

Zahnprofil am Profil

Klemmschiene nicht über Profil

längen neben Zahnpföhl

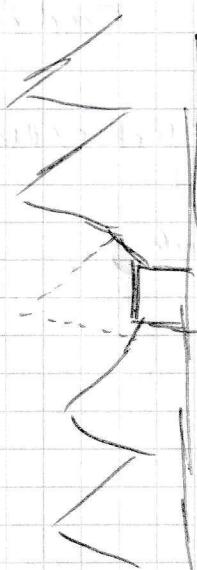
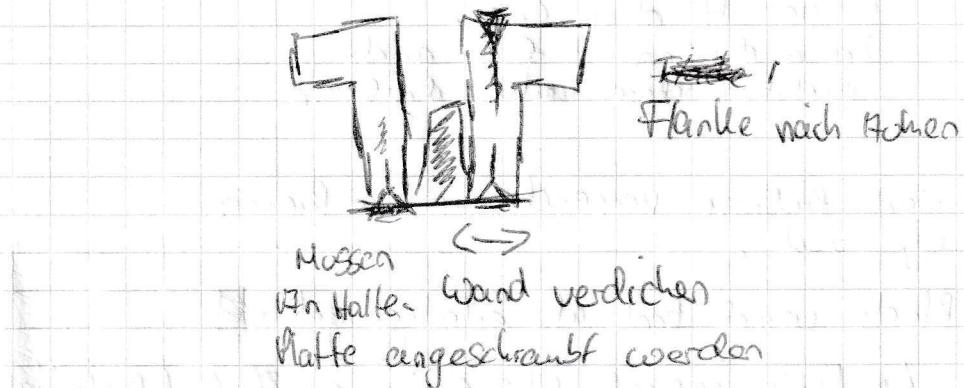
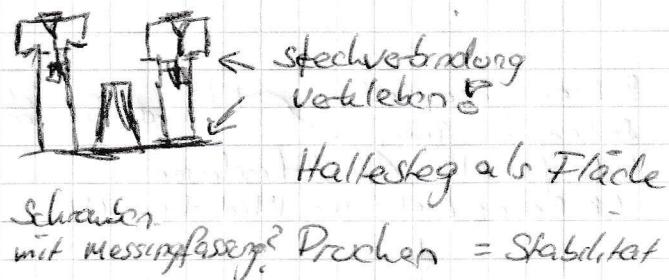
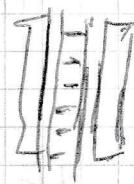
1 Element aus

5 Teilen

1x Z Profil

2x Haltesteg

2x Halte "Lippe"



Löcher im Z Profil zum Anbringen?
Von Mitte versetzen!

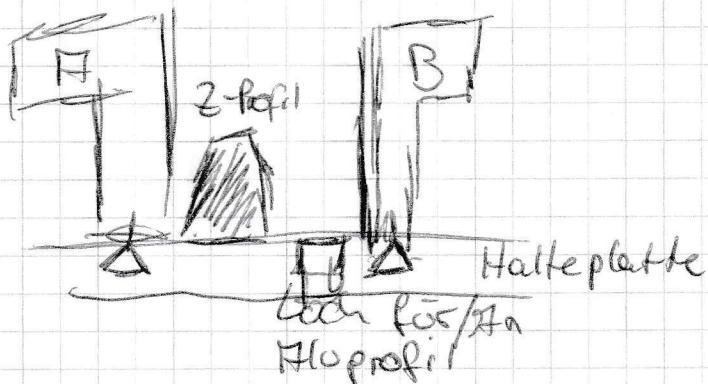
1 Element

↳ A 1x

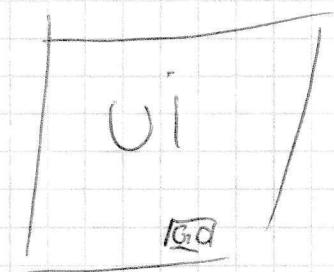
↳ B 1x

↳ Z 1x

↳ Halteplatte 1x



Python for FTW BT Modules



Wertefelder 12x Höhe
12x Breite

Buttons → ~~speichert Data~~
~~in Json~~

+ führt Script aus

Buttons speichert → kann überarbeitet werden oder neu anders? keine Bestätigung o. Status
+ Prüft alle 12 Verbindungen
+ Rotes Licht an Modell in UI

Wenn F ohne Cores.
Dann dürfen B & C auch
nicht. Andere Reihen dürfen B

→ Nach Prüfung versenden in 3er-Paaren
(je Block)

Block wird bool als erledigt gekennzeichnet

↳ Fack visualisiert Grün pro Block (Bestätigt nur Msg.)
Angekommen

⚠️ braucht Json für Init → welche ID = Welches Modul

Visualisierung an
richtiger Pos im Overlay

Cam Model

► 1 Cam

↳ 1 Motor

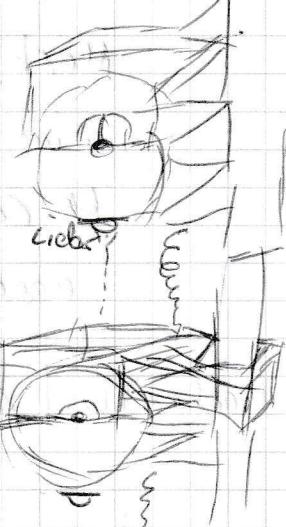
↳ Lider-Sensor

↳ Servo

↳ ESP 32 (BT)

