

In some cases 2 of these are logged, even for only one device being plugged in.

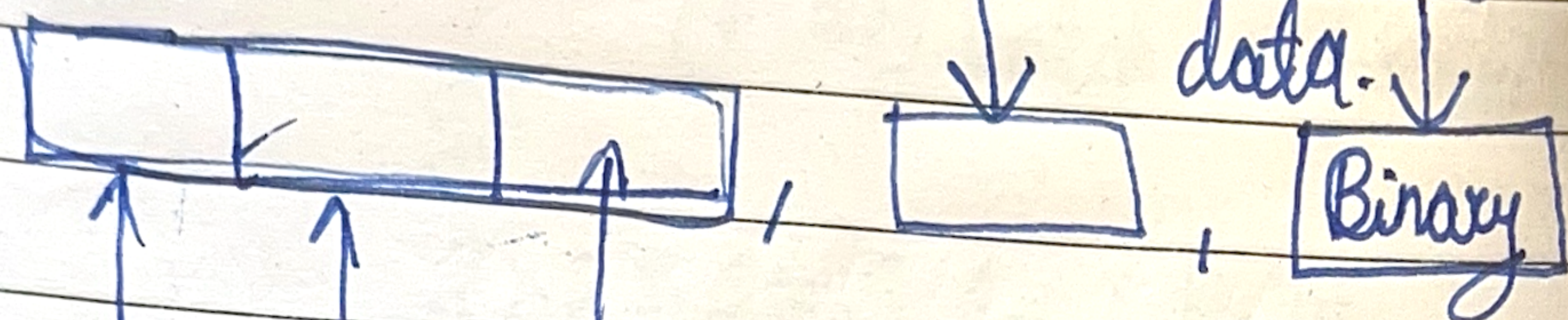
Explanation of device-block logs:

- $X, 056, X$ → Device boot + port #
 ~~$X, 056, X$ → In some cases, ~~two of these~~ logged 4.1~~
- $X, 038, X$ → Device software version MSB
 $X, 039, X$ → Device software version.
- $X+1, 037, X$ → "Device connected" + Model ID
+ Port number
- $X+1, 001, X$ → "Device connected" + Family ID,
+ Port number
- $X+1, 033, X$ → Device serial number Year
(Use 256 weighting) for all below
- $X+1, 034, X$ → Device serial number (byte 0)
- $X+1, 035, X$ → Device serial number (byte 1)
- $X+1, 036, X$ → Device serial number (byte 2)

Arayan Gorai

Event code

Event code
data. ↓



Number from 0-~~255~~ 256

Number X ~~255~~ 256

Number X ~~255~~ ~~255~~ 256 X 256

→ Total these, and then times by 0.1, to get seconds.

Note: Port number = which port the machine is drill is connected to

0

4 2 → 8 total ports possible.

Y values can be assigned based on device number.

