

24/9/22

(w/ eccu binning (to 1) of spike train see if
different

Global error minimisation function (what is the cost function)

MHz sample test.

Spin test

More experimental results

>Error of quantification per cluster (convol-to steps) %
over.

Switching up generation

Clockwise deviation from ideal. Spacing error.

Ratio of Zc displacement / switching zone size

$$| \frac{^0}{e} = \frac{2}{\text{poles}} | \frac{^0}{m} * \text{Number of zones} = 3 \cdot \text{poles}$$

Full electrical cycles is 6 switching steps 360°_e is 6 steps

$$360^\circ_e / 6 = 60^\circ_e$$

$$* 60^\circ_e = \frac{2}{\text{poles}} 60^\circ_m = 1_e^{\text{step}} = \frac{120^\circ m}{\text{poles}}$$

$$\text{in steps } \cancel{\frac{2^{14}}{360^\circ} \times \frac{120^\circ}{\text{poles}}} = \frac{2^{14}}{3 \cdot \text{poles}} \text{ "Number of digital steps per electronic switching step"}$$

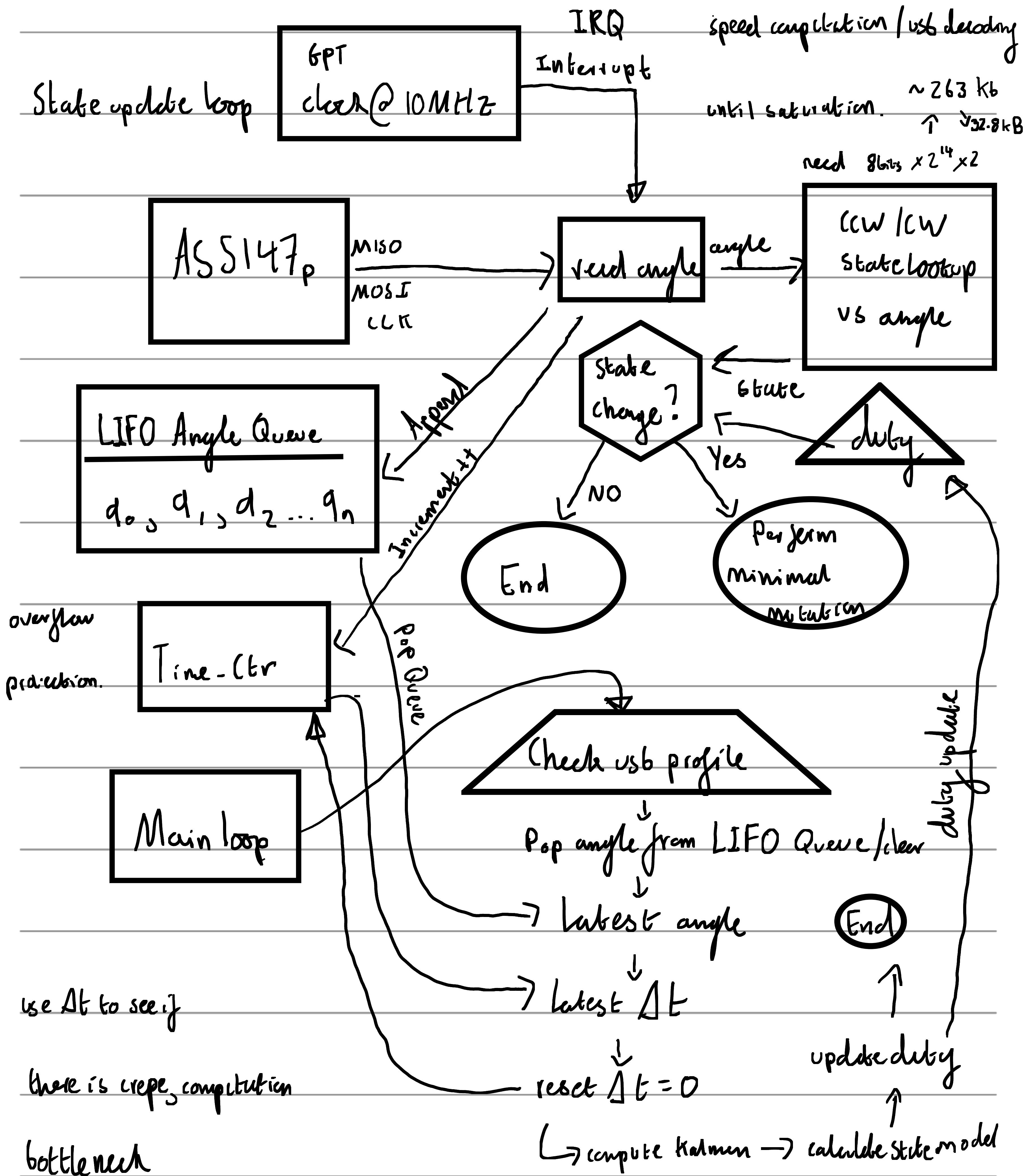
For poles = 14 we get $\frac{2^{14}}{3 \cdot 14} = \sim 390$ steps is the switching zone size.

$$\text{The Zc displacement is } 30^\circ_e \quad d_{Zc} = \frac{2^{14}}{6 \cdot \text{poles}}$$

Teensy 4.0 2Mbyte flash, 1Mbyte ram

Thinking about speed control loop

The IRQ can interrupt the



26/9/22

Will attempt another cw/lcw run of about 10s with
node memory flag set.

Update combine report to deduce mean & error & relative error.

