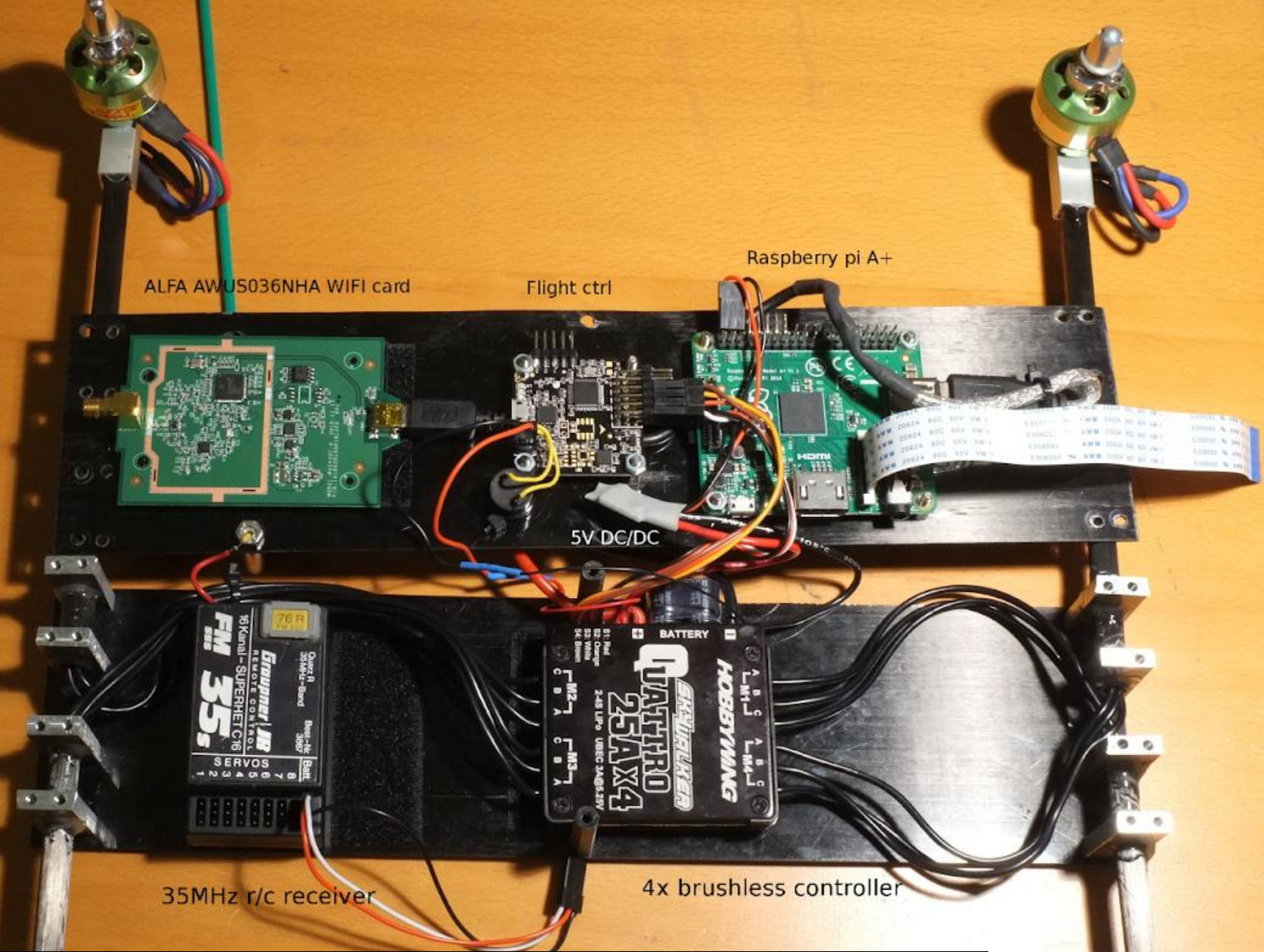
46°
W N E S ↑
BATT 0%



20 17.01 V
0 % 2.42 A

AUTO

121.1km



ALFA AWUS036NHA WIFI card

Flight ctrl