Profil mit Zähnen

Zahnrad am Modul



Strom wird
durchgereicht

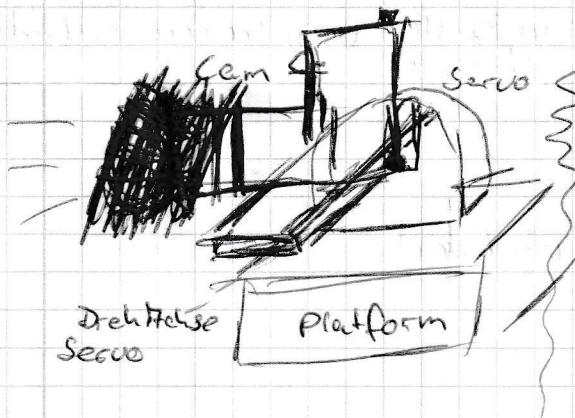
Fahrrichten
nach unten

Nachteil: Kettenhaft

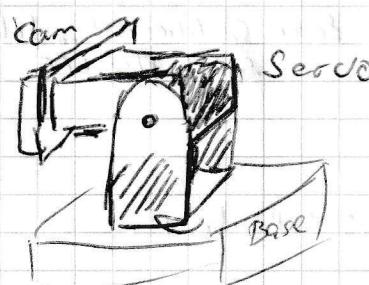
Software: Relative
Höhe in RL in UI
als Absolutwerte Höhe

Versteckungsschutz

Wema 17?
Zahne Profil Groß
Zahnrad klein



Servo mit
Bügel (siehe 180kg Servo)?

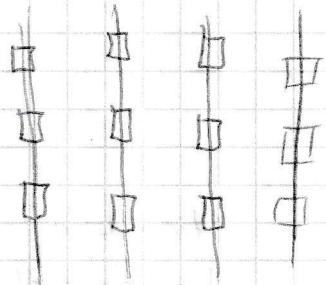


~~Halter nicht fest genug für Profil~~

90° drehen oder weit nach vorne setzen (Belastung an Profil)

Software:

3x4 Knoten



Mesh! unnötig 12 esp BT an PC
PS: wir Hat
wird nur vor Scanvorgang gebraucht
hach
→ Cam wird an Pi über Lattepunkte "Gateway"
angeschlossen

| | | | | |
|--------|-------------|---------|------|--------------|
| Header | 4 Knoten ID | 4 Servo | Höhe | von PC an PC |
|--------|-------------|---------|------|--------------|

| | | | |
|---------------------|------|--------------------------------|---------------------|
| Comport (Knoten ID) | Soll | Δ SollHöhe Von PC zu PC | Bei Höhe=0 = Hos |
|---------------------|------|--------------------------------|---------------------|

Daten von ist/soll in Jason da. → in 3D Viso vom Scanner einbindbar?
Sinnvoll?

Funktionen:

Must → Calibrate (Fotoleveling via Lidar)
↳ inkl. ACK

Calibrate Cam Δ

↳ wir gehen davon aus das Servo funktioniert... Fehler?
Scout FSS5600

check Status

↳ wollen Verbindung estatus + Node ID
↳ evtl. Alt vom Lidar

Cam Modul

► 1 Cam

↳ 1 Motor

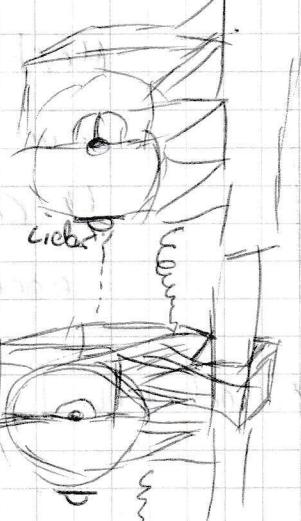
↳ Lider-Sensor

↳ Servo

↳ Esp 32 (BT)