4 2 1

$$4+2+1=7$$

7+1=8 total values with 3 bits

8

$$2 \times 2 \times 2 = 8 \text{ total values.}$$

4 2 1
0 0 0

21

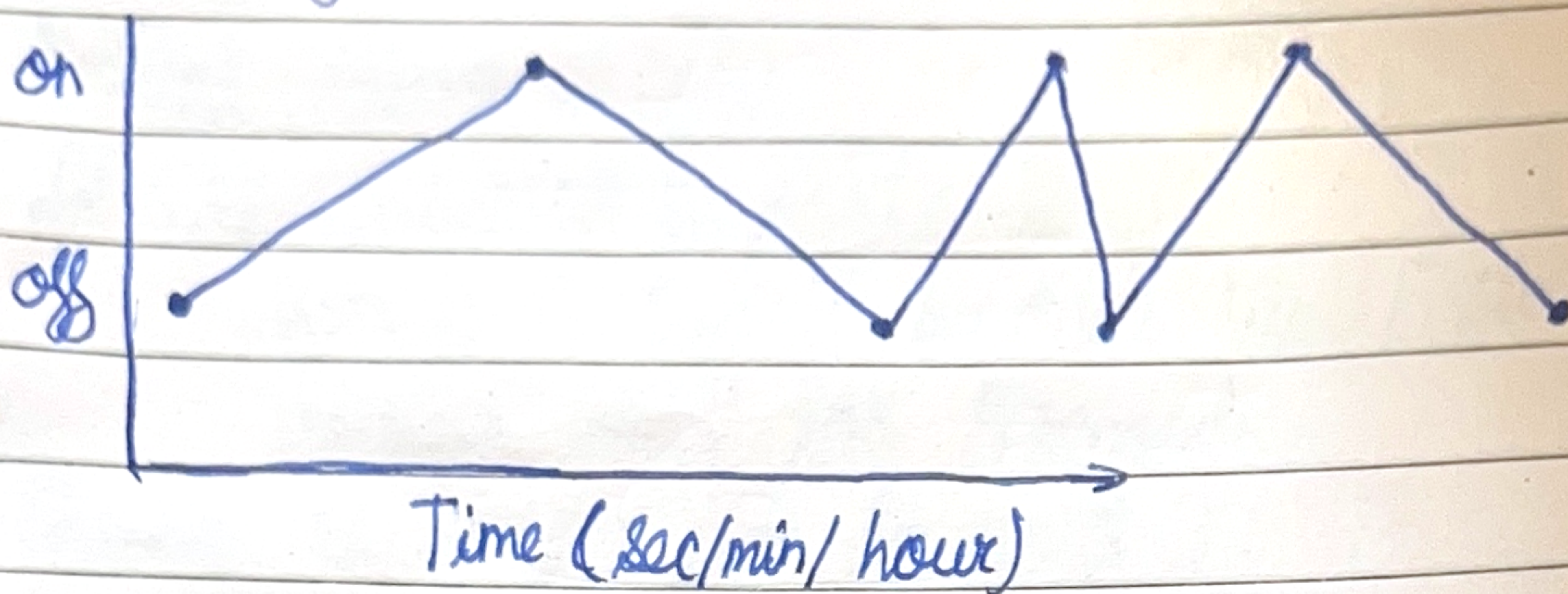
DEVICE

~~ON~~

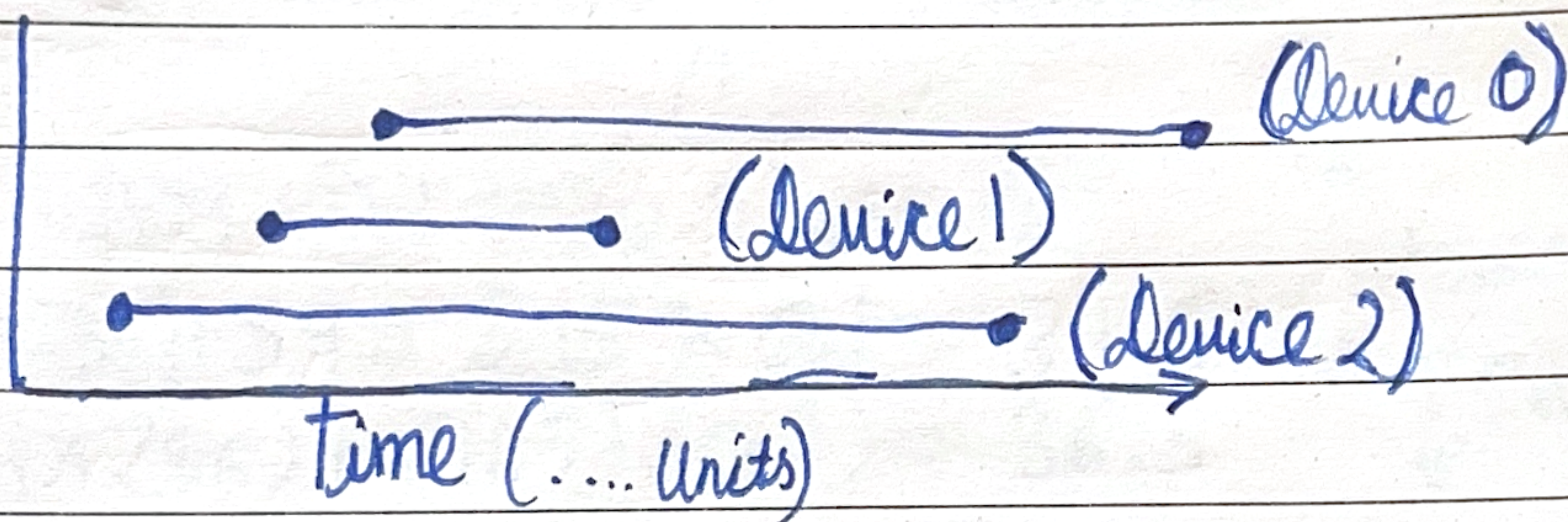
[S | E] , [S | E]