Raspberry pi A+

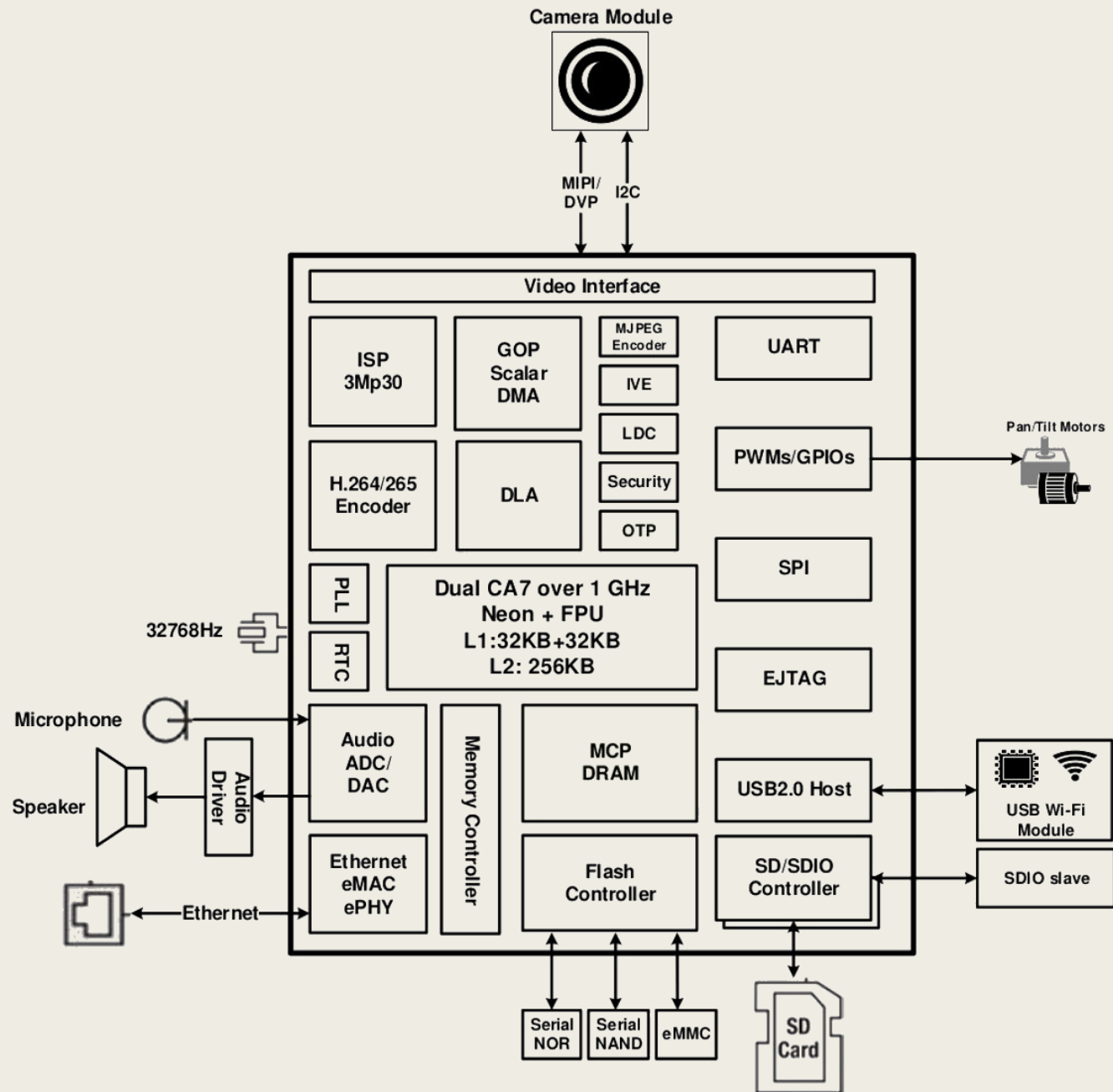
5V DC/DC

35MHz r/c receiver

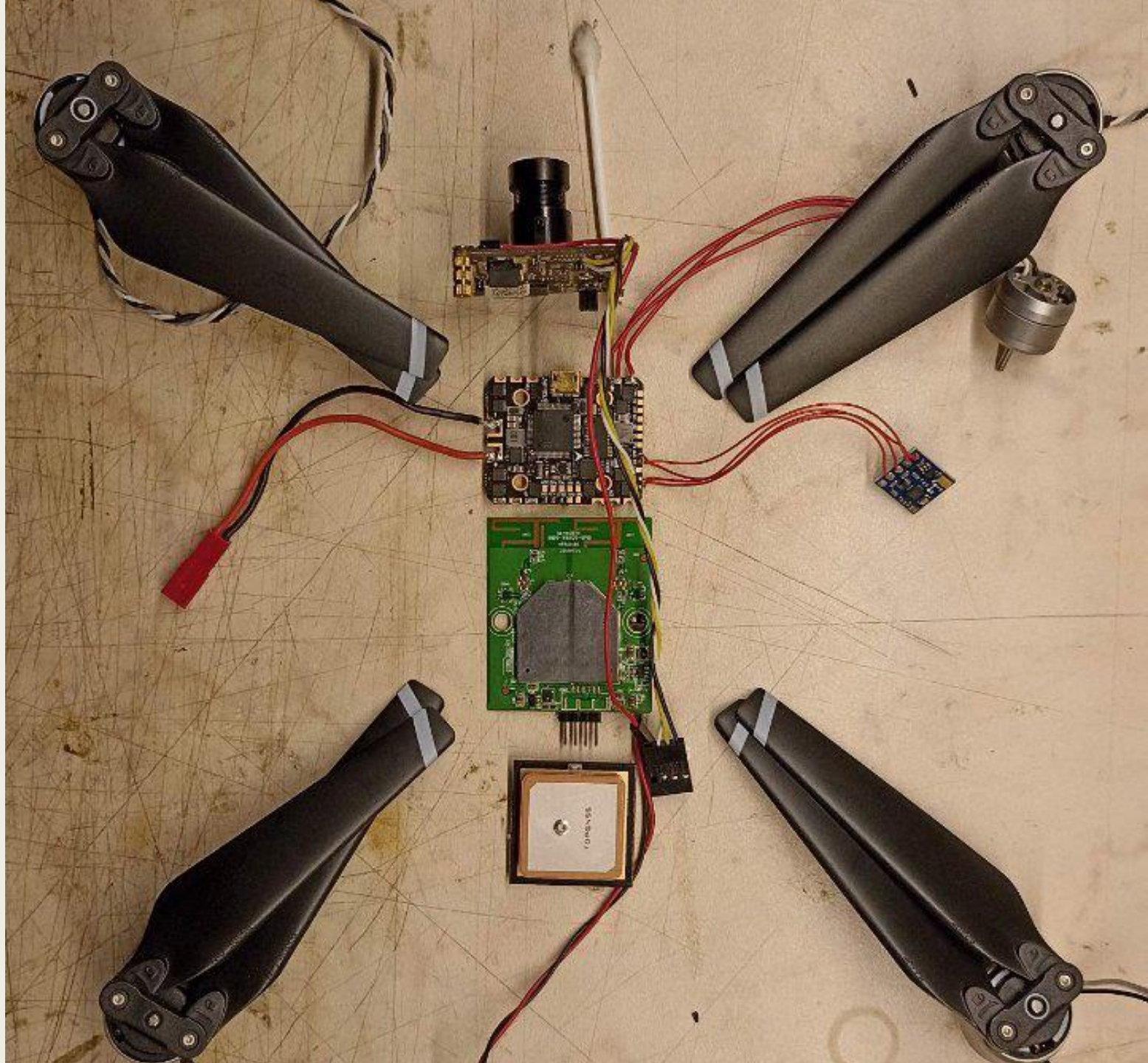
4x brushless controller

