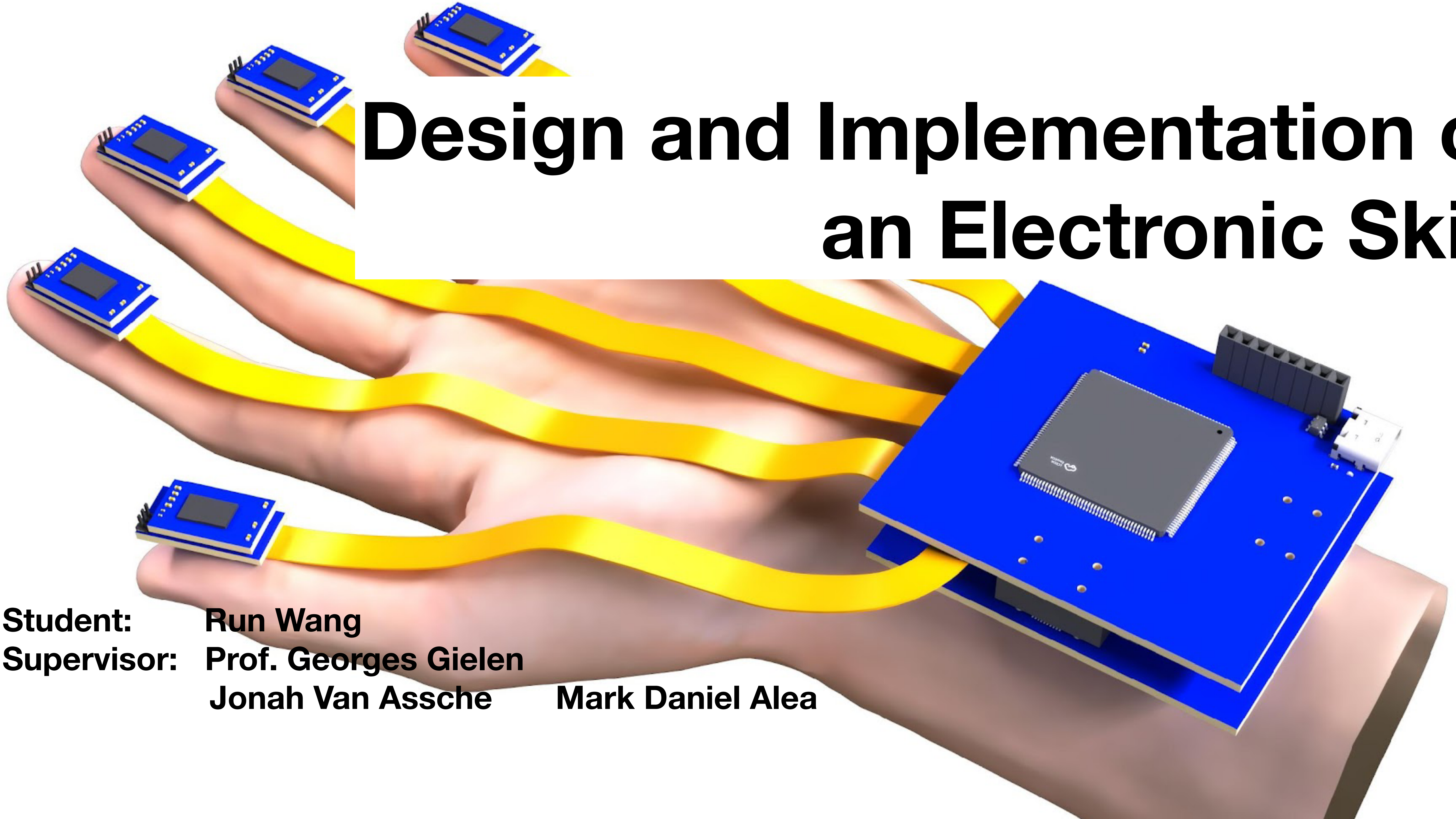
Design and Implementation of an Electronic Skin