device-on-list [

Possible graphs:



Possible graphs:



~~Wrong~~
Test run calculation for timings of logs.

Actual Things ~~to~~ done:

Device connected to port 1 first:

Run drill for 10 seconds. Wait 5 sec.

→ Run drill for 5 seconds

Device then connected to port 0:

Run drill for 20 seconds

~~Run drill for 10 seconds, after~~

Wait 5 seconds

Run drill for 10 seconds.

$$209 \times 0.1 = 20.9 \text{ seconds}$$

$$(1 \times 256) + 59 = 315 \times 0.1 = 31.5$$

$$((1 \times 256) + (107) \times 0.1) = 36.3$$

↑ 10 seconds
↓ 5 sec

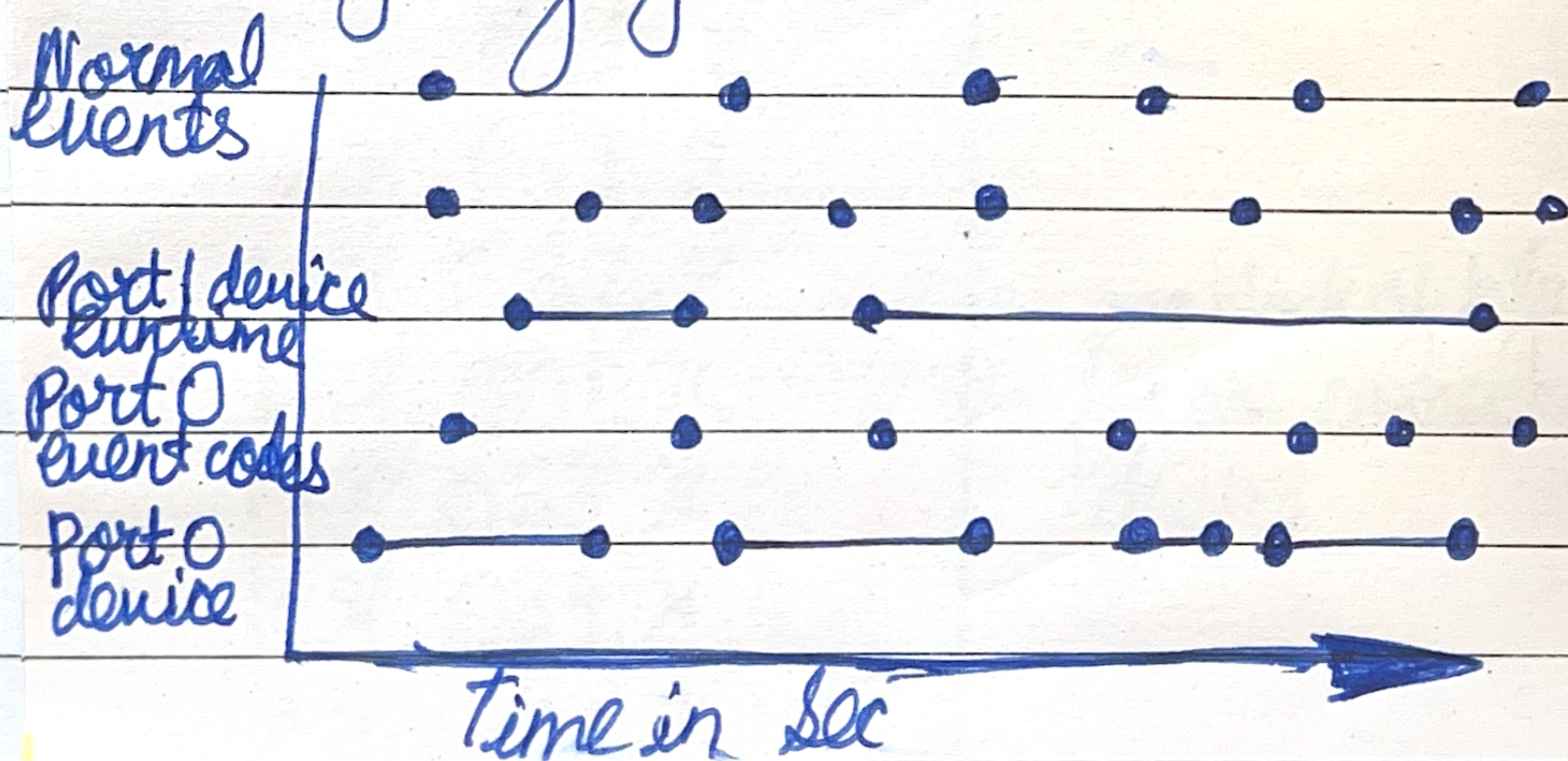
Improvements to graph:

- Make the plot point for "Device connected" the colour matched to serial number instead of port, because the port can be read already from the y-axis. ✓
- Make the popup on "Device connected" have more information about that device.
Possibly even add image of device to dropdown.
- Instead of giving raw binary, return the letters for which each binary "1" represents. ✓
- Make a default function to convert strings to binary. ✓
- When trying to find device id/family ID ✓

Improvements round 2 ☺

- ① Don't make a separate graph for event codes at startup
- ② Don't plot points that have port data AND other information in the last number twice. They ~~show~~ If they have port number info, they should show all expandable data on the points ~~on~~ even if they are port associated.
- ③ Split the graph up even further, so that there are two rows for each port, ~~one~~ one for event codes, and another for ~~the~~ drill runtime.

Example graph:



④ Make the "device connected" dots* have the information of device serial number etc... and device software version, as hover text.

⑤ Information about this:
Even if there are two devices connected at the same time, only one of them will appear on the logs first.

There is a known structure to this, as below:

X	Event codes	X
X	056	X
X	056	X
X	038	X
X	039	X
1+X	037	X
1+X	001	X
1+X	033	X
1+X	034	X
1+X	035	X
1+X	036	X

one block of info for one connected device

Explanation of device-block logs:

$X, 056, X$] $\rightarrow$ device boot + port
 $X, 056, X$]

$X, 038, X \rightarrow$ Device software version MSB

$X, 039, X \rightarrow$ Device software version

$X+1, 037, X \rightarrow$ "Device Connected" + Model ID
+ Port number

$X+1, 001, X \rightarrow$ "Device connected" + family ID
+ Port number

~~$X+1, 033, X$~~

$X+1, 033, X \rightarrow$ Device serial number Year

$X+1, 034, X \rightarrow$ Device serial number (byte 0)

$X+1, 035, X \rightarrow$ Device serial number (byte 1)

$X+1, 036, X \rightarrow$ Device serial number (byte 2)

When converting last number, ~~from~~ do:

① Translate to binary.

② Use 256 weighting for every 3 bits.

info_about_device=[]

Test code to extract information about each device connected to the console:

```
for row i in range(0, len(split_list)):
    column = 1
    if split_list[row][column] == "056":
        info_about_device.append(split_list[row+1][colcol])
```

~~List of Extra info~~



Making changes:

~~line 184~~ → —+—

line 182 → -1 is added

line 114 → -1 4

Improvements round 3:

- Make a yellow line go from device connected to device disconnected, and then layer a green line on top of that during when a drill is actually used.
- Make a list of all the times a device is connected and disconnected, along with time and port #.
 - Then make a loop to go through the list, and make a new loop ~~the~~ which is paired with port number, ~~at~~ and connected/disconnected ~~at~~ times side by side.
 - From this, then make x and y plots lists.
- Add elapsed time drill was running to end point of green line.
- Separate normal events from actual errors.