Дедушка всех Open Source систем

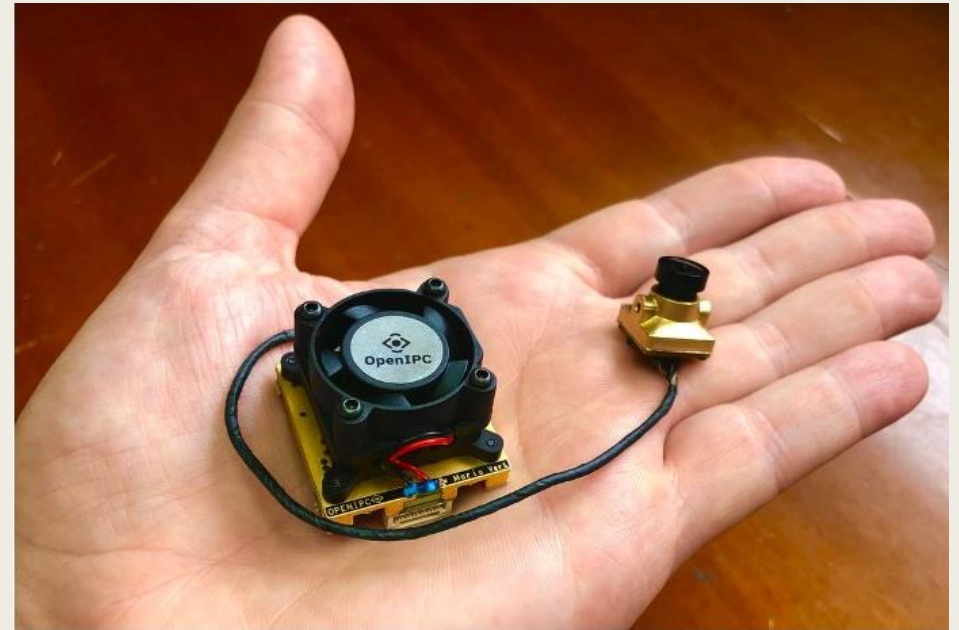
Что такое CCTV IP камера?



Современный вариант



OpenIPC AIO (all-in-one)