Profil mit Zahnen

Zahnrad am Modul



Strom wird
durchgereicht

Forsrichten
nach unten

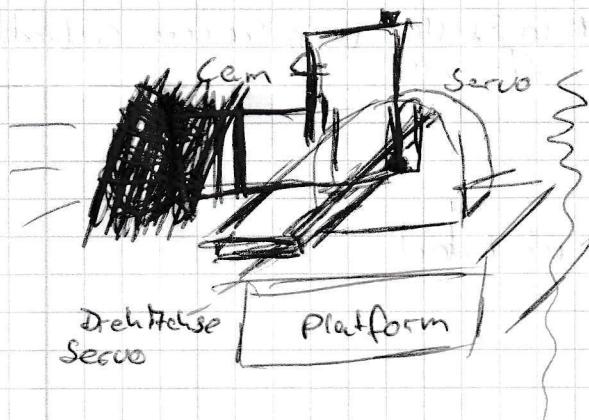
Nachteil: Kettenmaß

Software: Relative
Höhe in RL in UI
als Absolutwerte Höhe

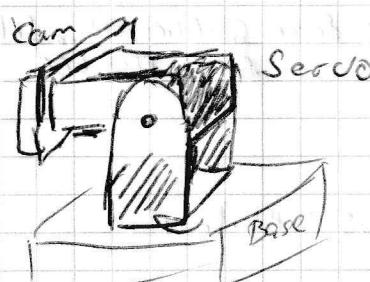
Versteckungsschutz

Name: 17?

Zahne Profil Groß
Zahnrad klein



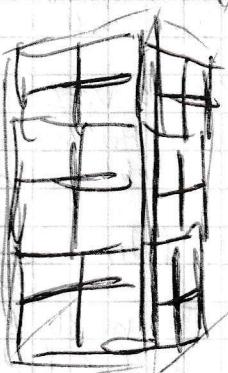
Servo mit
Bügel (siehe 180kg Servo)?



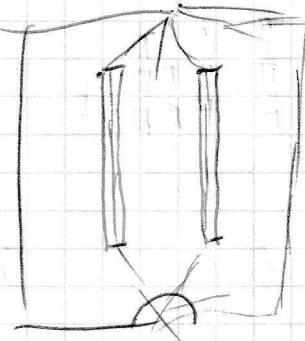
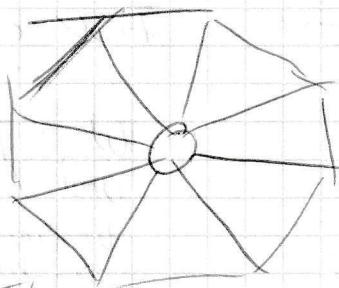
~~Halter wird dort aufgefunden~~

90° drehen oder weit nach vorne setzen (Belastung an Profil)

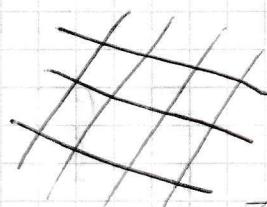
- Kammer Octagon | Wie ein Zelt Umbrella



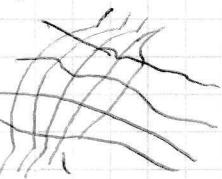
40x40 Platten
Rücken



"LichtGitter"-
Stäbe
Bilden
Waben/Muster
Fb und
Zeigen
in die
Mitte

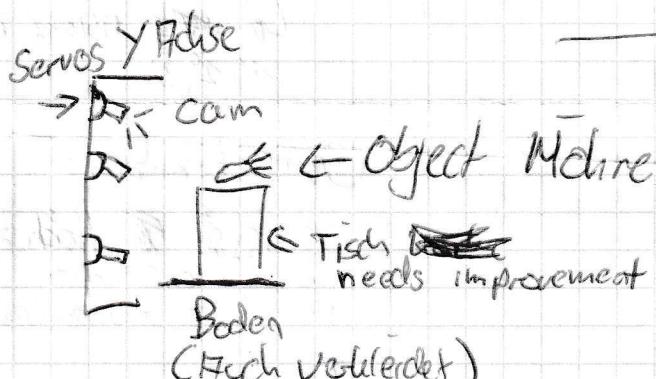


Verzierung
Muster
je Perspektive -



Perspektiv wechsel
(Objekt Größe Primär)

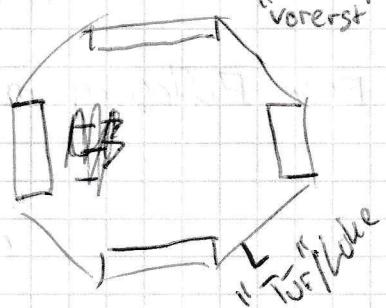
Kann auch genauer?
→ weil mehr unterschiedliche
Perspektive?



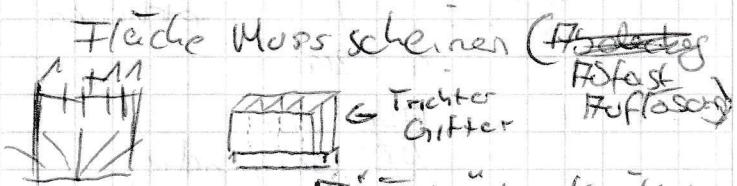
Innen Raum selbe Farbe / Material
(Das Doppelte Schwarz?)
Soll intensiv sein

S.10-S.11.contras

Warum nicht 8+0+0?
7+0+0 = keine "zu aufwendig"
"vorerst"

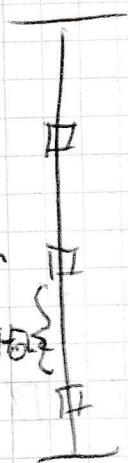
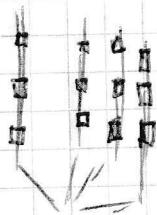


Dach nein zu Stufen



Fläche muss scheinen (Abstand
Rofst Roflasch)
„Fränen“ Lichtmittel
so dass Ränder weniger
Licht
Lichtgitter
von 4 Seiten
Ober? Unten?