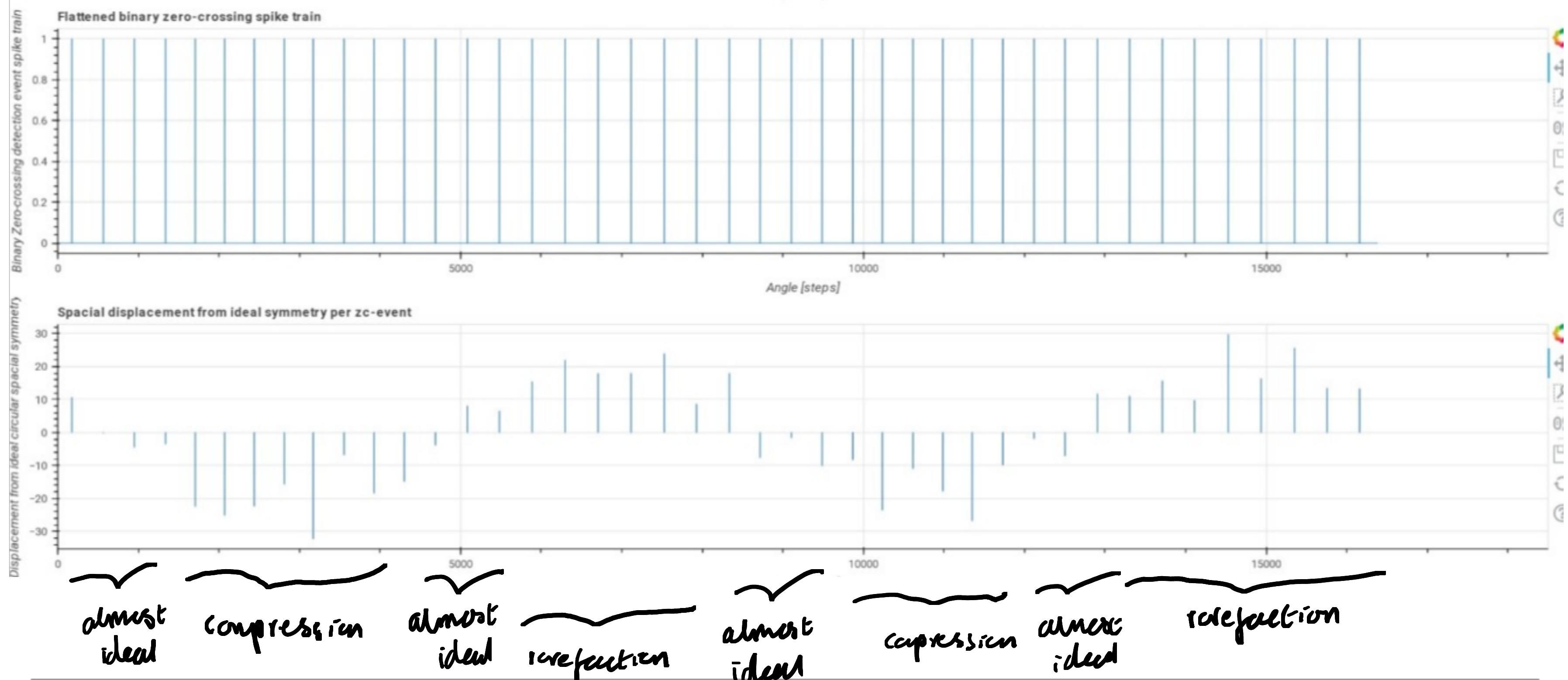
Displacements from ideal now that I have derived the formula, global ✓
error function.

→ displacements from ideal, new-mean: $\{ "z_{c-\text{chun}}": \{ 0: \text{up},$
per chunk get key, value tuple $\text{dist} \rightarrow \text{tuple} + \text{check}$ ✓
interesting oscillating pattern! 30 disp
max

→ cluster error

cluster me cluster 1 ... cluster N ✓
zcris... $10^{\pm 0.1}$... cluster min
✓ channel error avg. of errors max

Global errors based of displacement from ideal, do errors for abs as well as raw value analysis.



Thinking about the above displacement away from $\frac{\pi}{3.14} \approx 390$ steps

Ideal spacing

chart above, interesting it seems to oscillate like a sinusoid with

a period of $2 \times$ the circular period. As the motor has circular symmetry

it make sense that the sum of displacements $\sum d_i = 0$, as

if not the pattern would shift specially on the next turn & it would not be periodically

stable over the circle, its not obvious to me why we get a 2 period sinusoidal like

pattern, perhaps this is evidence of some error based on angle, or

We have revealed some real manufacturing asymmetry for the A2Z1Z motor.

Max displacements on the order of ~ 32.2106 angular steps
from ideal

$$\text{Conversion factor is } \frac{360^\circ}{2^{14} \text{ steps}} \quad [\text{degrees per step}] \quad \xrightarrow[2dp]{\text{up}} 0.71^\circ$$

$$\text{Average displacement from ideal spacing } \sim 16.1303 \xrightarrow[4dp]{\text{up}} \xrightarrow[2dp]{\text{up}} 0.35^\circ$$

$$\text{Smallest displacement from ideal } \sim 0.3578 \xrightarrow[4dp]{\text{up}} \xrightarrow[2dp]{\text{up}} 0.0079^\circ$$

$$\text{Best cluster error } 6.3774 \rightarrow 0.14^\circ$$

$$\text{Worse cluster error } 87.8996 \rightarrow 1.93^\circ$$

Would be nice to know how cluster error correlates to
error of the deviation from ideal symmetric spacing.

Thinking about ZC displacement, if it turns out the displacement from
ideal symmetry has a real physical pattern, then displacing the ZC by 30° en
would be a bad choice, instead take the middle point of adjacent ZC-events.

29/9/22

Next step long duration cw/ccw run ~10s+, see what length
of time is stable.

iteration.

What naming convention for runs XX — Sept — A2Z12 — CW — X
date marsh meter direction

→ 29-sept-A2Z12-CW-1 99.95% match

rate

→ 29-sept-a2z12-ccw-1 99.96% match

rate

Combined reports for these two cluster error went up, deviation

from ideal went down. Interesting that the error (cluster) does
appear dependent on runtime! Big incentive to make sure cw/ccw
runs are of equal length!

How can we ensure run lengths are nearly the same?

Have host PC send the number of iterations required to
teasy (ADC), have that teasy only poll the slave n times,
have master teasy send termination serial print to host
so host can auto exit, giving the user some indication
of termination.