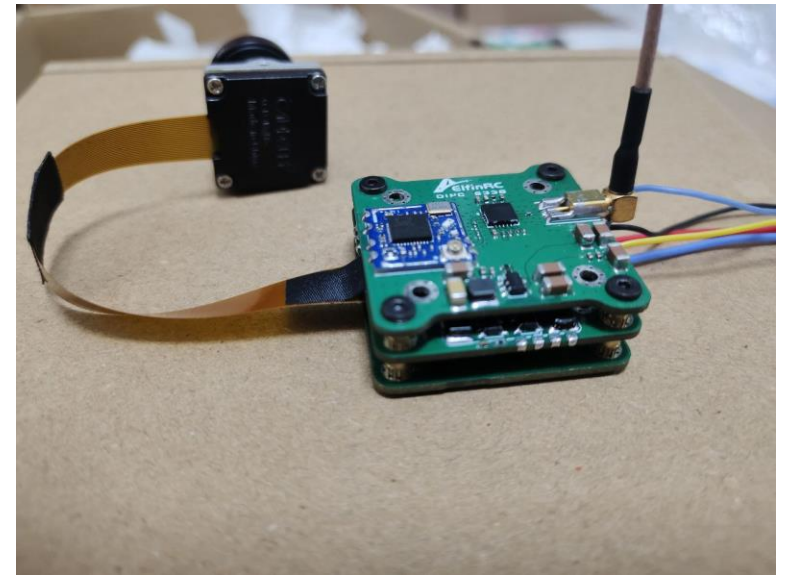
OpenWRTIPC



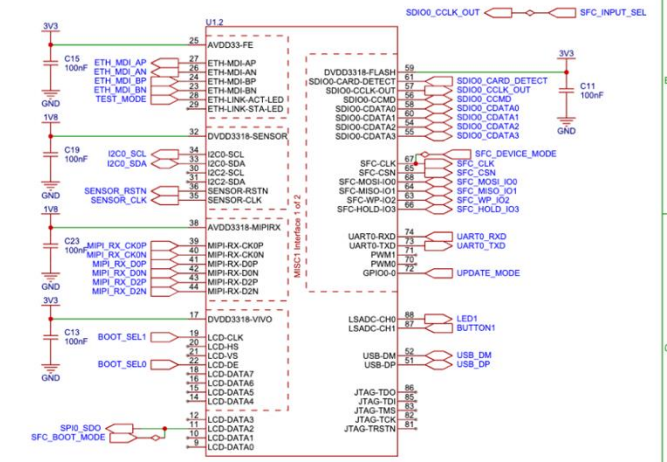
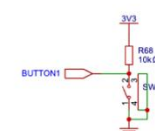
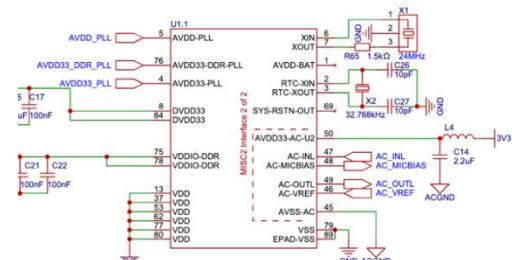
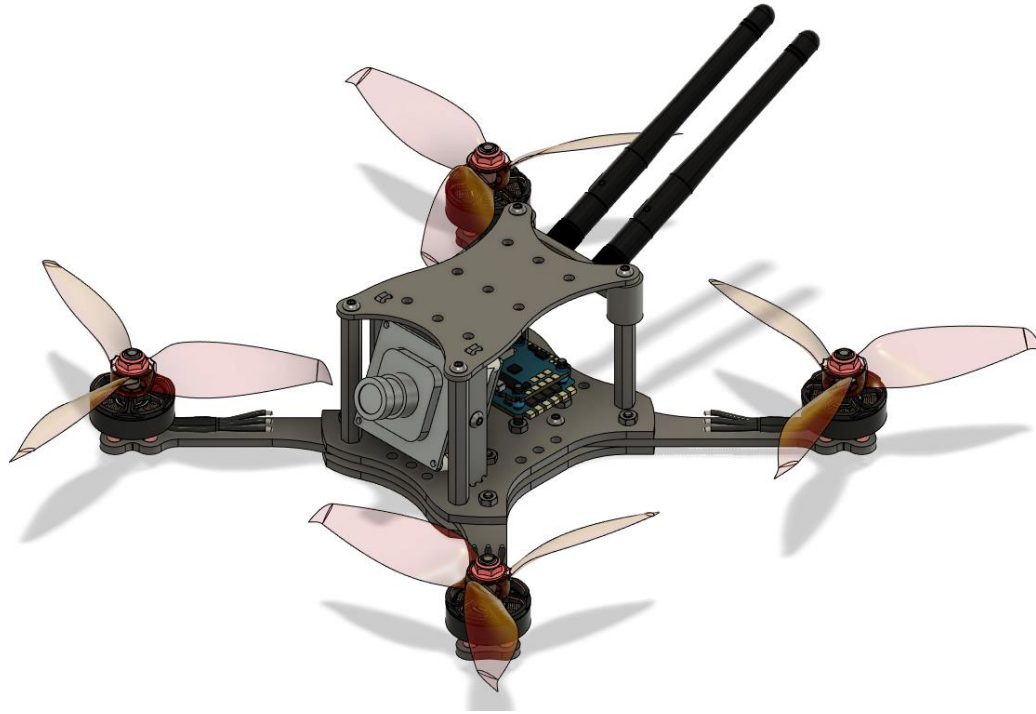
RunCam[®]
Born for RC Fanatics