Aufnahme geräte anstreben zwischen den platten



FOV ca. 160°

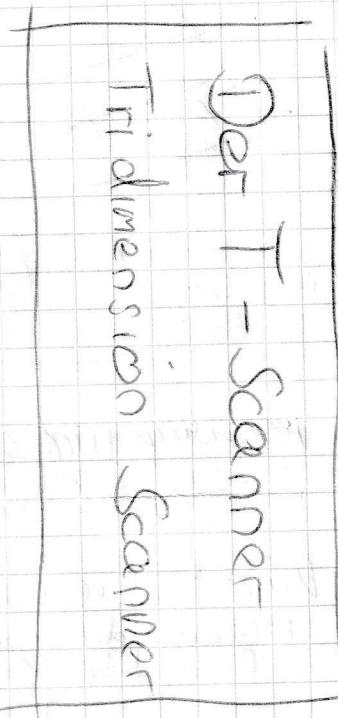
Wenn Kammawinkel und Lichtgitter konfigurierbar? bzw. Berechnung der Krümmung des Schaffens Nestes

Cams verschreibbar? um Auflösung ~~an~~ der C's der Objektgröße anzupassen?

+ Winkel für Y Fläche über

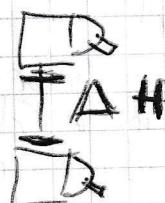
Je nach Grad andere Perspektive

~~ca.~~ 100° auf Zeitlich
 $1^\circ, 2^\circ, 3^\circ$

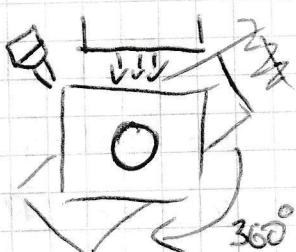


Umsetzbar

Wenn ~~die~~ Höhe zu nicht verschoben wird \rightarrow zu aufwendig



müssen nur abgesunken werden



Auflösung durch 3 Dimension erhöhen

Für nur 90° Notwendig

4 Perspektiven mal \times Y Fläse Auflösung in Geräte oder Pos

was Gerät ~~ent~~ in entspricht

► Mit Python

↳ Webserver starten auf Port n

↳ Ordner freigeben über HTTP

↳ Hatlt : json.Action/Tasks

JSON.Protocol-N

↳ Ordner 2 freigeben

↳ Hatlt Bilder in Ordner

↳ Ordner N

↳ Bilder.Pic [CamName. No. #]

Sortierbar nach No. # & CamName

↳ Ordner N+1

↳ ...

Impuls zum auslösen des Seriale Vorgangs!

⇒ Zeitgesteuert → Feinfach

→ Prozess muss von LPC ausgehen

↳ MQTT Style

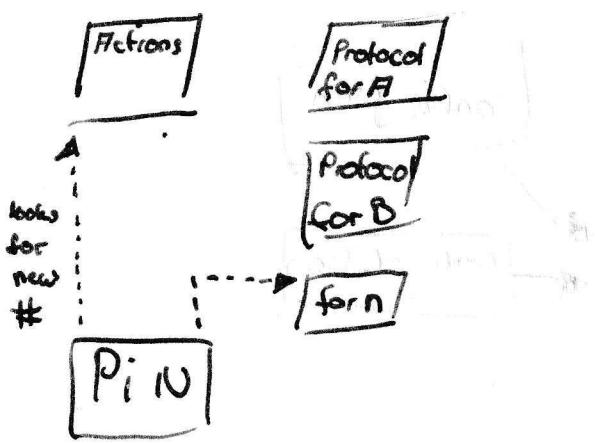
↓
↳ Ordner mit .json

↓

Pi N looks in this File

↳ [Action for A ; Action for B] [# No.]