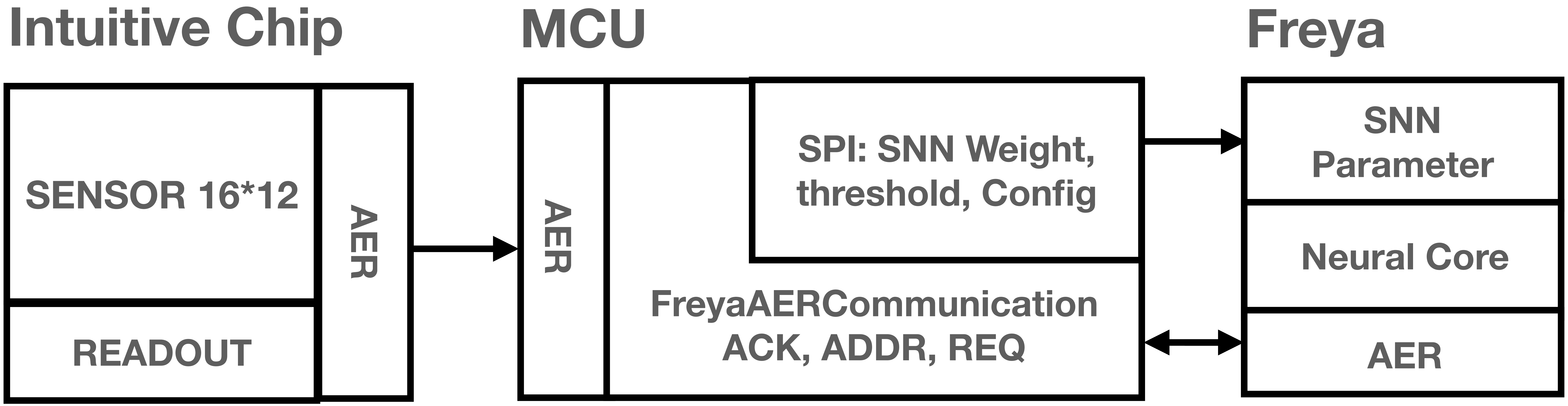


Student: Run Wang

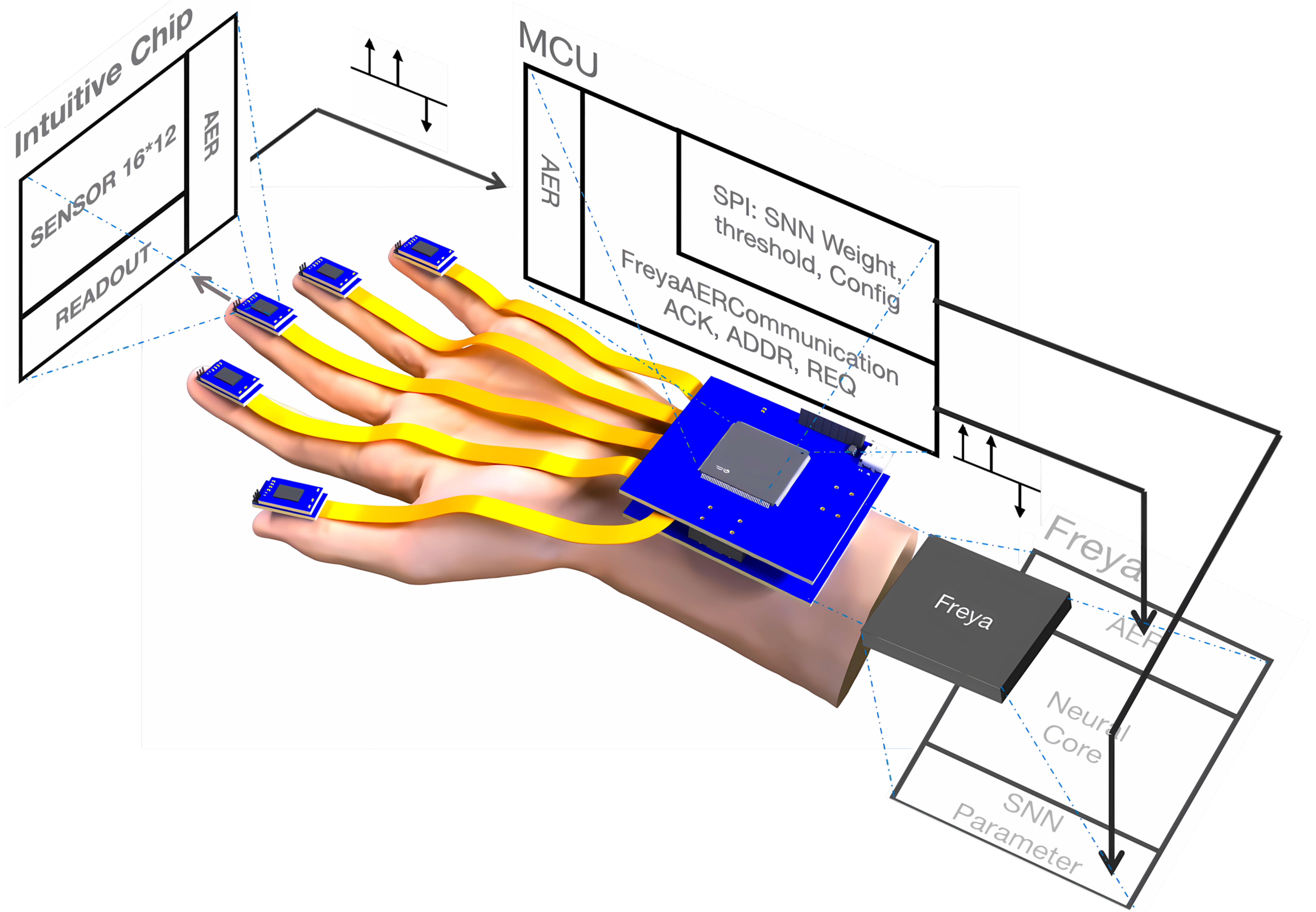
Supervisor: Prof. Georges Gielen

Jonah Van Assche Mark Daniel Alea

System Overview



Intuitive Chip
MCU
Freya
SENSOR 16*12
READOUT
AER
AER
SPI: SNN Weight, threshold, Config
FreyaAERCommunication
ACK, ADDR, REQ
SNN Parameter
Neural Core
AER