EMAX



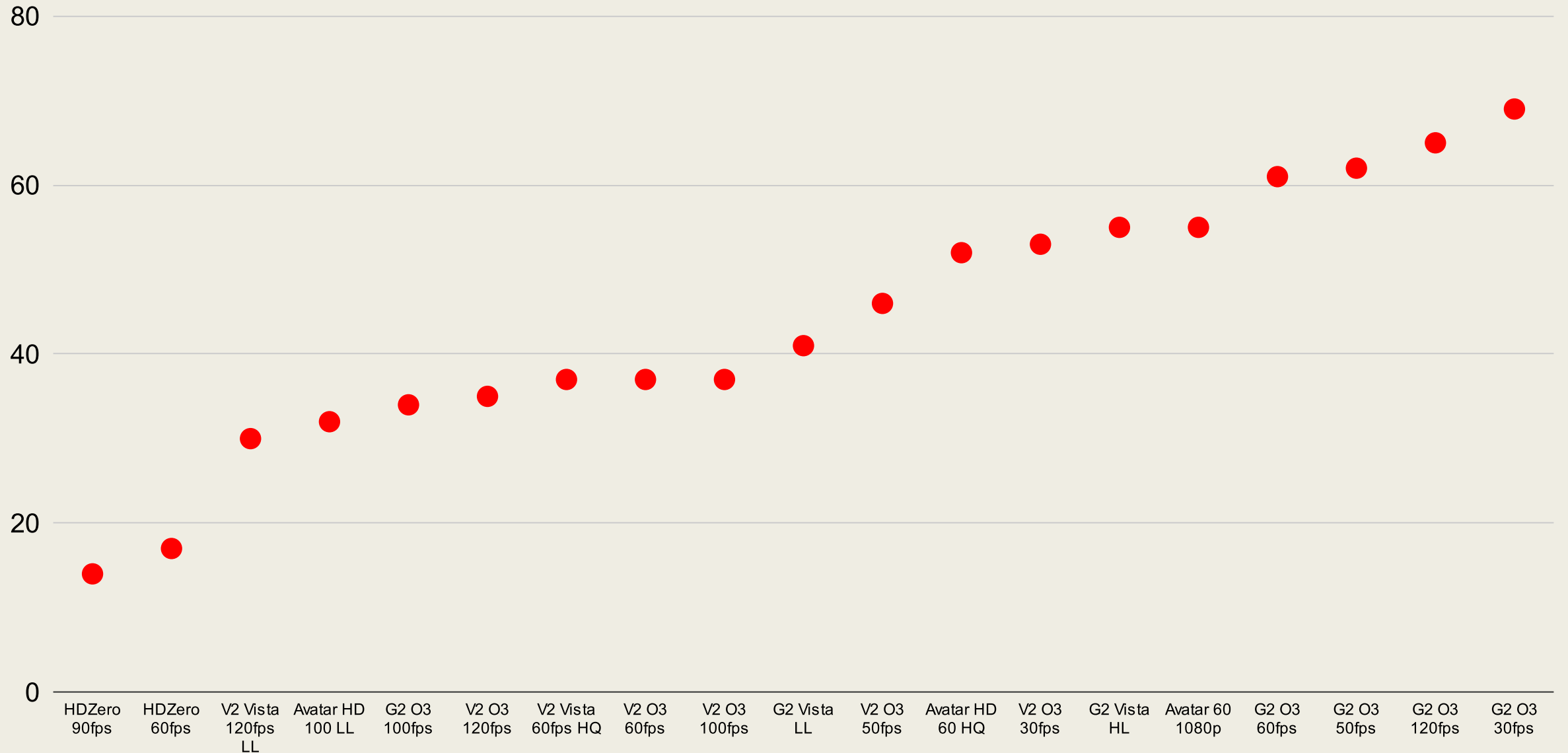
Open hardware



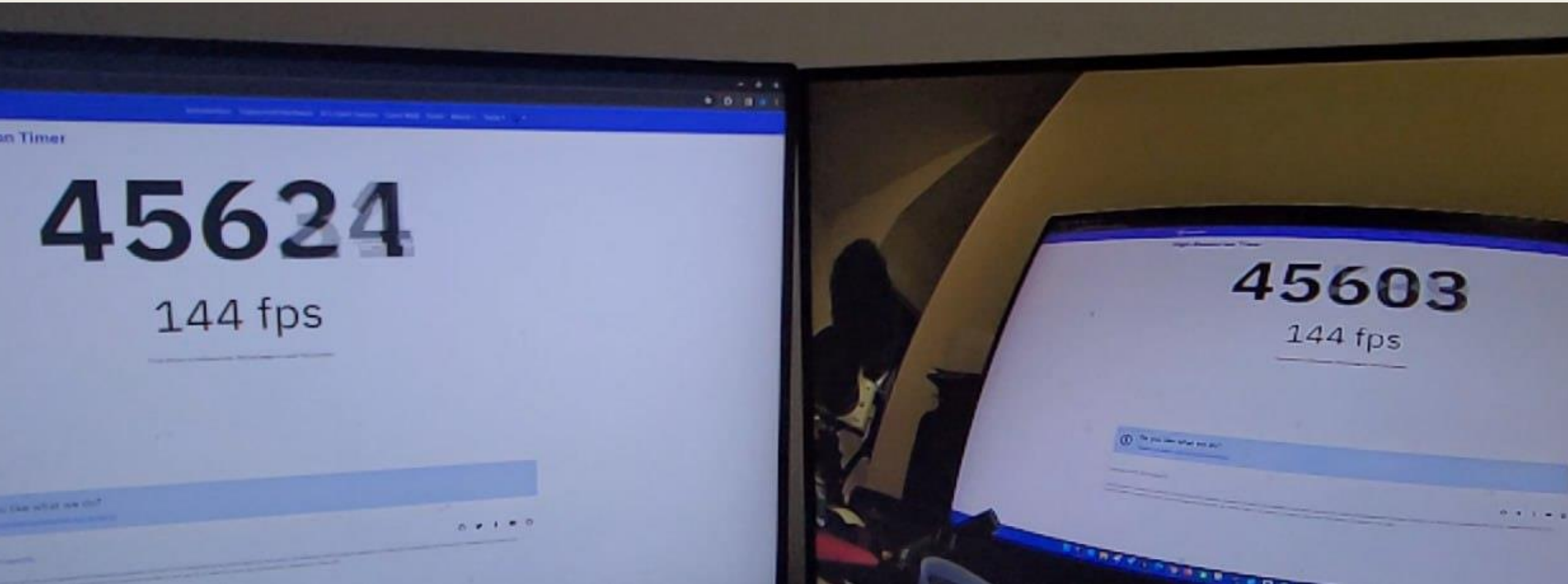
Schematic	Schematic1	Update Date	2023-10-19
Page	main	Create Date	2023-10-19
Drawn	kunlin	Part Number	JLPCPB-001
Reviewed	kunlin	openipc-hi3516	