"# lists Main



Write in second .json

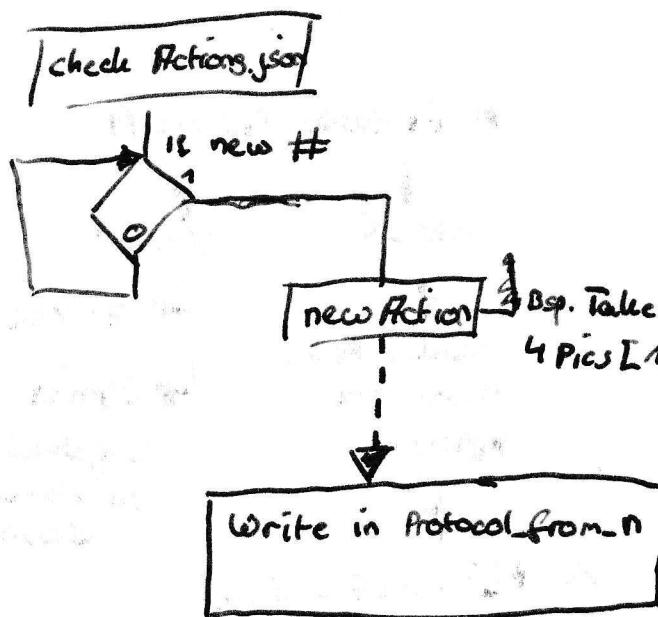
↳ Action for n

⇒ {

Timestamp, #, Which Action

→ Each Node }

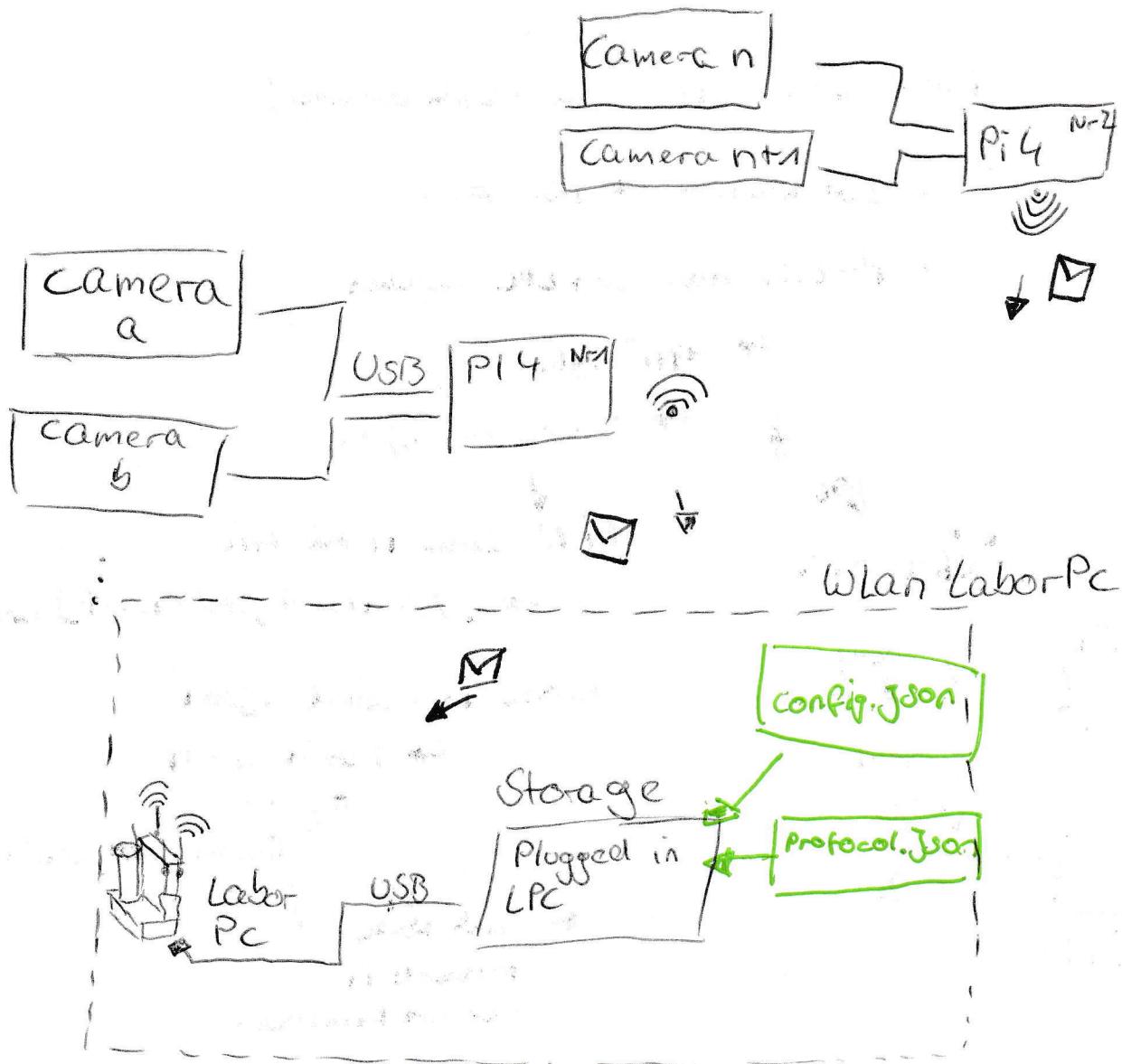
Protocols in
separate Data stream



- [#; Referenzno., Timestamp, Status]
(mit in der
Berechnung der
"Pakete" mit
Pics)

↓
Error
etc.
Possible

LPC can check for no. # if Action
is done or [Status].



• Ein Pi Model can hold
4 Cameras

• Pi connected via WiFi
with own Network (LPC)

• End Hardware = LPC

► Pi takes Picture A

↓
store?