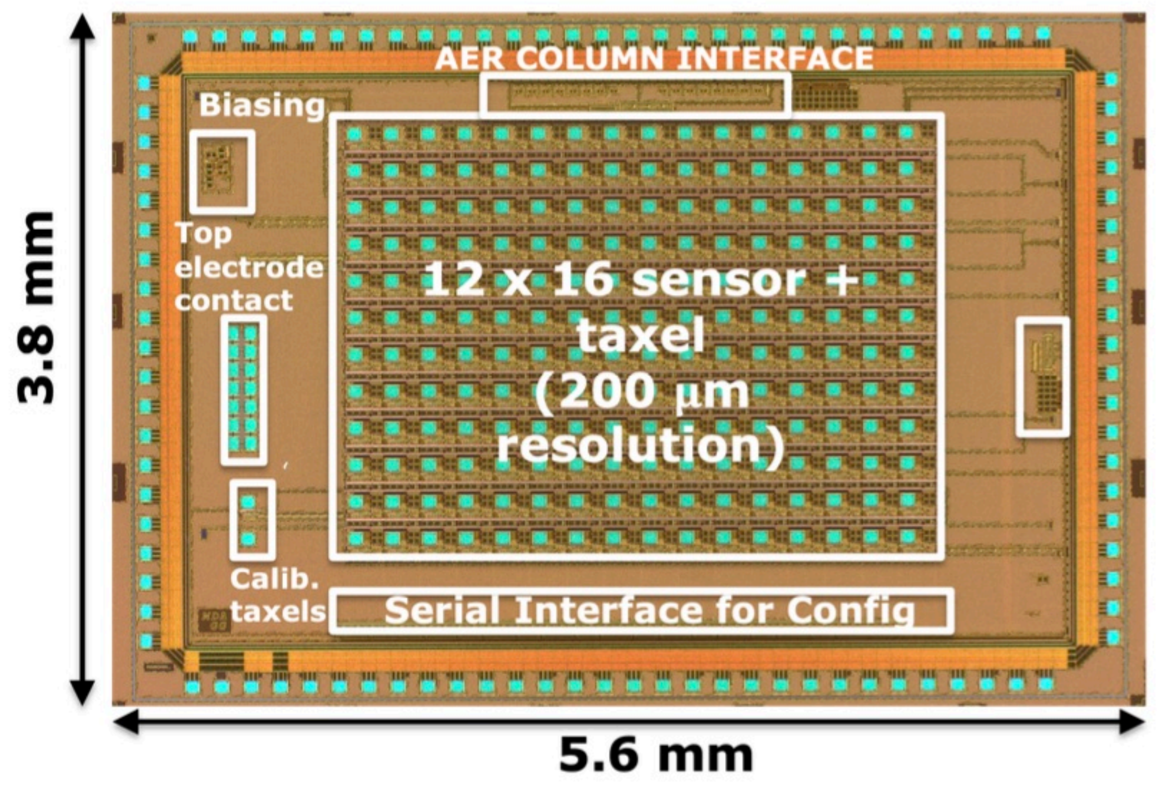


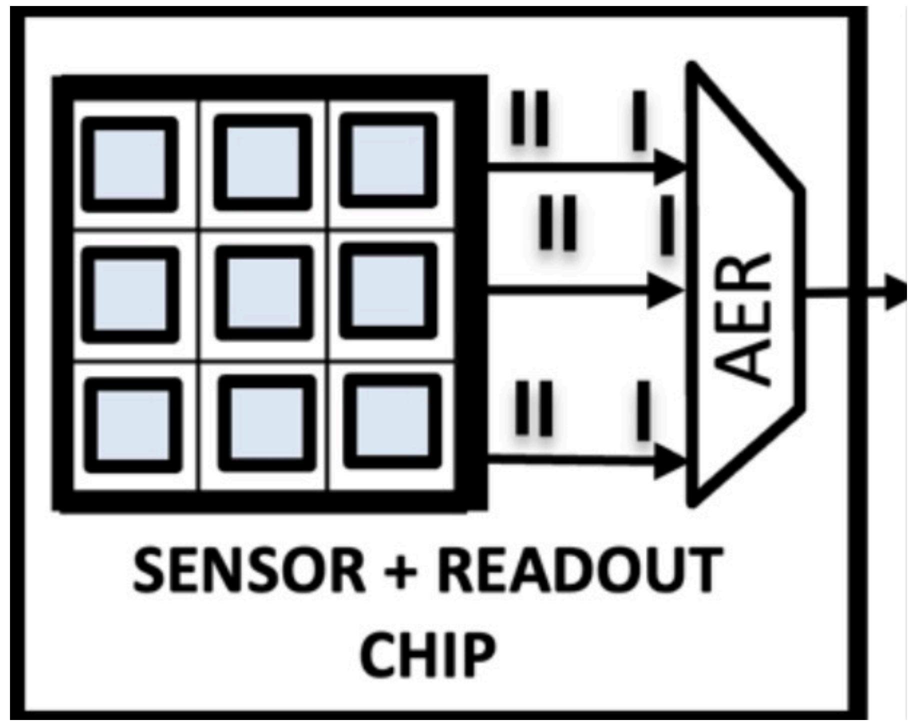
Intuitive Chip
SENSOR 16*12
READOUT
AER
MCU
AER
SPI: SNN Weight, threshold, Config
FreyaAERCommunication
ACK, ADDR, REQ
Freya
Freya
AER
Neural Core
SNN Parameter

Introduction

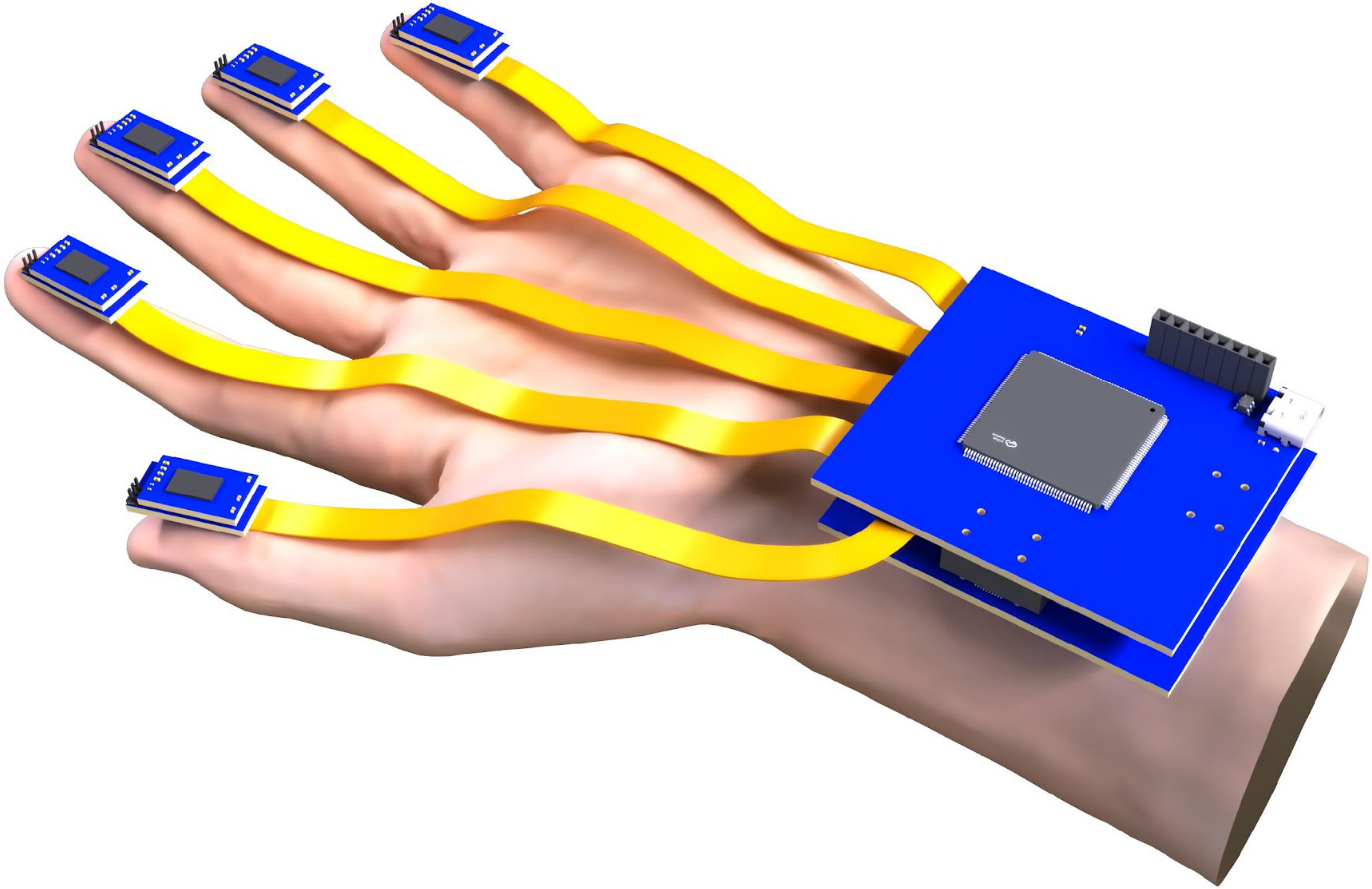
E-skin Taxel Readout Neuromorphic Chip

- Electronic skin taxel readout chip in 0.18um CMOS technology
- 200 um spatial resolution
- 12*16 taxel array
- Spike based readout design