Digital FPV System Latency - Mads Tech*

* <https://fpwwiki.co.uk/digital-fpv-latency-testing>



G2G latency tests



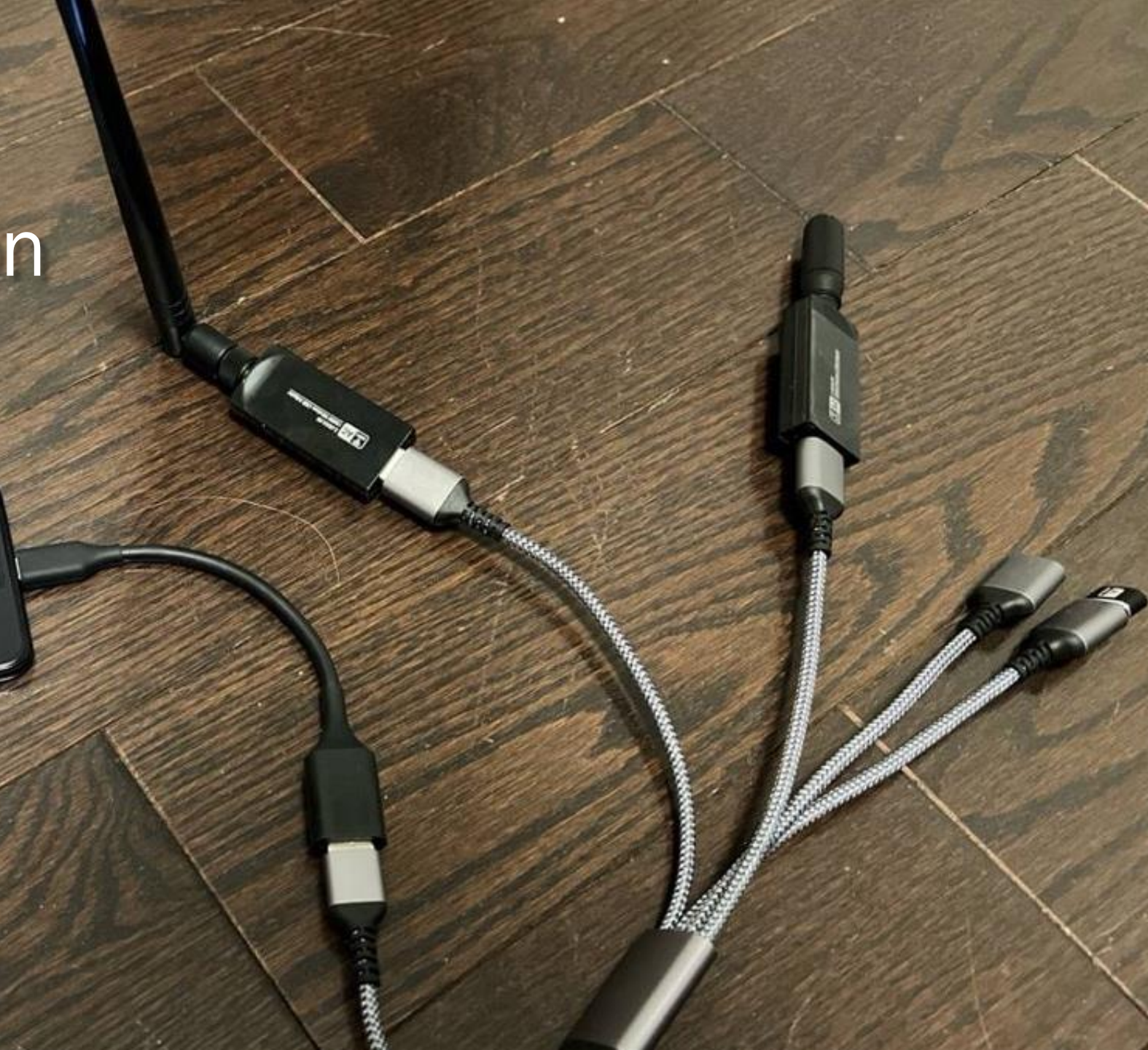
Тюнинг кодека

- H.265 со слайсами
 - *Мы хотим отправлять в сеть данные пока идет readout сенсора*
- 60/90/120 fps (240 fps для гонок)
 - *Обязательный тюнинг драйверов сенсоров и ISP*
- FEC
 - *Например, восстанавливает 4 пакета из блока 12*
- Битрейт кодека зависит от кейса
 - *Минимум для гонок*