↓
Sending
Packets with
Pictures

Pos A [Pic Cam A, Cam B...]
Pos B [...]

Muss Winkel | Camera | Zeitstempel?

→ erstmal zum
Profiling Programm
ablesen

↓
stream!

↳ zu viele Redundanzen

↳ Option sollte
eingebaut werden
um Hardware zu
checken?

► Generierung der Kamera Container?

↳ Dockerfile

↳ Python Django Server

↳ wird definiert durch Post befehl

"change" config.json → Hinterlegen triggert alarm erzeugen

Braucht jetzt ↓

Get Status CamDDX | Returns Possibility/function of camera stream
Return Bool

Comfort Define Cam

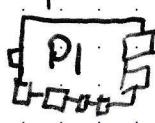
Get Log CamDDX | Returns log msg

↳ last few Actions or fails in detail



Post config CamDO

GUI möglich



► Comfort Cam

► Resolution

► Fangle (Servo controlled separat)

► Sonst.

ATMega328P

→ controls

n Servos

Set Fangle Motor X

Get Fangle Motor X

Container A

~~MAIN~~

- Save Config.
- trigger script

Docker file
(inkl. Layer 2#)
Creates CAMDO

Container B

CamDO

Cam

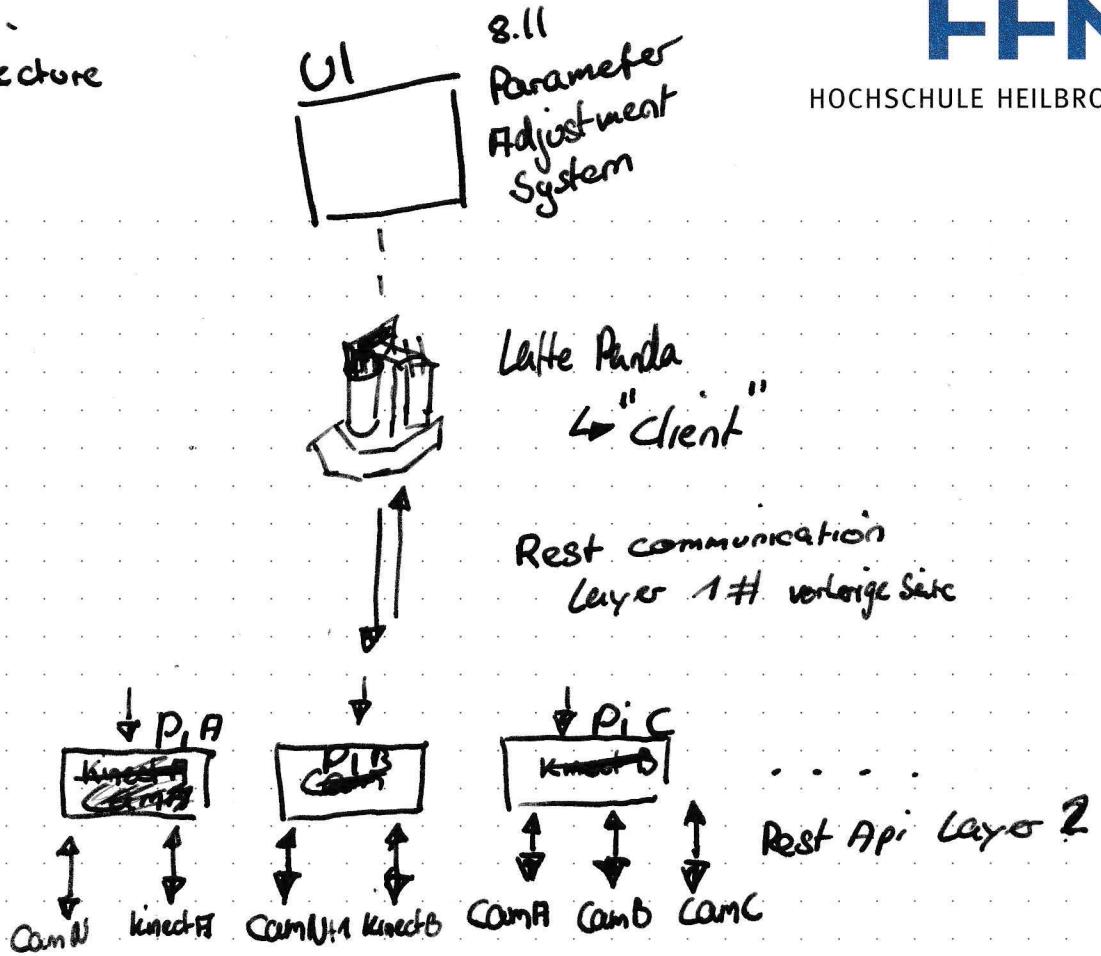
A) möglicher Debug Prozess: Set Fangle/get Fangle → trigger Pictures again set Fangle → www.Pictures-heilbronn.de

Status als

Geter Server

→ Compare Pictures

Architecture



Layer #2 :