Then with source program for ADC teasy, have a flag for
run size, lots of short runs should minimize detector error, & lots
of these runs. \rightarrow Better capture \rightarrow Auto analyse command!

We probably can't prevent skew from misrabdes!

NuForce DDA 100

pink capacitive

Ohm meter
VU meter

Blue, detent mix

orange (drilled)

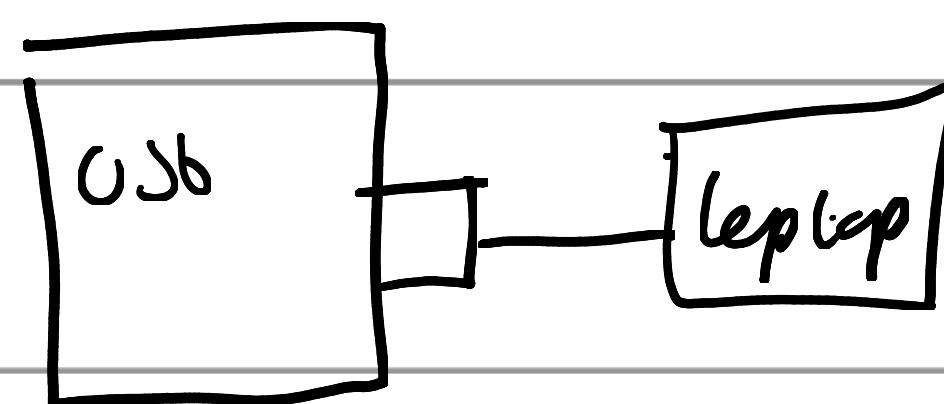
C1 30

chain 30

OLED

Inputs

Outputs



single button

Target Speed 33.3 rpm 33.0 X

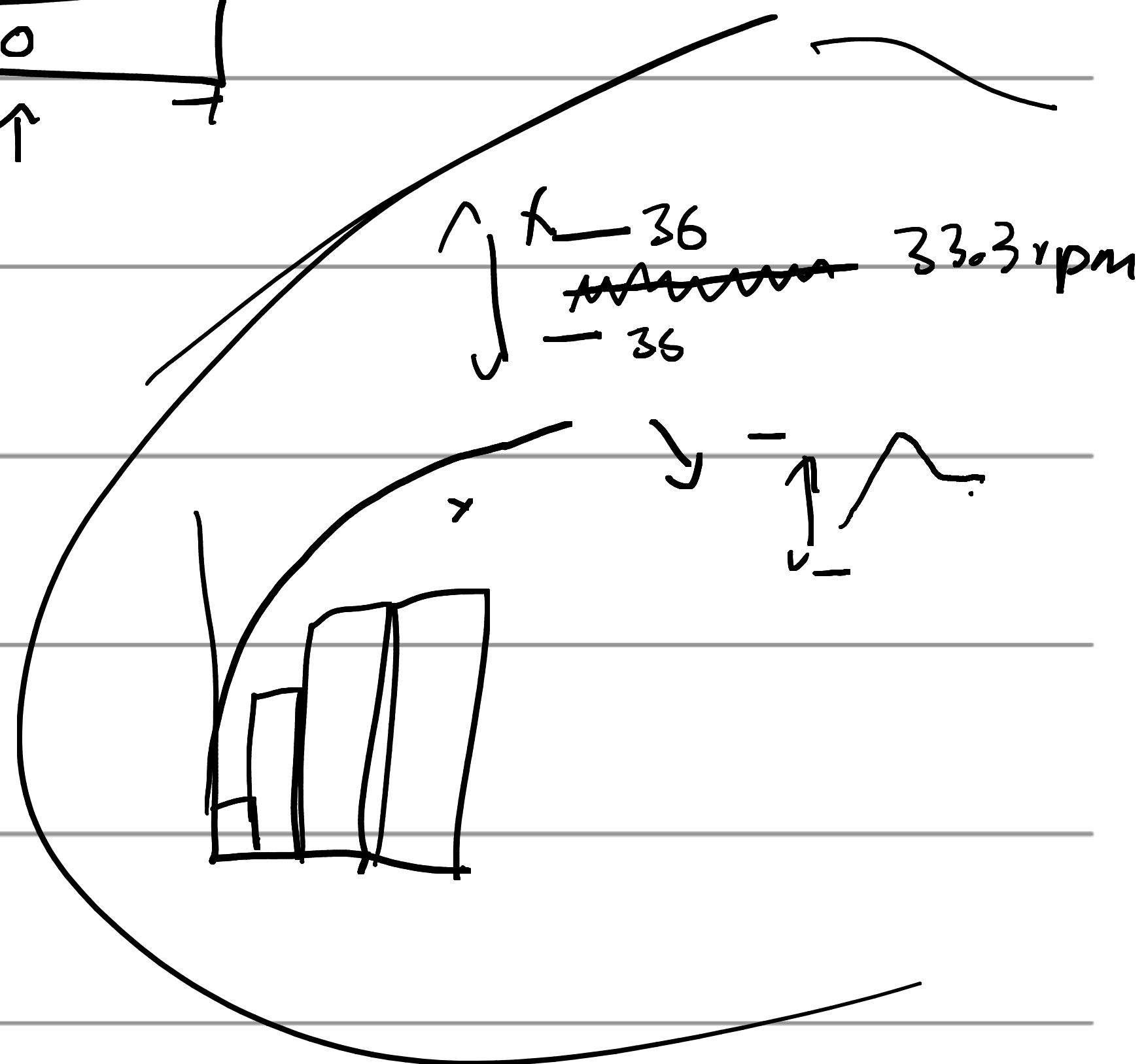
pitch (linear)