Hardware

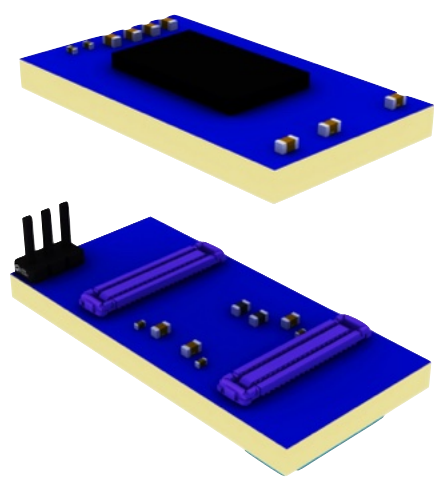


Sensor Process

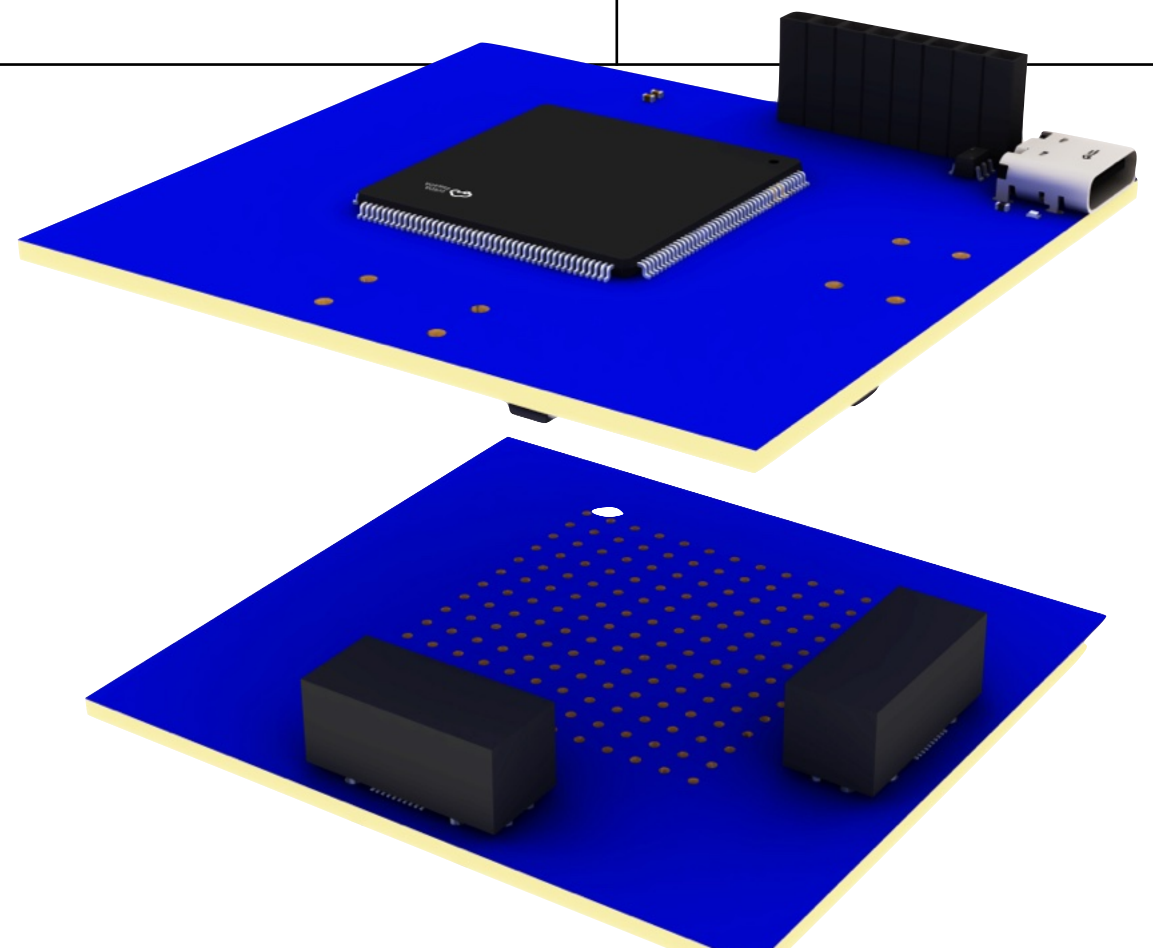
- Z-milling
- Encapsulation
- Poling: 25V sinusoid signal