~~Post | shot Picture~~

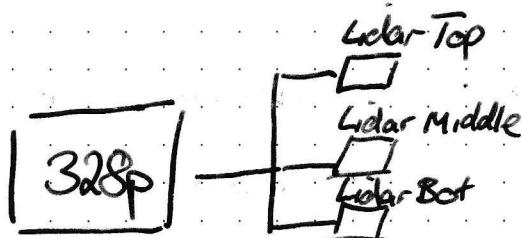
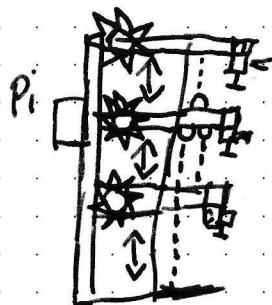
get | get Picture → shot Pic

Kinect : return ← save Pic

get picx |

Post | config | Resolution, etc |

Höhenmessung der Kameraframe



Richt cm Resolution? Dann TF-mini oder Luna

→ Kamera mit I²C II Uart

I²C Switch
wenn Addr nicht
auspassbar

Gedanke:
4 Lidar
+
Redundanz erzeugen

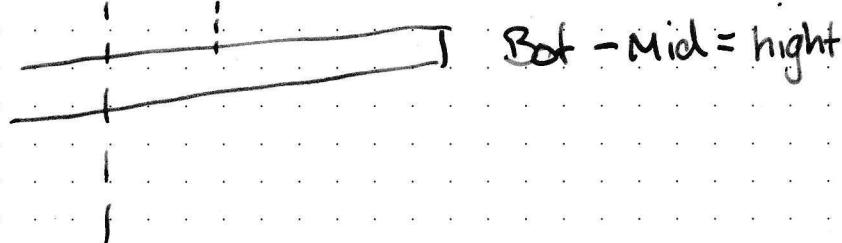
Top + Bot = height

1 Kettensmaß

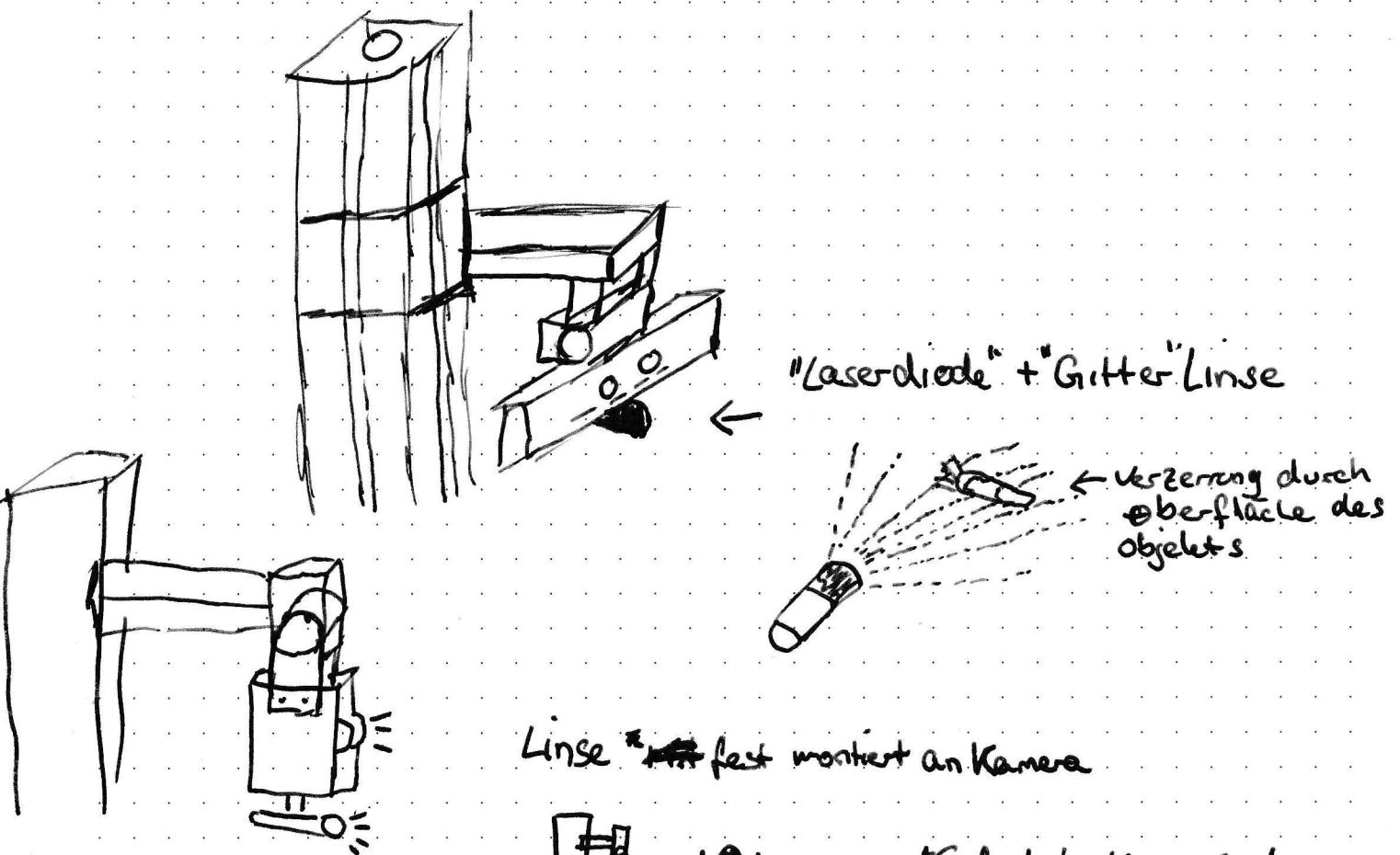


Bot + X || -X

somit 2cm? - schlecht



Δ Lasergitter (Durch Verzerrung Tiefe ermitteln)



Kreativ! Wenn Cosplayer
gescannt werden → Model
als Spielfigur (3D Model)
erzeugen?

↳ Dann können andere Tools
zum "designen" von Spielcharakteren
eingesetzt werden.

Dsp. Nekomimi kann bei Video-
spiel Firma Mitarbeiter und
Kostüme machen.

↳ Kann im Laufe
nachbearbeitet werden

6.11.24

► Kinect Odysee (Win 10 Lettelpanda)

- ↳ Mit OpenCV (Kein Kamera Stream erhalten) ab oder bis 2
- ↳ Demo Programm braucht Legacy Python Version (3.8)

Kinect SDK installiert

- ↳ Danach kein Kamerainput über Win 10 App (Kamera)

- ↳ kein OpenCV Cam Port

- ↳ out of Range Nicht keine Cam erkannt

Problemlosung:

Pi 4 mit 20.4 LTS Ubuntu (wird in Tut. verwendet)
+ #

Django Server:

► Tutorial

► Server läuft ► Migrieren von Änderungen

► /Endpoint gesetzt

Meet IT

IDS Cameras

USB

↳ Haben Interrupt Leitung & für Tofos schließen

7.11.

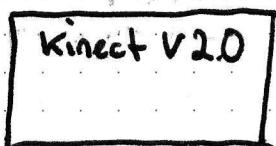
► Kinect Odysee 2

↳ Kinect DS325 sozialistisch umgelagert

↳ CamStream bekomme ich instant (ohne SDK)