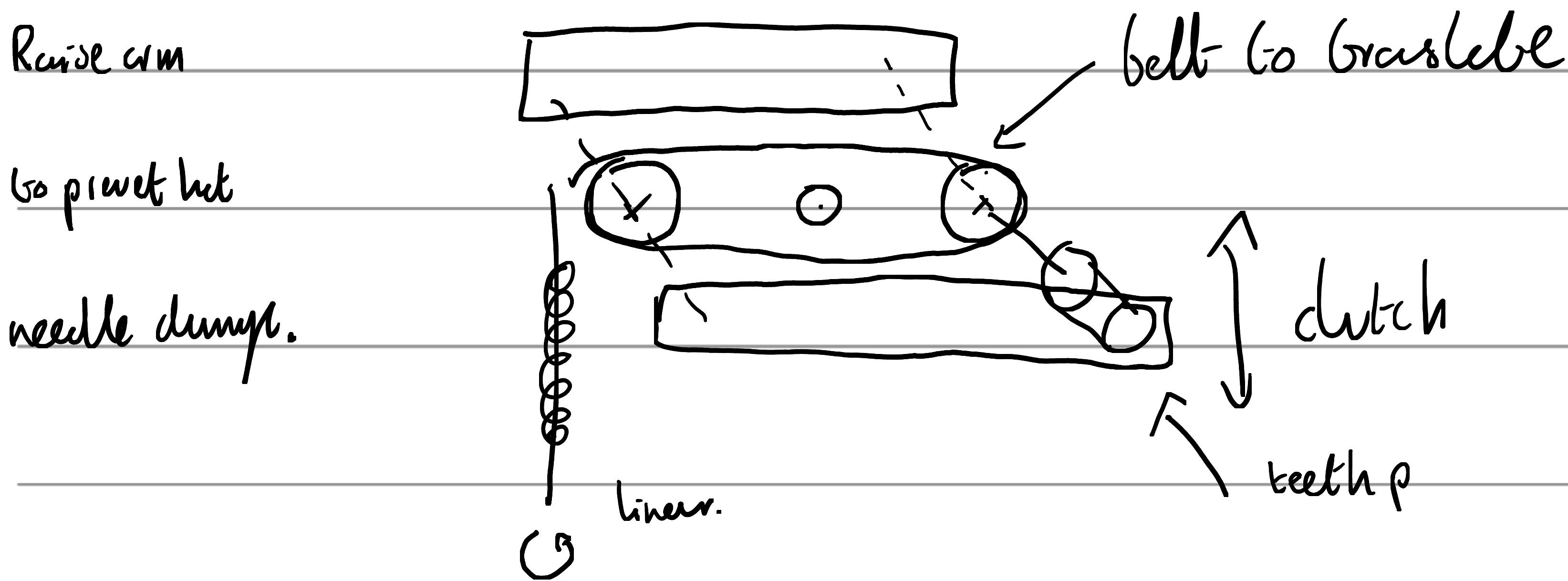
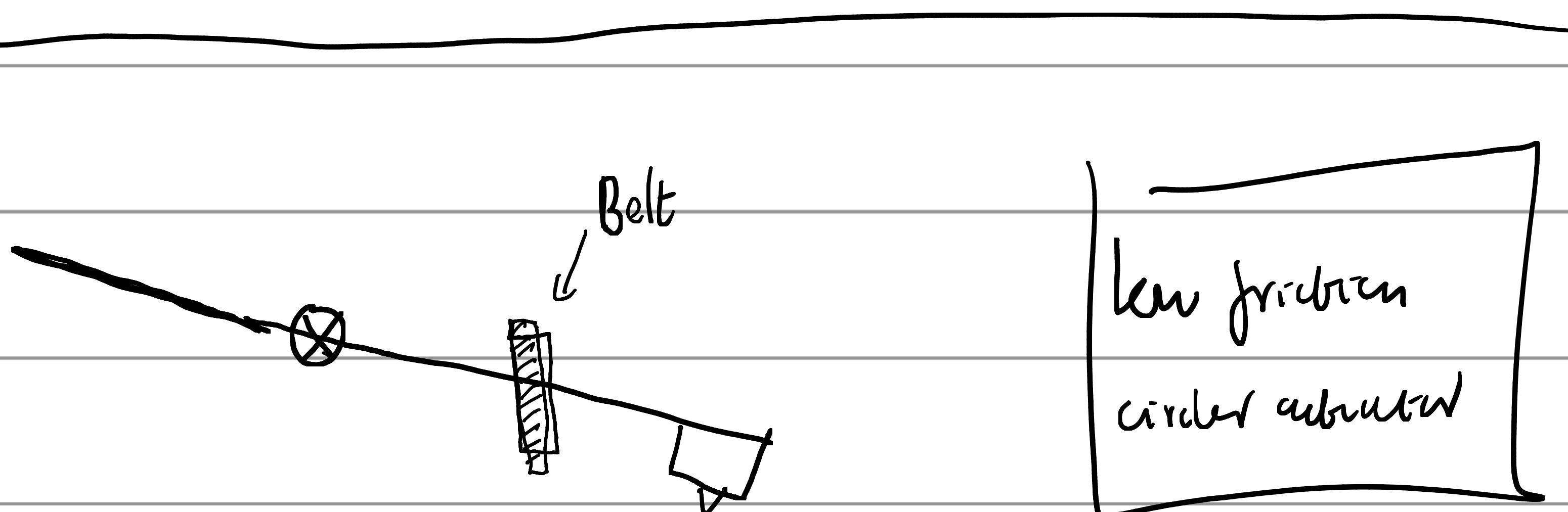