Видеоочки



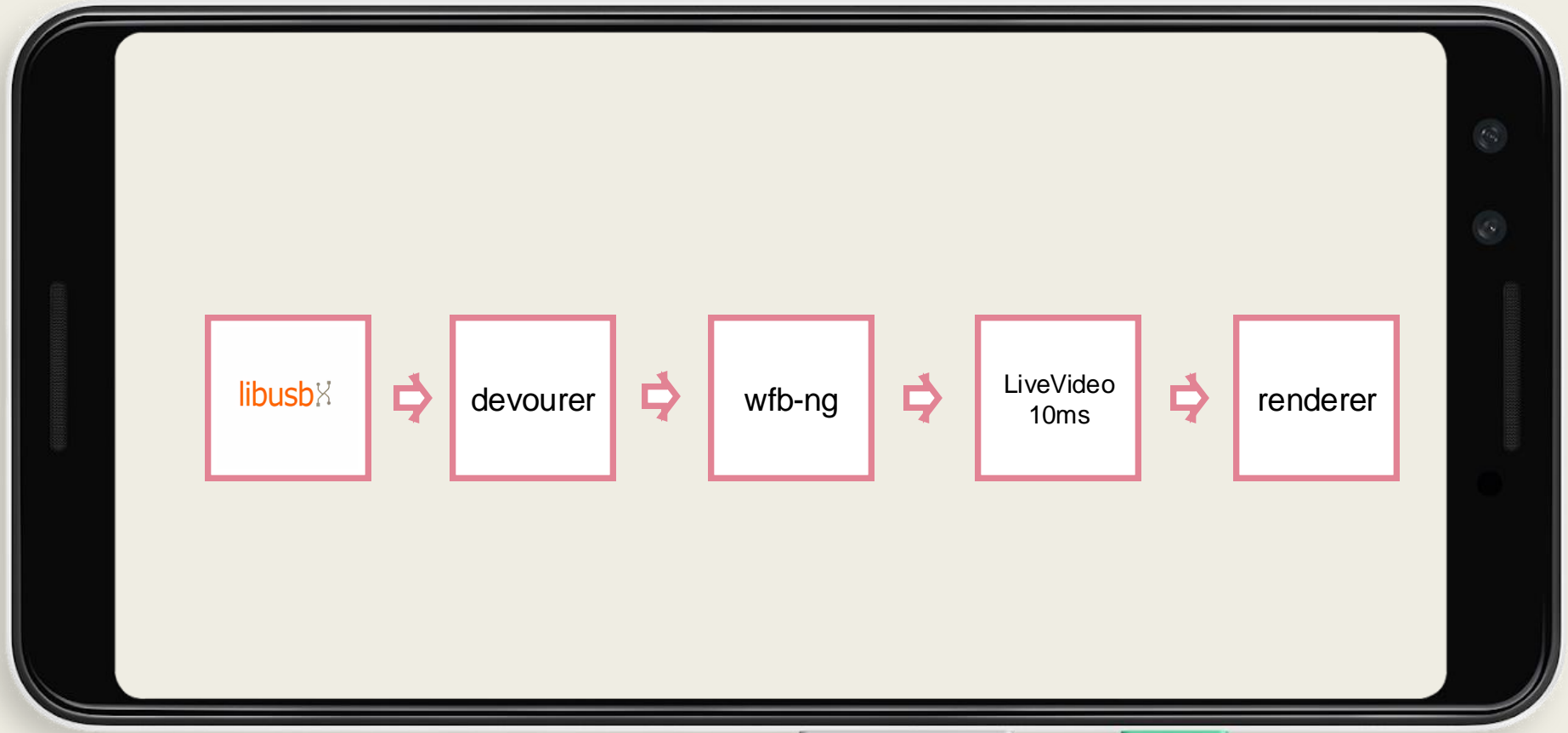
Android как GroundStation



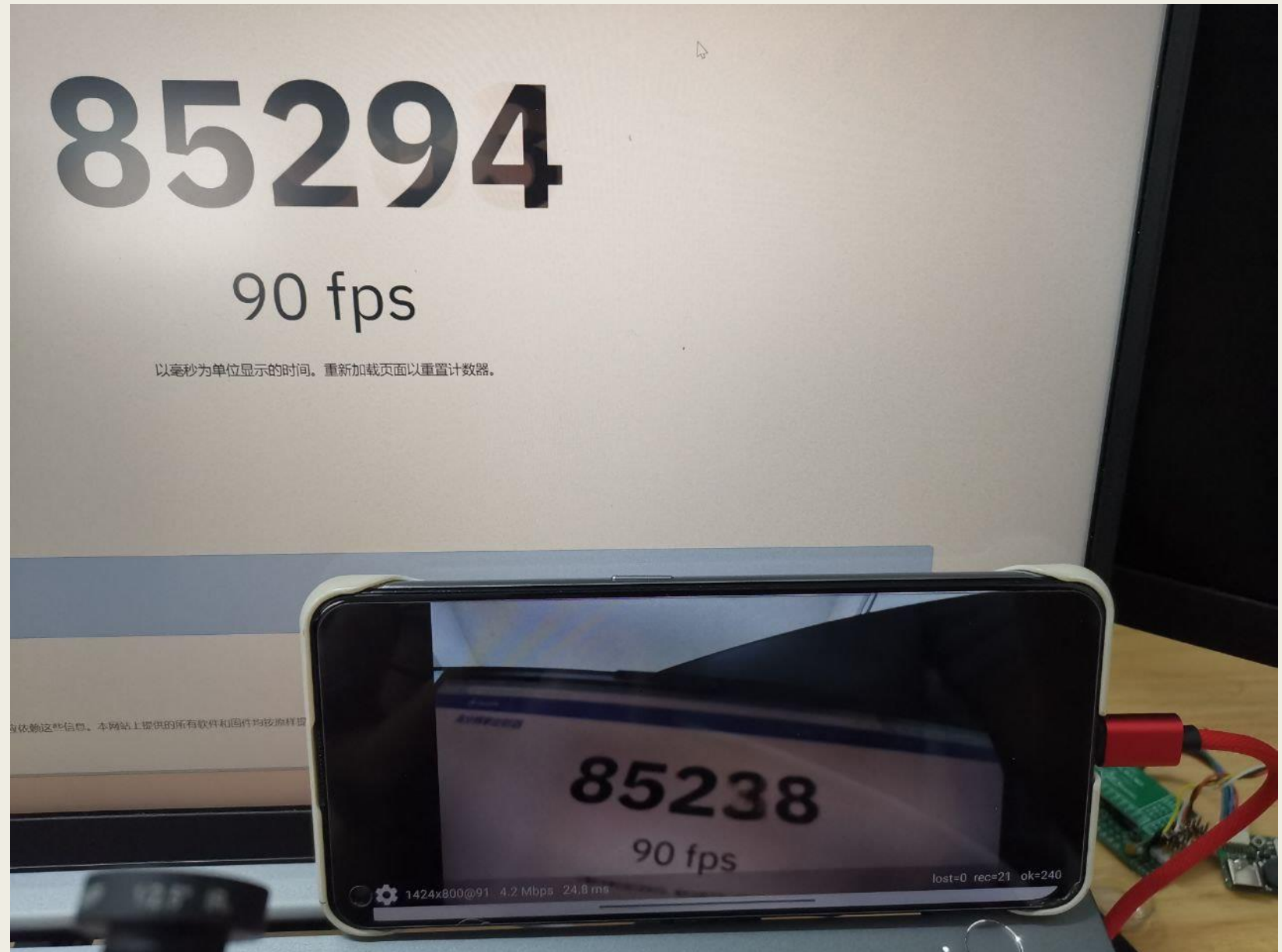
USB + Android

```
activity_main.xml x MainActivity.java x AndroidManifest.xml x device_filter.xml x UsbManager.java x
29
30 @Override
31 protected void onCreate(Bundle savedInstanceState) {
32     super.onCreate(savedInstanceState);
33
34     UsbManager manager = (UsbManager) getSystemService(Context.USB_SERVICE);
35
36     for (UsbDevice usbDevice: manager.getDeviceList().values()) {
37         String deviceName = usbDevice.getDeviceName();
38
39         Log.e(TAG, msg: "DeviceName " + usbDevice.getDeviceName());
40         Log.e(TAG, msg: "ManufacturerName " + usbDevice.getManufacturerName());
41         Log.e(TAG, msg: "ProductName " + usbDevice.getProductName());
42
43         UsbInterface usbInterface = findInterface(usbDevice);
44
45         UsbEndpoint mOutEndpoint = null, mInEndpoint = null ;
46