Fabrication

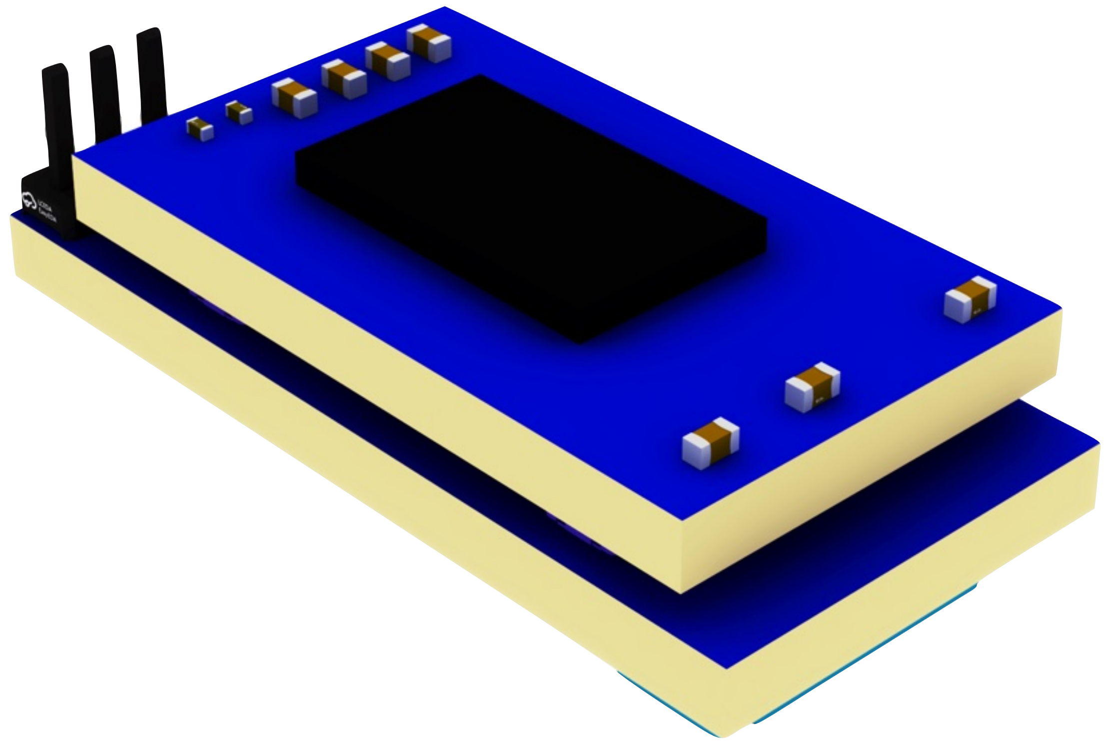
	Size	Layer
Sensor Board	18mm*10mm	4
Finger Board	20mm*10mm	4
MCU Board	64mm*59mm	6
Freya Board	64mm*59mm	2
Middle Board	53mm*44mm	2