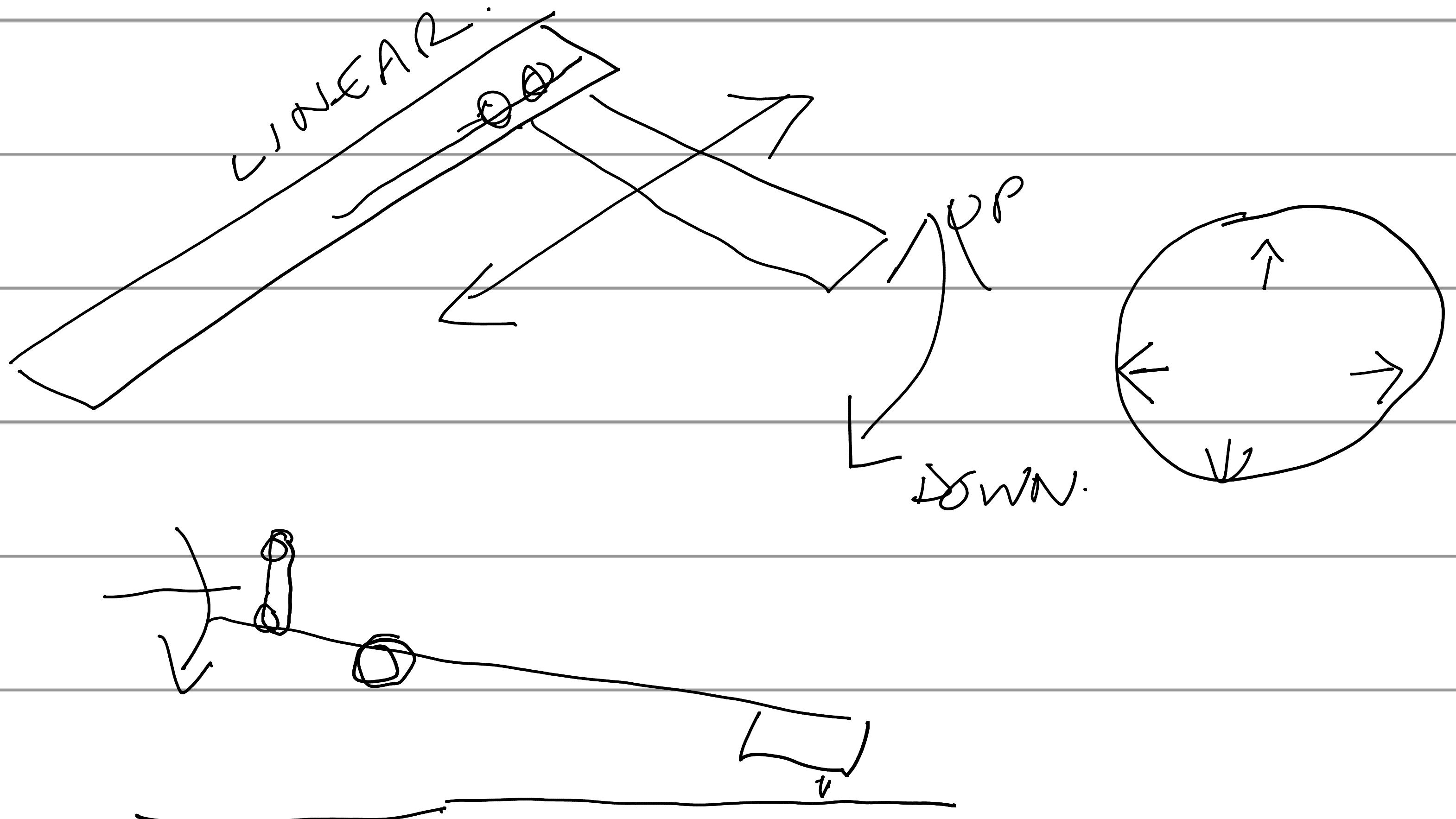
True Speed 33.2 (diagnostic usb)