47         for (int nEp = 0; nEp < usbInterface.getEndpointCount(); nEp++) {
48             UsbEndpoint tmpEndpoint = usbInterface.getEndpoint(nEp);
49
50             Log.e(TAG, msg: "endpoint = " + tmpEndpoint);
51
52             if (tmpEndpoint.getType() != UsbConstants.USB_ENDPOINT_XFER_BULK) continue;
53
54             if ((mOutEndpoint == null)
```

FPVue



G2G latency с userspace драйвером



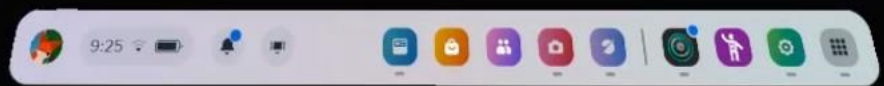
VR видео



Video

8K60 decode, 8K30 encode, low-latency slice-based decoding

AVC, HEVC, VP9, and AV1 decode



Open Source community

- github.com/orgs/OpenIPC/repositories
- github.com/svpcom/wfb-ng
- github.com/PetruSoroaga/RubyFPV
- github.com/gehee

С чего начать?

- Aliexpress.com
- Ozon.ru
-

Что дальше?

- Стереоскопическое видео
- Замена Wi-Fi на 5G
- Тюнинг современных кодеков AV1 и H.266
- Железо: адаптивный подход к потреблению мощности
- MIMO everywhere
- Софт: использование дополнительной hw акселерации

Q&A
session

Если ли жизнь после Wi-Fi?