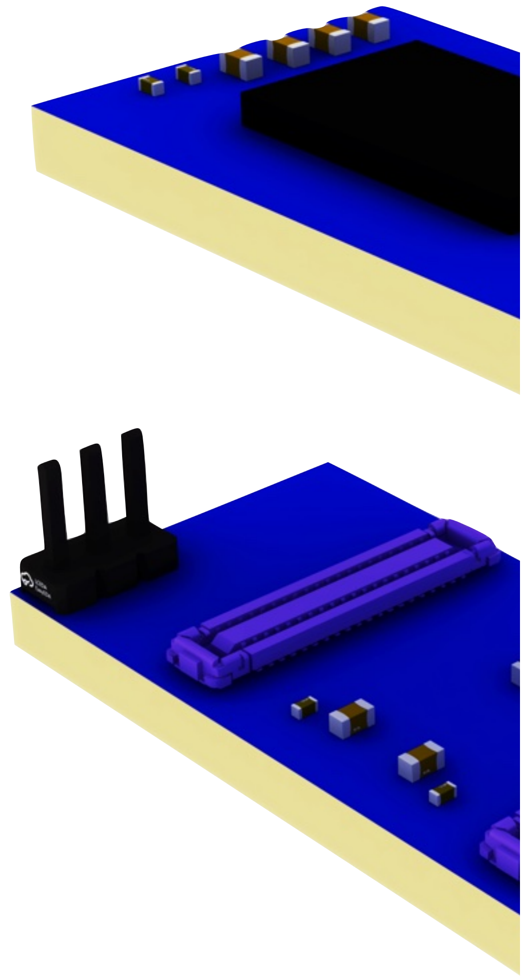


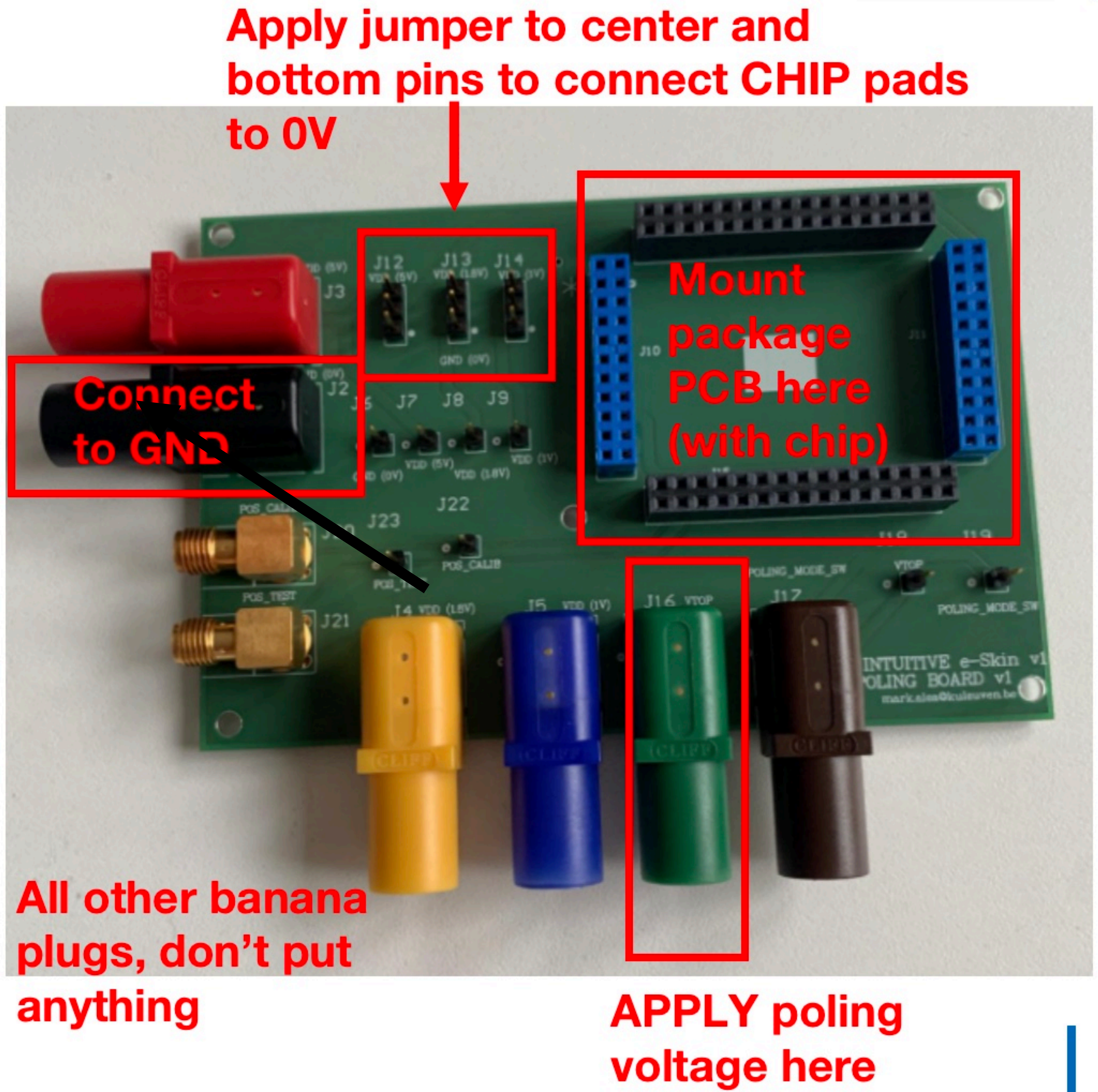
Sensor Boar
Finger Board

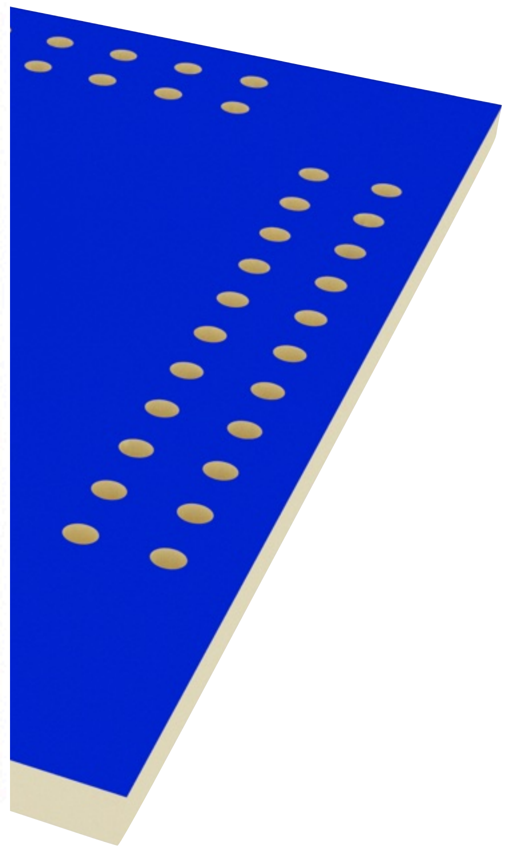


MCU Boar
Freya Board

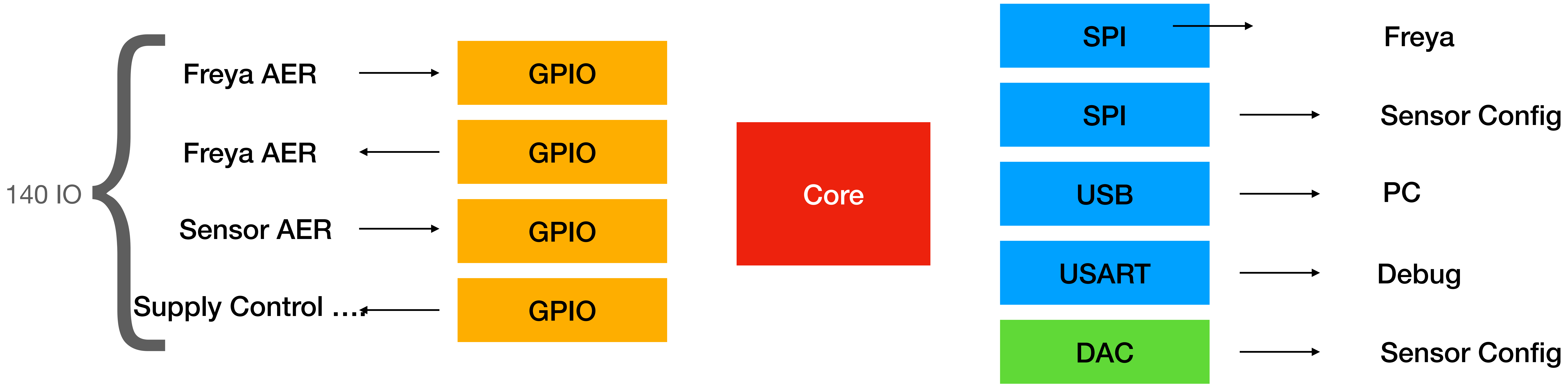