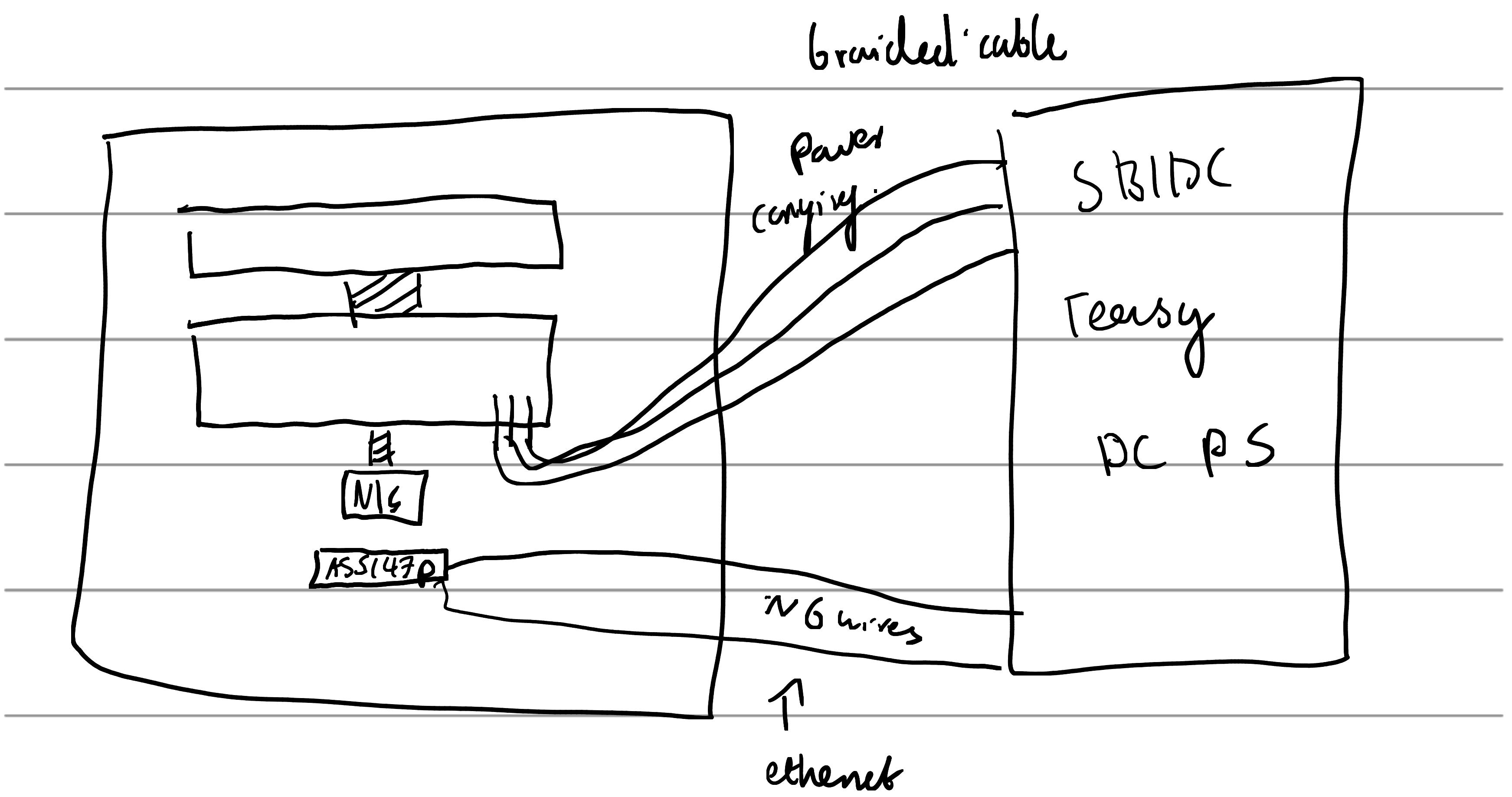
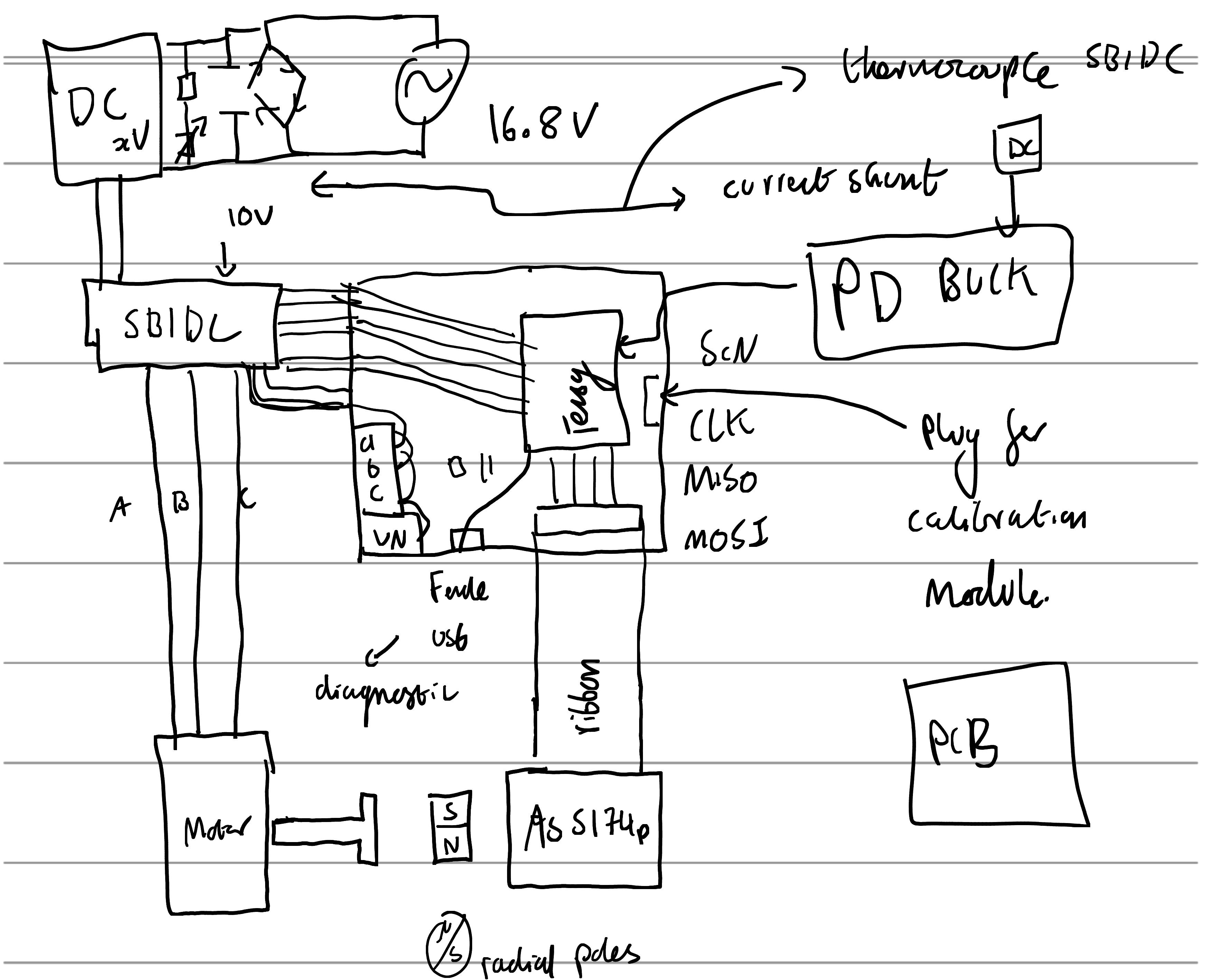
speed (button)
dial