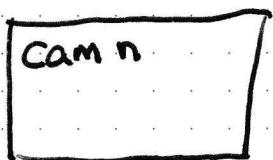
→ Tutorials vorhanden (auch alte SW-Versionen)

Dockercontainer



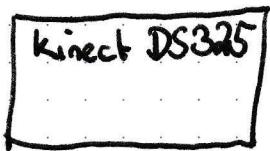
► Python 3.8 Ubuntu example
(4 pics + 5 tcs)

| Legacy Software



► Bild

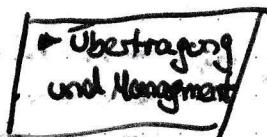
| State of the Art Software



► Multiple Videos

| Legacy Software

~~► Im Docker Container Cam n läuft auch das Main Program welches den Datenstran~~



► Befehl: Schreß Foto

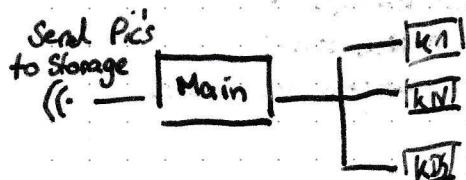
↳ Post Statement in Docker Container Kinect etc

► ~~Start Server / Endpoint → Return~~

► Befehl: Get → Forder Bilder Rn,



In Jeden Container läuft ein Django Server
diese bieten eine Rest Api Rn (Get, Post, Put, Delete)



Zusammen

Prototyp mit Partnern

↳ Ubuntu 12.4/n

Version :D

↳ Django Server evtl. nicht möglich

→ Existiert Basiccode? (DSKinet Disp war in Python 2.7. | gibt's 3.12?)
Cam von 2012)

1 Input & 1 Output

Einschub:

↳ Rotierende Bodenplatte

↳ Eigener Pi

↳ Selbes Konzept wie bei Bilder

API - Cam Server

Post | Post config (num.of Pics, Resolution, Settings)

Post | Take Picture → Save with No.

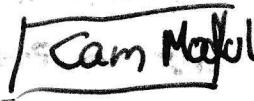
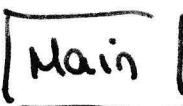
get | Returns SavedPic ↗

delete | delete last Pic.

get | debug? | Wording | → Bool | Log Msg. ↗ last Pic No.
↳ Status
↳ sons. log msg.

D. container

D. container



Post config ↗ save in json

Post Picture (Go)

Cam Routine → save Pic

Get Pic

Pic as format ↗

Return Pic

Protocol in
Config.json

→ Falls Fehler

↳ Normal Status

delete

check for ↗

If ok → delete

get debug

In Config.json

Bool oder Message
"log"