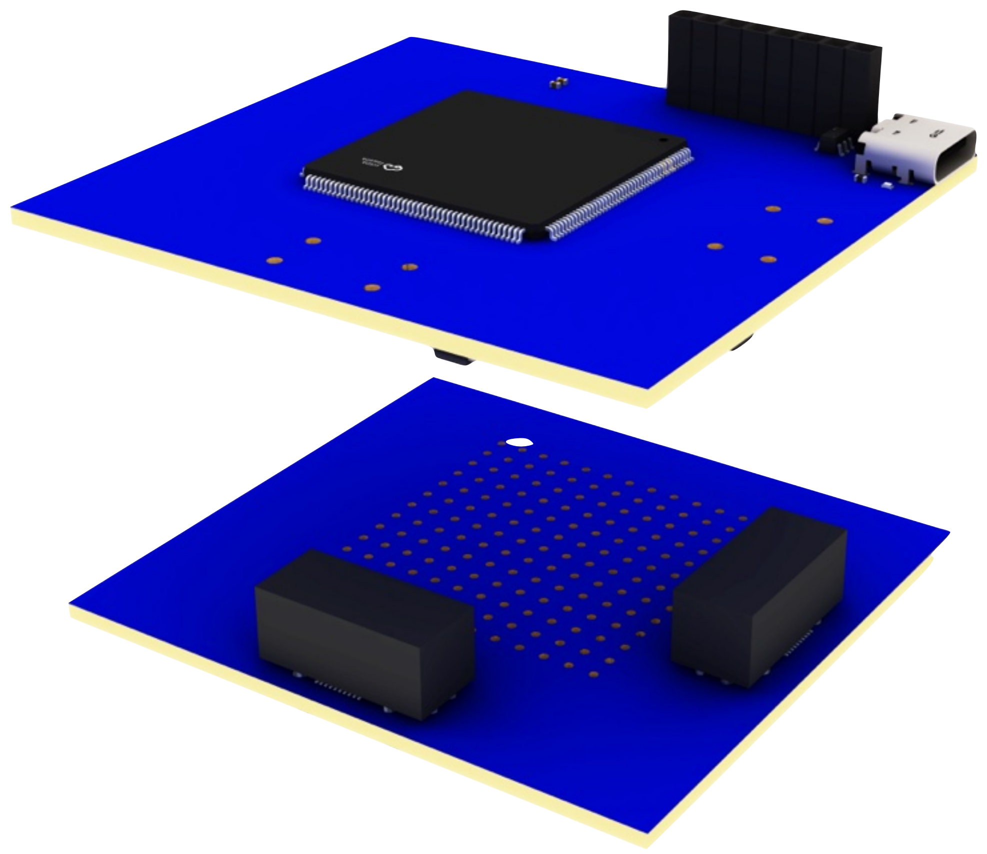

Apply jumper to center and bottom pins to connect CHIP pads to 0V
Connect to GND
Mount package PCB here (with chip)
All other banana plugs, don't put anything
APPLY poling voltage here