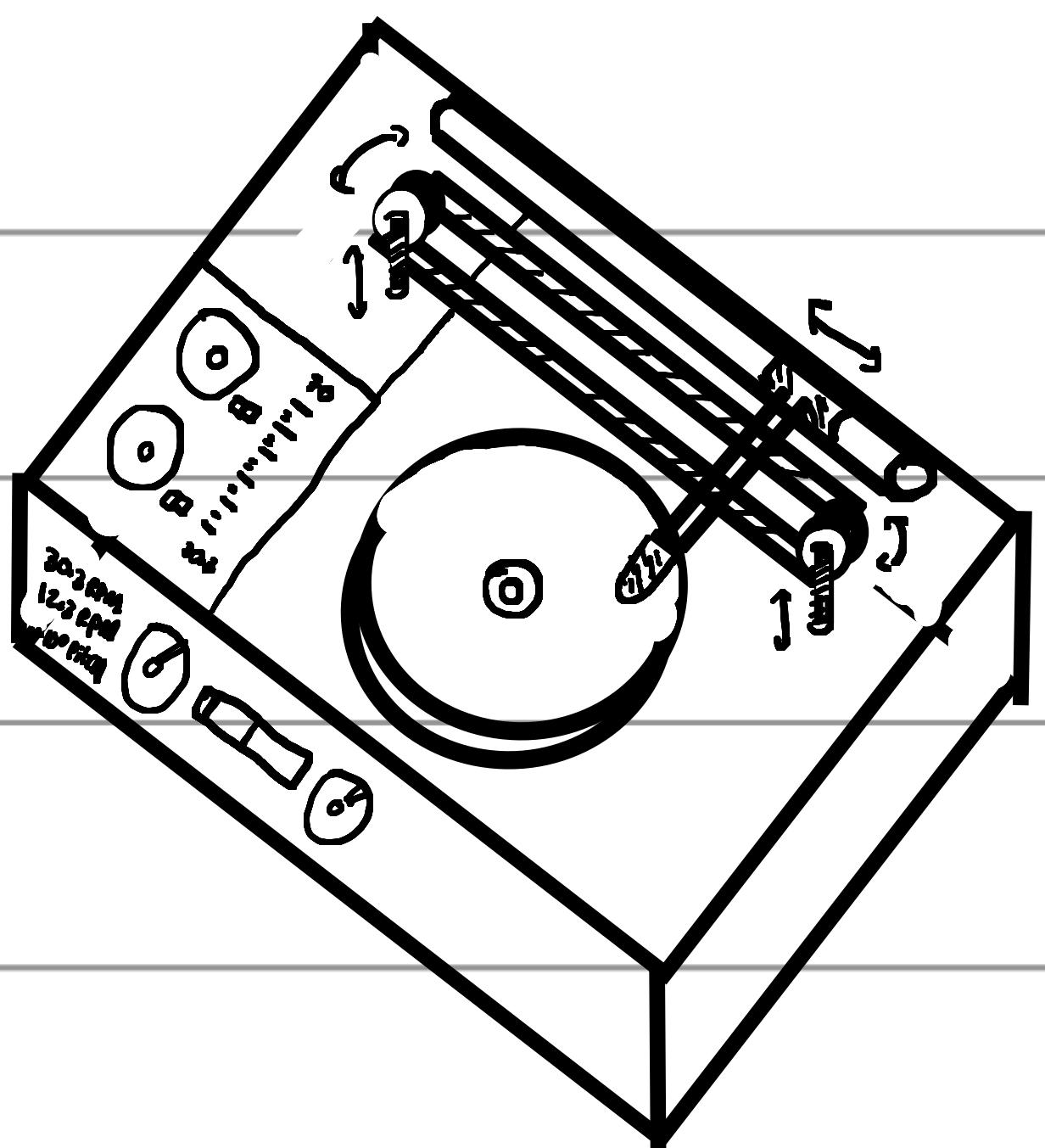
Pitch 0

↓
coarse grain
vs fine grain









28/9/22

(created sample rate program, capturing cycles @

1,041,611 Hz ... need about $5 \times$! or at minimum 2.84 MHz

removing nano second delay (3ns) ... 1.515064 MHz
removing parity check 1.6744 MHz bad values

running at 816MHz 2,371,990 Hz bad values.

Remove ns delay blocks reading cycle

try 2ns fail

try 3ns 814MHz fail

3ns 720MHz -71.412 MHz

2ns 720MHz → fail why rye missing 8000-16000

$$\text{Sample Rate} = S_R = \frac{2^{14}}{60} \cdot w[\text{rpm}] \quad \frac{60 \cdot S_R = w[\text{rpm}]}{2^{14}}$$

3ns 720MHz $S_R = 1,411,700$

$$w[\text{rpm}] = 1,169,8 \text{ Hz}$$

Code optimised for "fastest" 1,451 MHz

720MHz / Fastest | 3ns

goal 10,416 rpm

1,451,500

$S_{S3} | 5.55 \text{ Hz}$

→ 0.01 steps so

$$\text{best case sample every } \frac{2}{2^{14}} \times 360^\circ = 0.04^\circ$$

This is for target number

10,416 rpm.

For azz12:

target

16,800 Hz

$$\frac{60 \cdot S_R}{2^{14}} = w(\text{rpm})$$

$$w(\text{rpm}) = 5,169.8 \text{ Hz}$$

$$= 0.307$$

16,800



Sample every $\leftarrow \checkmark_{0.307}$

0.671 $\leftarrow 3.2496 \text{ steps}$

Min cluster error is about 6.377 steps. X

Trying crasy loop optimisation

1,478379 MHz

1.8% improvement from JCSL

720 MHz, parity removed.

trying remove 3ns delay on last 2 bits.

7.8% improvement

1,565 MHz

~5731.73 rpm.

Remove all logic from last 2 bits, apart from
value shift.

13% improvement

1,651 MHz

5% improvement
from shifting delay

6047 rpm

Convert 16 \rightarrow 32 bits mask 14 1's

goal 2.084 MHz

2.3% input from 16 bits @ 1.69 MHz

alternate 3/2 ns @ 1.743 MHz

all 2ns @ 1.88 MHz

Try no op for wait 1.8996 MHz

6956.73 rpm

→ 800 sans
slowly unusable.

1.9 MHz With fewer first no-ops

goal was
2.084 MHz

from 0.2

1.935395

7687.3 rpm

$\sim 86\%$ improvement.

10,416 1.46 steps.

$$@ 16,800 \text{ rpm} \quad \frac{16.8 \text{ k}}{7087.32} \approx 2.37 \text{ steps.}$$

ie - reading the data sheet 2^{14} is not the accuracy of the encoder its 2^{12} , I can see ± 4 steps only so this makes sense.

$$\frac{60 \cdot S_R = w(\text{rpm})}{2^{12}} \quad S_R \text{ best } 1.935395 \text{ MHz}$$

$$w(\text{rpm}) = \frac{281350 \text{ rpm}}{4096}$$

1,935,395

looking at uncalibrated single direct dither errors they were on the order of 6 steps, as we are only getting ± 4 steps, the error is only slightly more than min step size. so worse

target
x

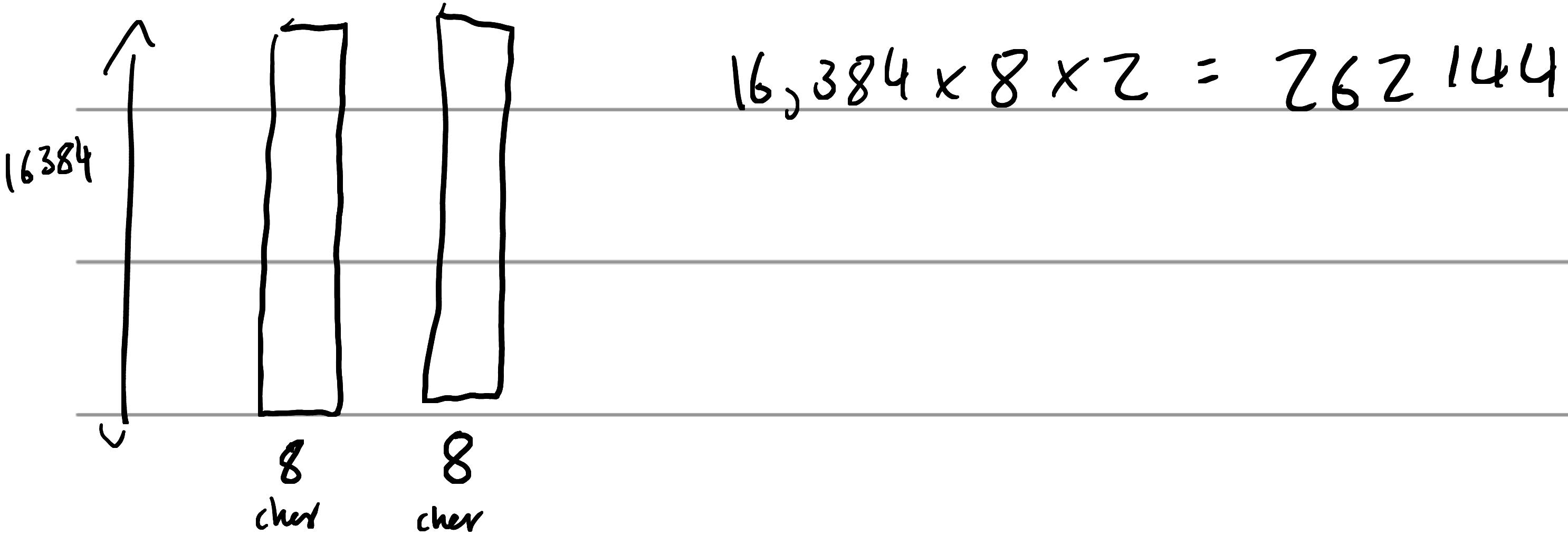
worse case 16

0 4 8 12 16 80 ± 8

| ~~xx~~ ~~xx~~ ~~xx~~ ~~xx~~ |

Another thing the config map was going to be

CW CCW



But in reality there are really only $4096 (2^{14})$

$$\text{so } 4096 \times 8 \times 2 = 65,536$$

would need to build a 2^{12} map &c for cw(ccw), should divide

we angle we get by $4_3 > 7^2$ bitshift. (CANNOT DO THIS) get odd angles at high speed

Should see how fast we can get with overclock using delay
nanoseconds. (try 1ns) interpolation
No NEED

(rate different versions based on different MHz (any clock))

NO NEED
stable @ 600 MHz

(rethink to combine cluster errors ignoring systematic error (multi-scale))

29/9/22

Re-testing fast angle encoder method.

	Smallest freq	Rpm Max	Round down
@ 720 MHz (max double)	1,924,956	28,197	
@ 600 MHz	1,604,086	23,497	
@ 528 MHz	1,411,581	20,677	

To be safe if we are going to do a loop for this we should back off the max speed by at least 10%. even @ 600MHz, with 10% reduction

we are over 21 k rpm.

10% reduction

Renewing clocks
after angle measure.

720	1732460	25377	304
600	1443677	21147	184
528	1270422	18609	112

Important observation, raw angle data does include odd numbers! So the T517up does interpolate at high speed!

Neat! Will have to keep ZC map as a 2^{14} bit number, as bit shifting odd numbers will loose precision. Fine.

So what is left to do:

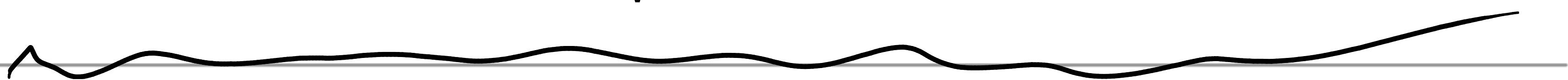
→ Combined cluster errors + burst after shifting clusters from different rows over the top of each other to reveal a cluster error divorced from the systematic displacement.

→ Programmable run-times & (easy ADC exit and to host).

→ Perhaps improvements to sync/sync so that multiple runs are easier to perform.

→ Final cw/cw displacement maps.

→ ESC high speed program



Thinking about how to get cluster errors without system's errors.

We have the uncombined histograms Liderbjer

14

dataset | (11)

۱۲۱

11

debaset 2

11

11

11

identifier

60000000

A row of ten vertical tick marks, each consisting of a short vertical line with a horizontal bar extending to its right.

2222222

We have reorganized
our

don't need

Convert this to

debut set cluster : } [angles,] → debut set cluster cluster mean

Then receive histogram (nearest int) with errors

mostly done just need to plot valves + best.