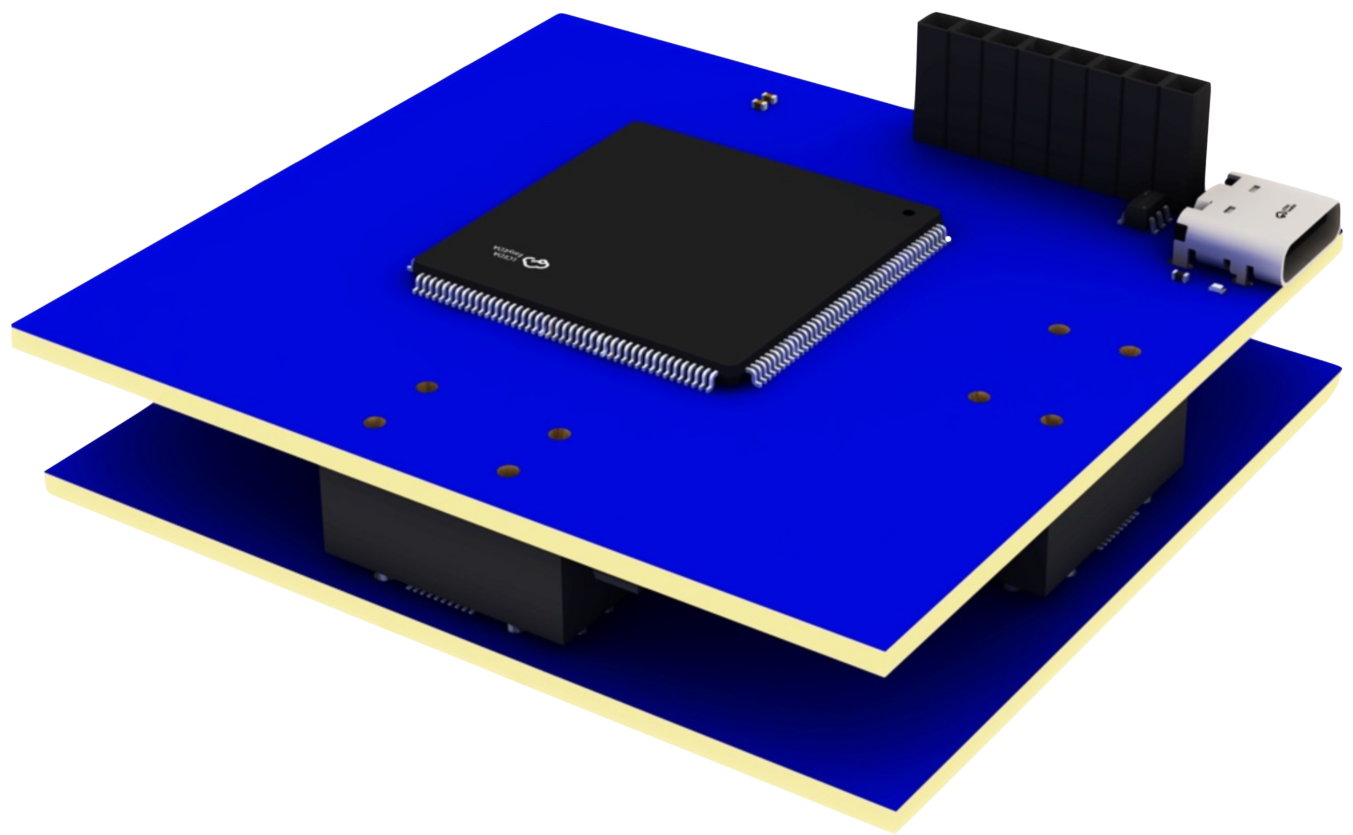


MCU Board



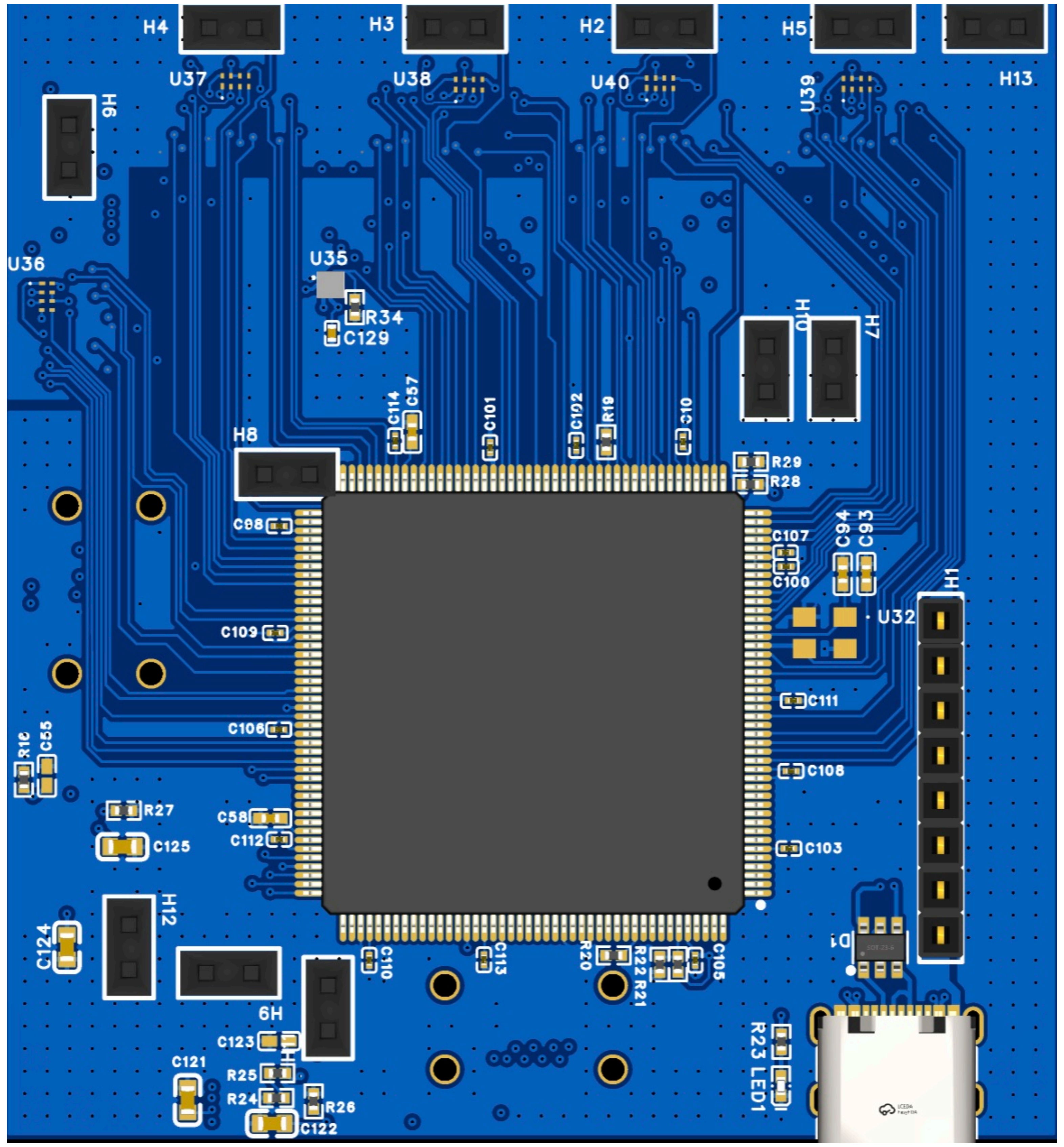
140 IO
Freya AER
Freya AER
Sensor AER
Supply Control ....
GPIO
GPIO
GPIO
GPIO
Core
SPI
SPI
USB
USART
DAC
Freya
Sensor Config
PC
Debug
Sensor Config

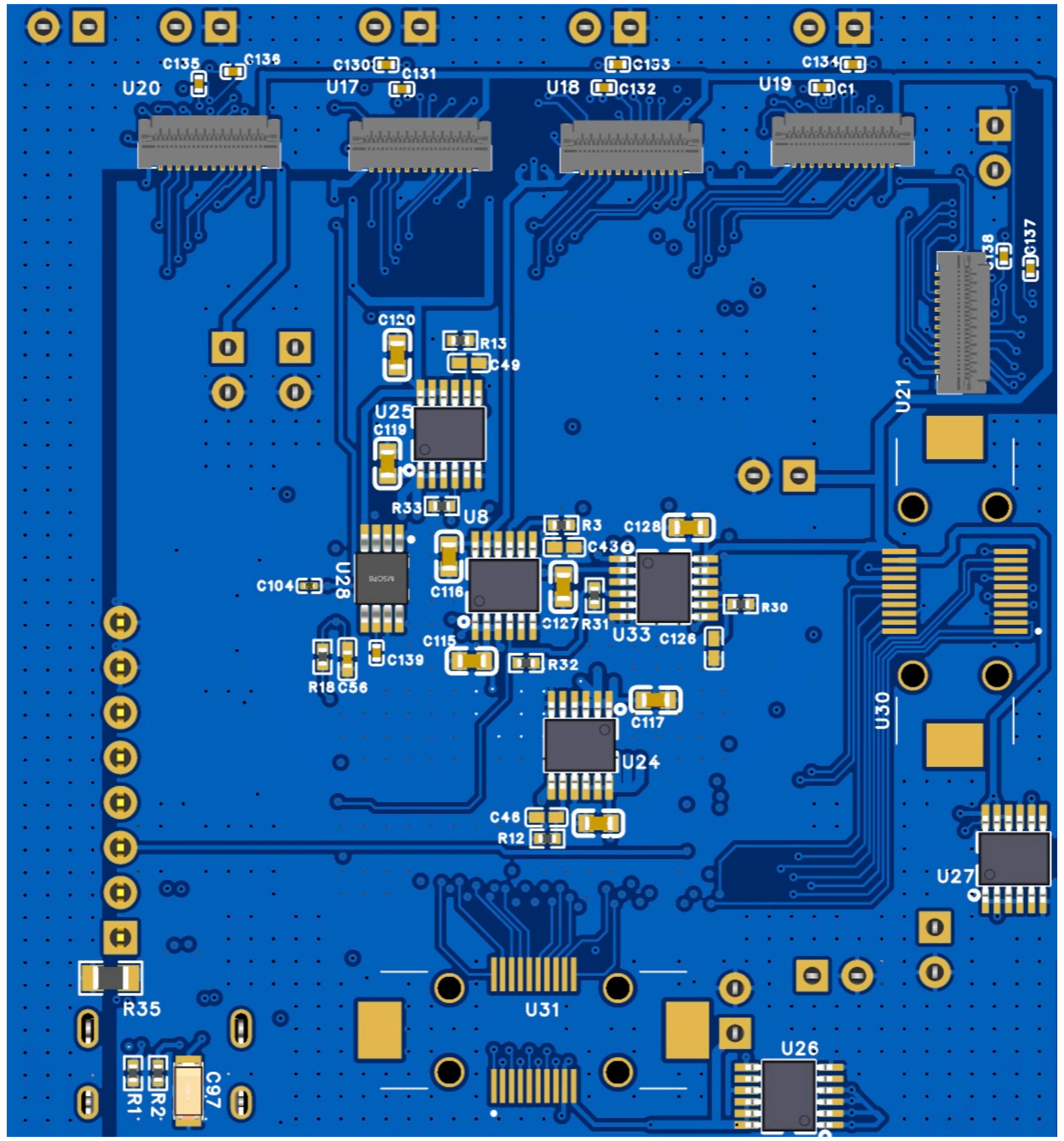




MCU Board Design

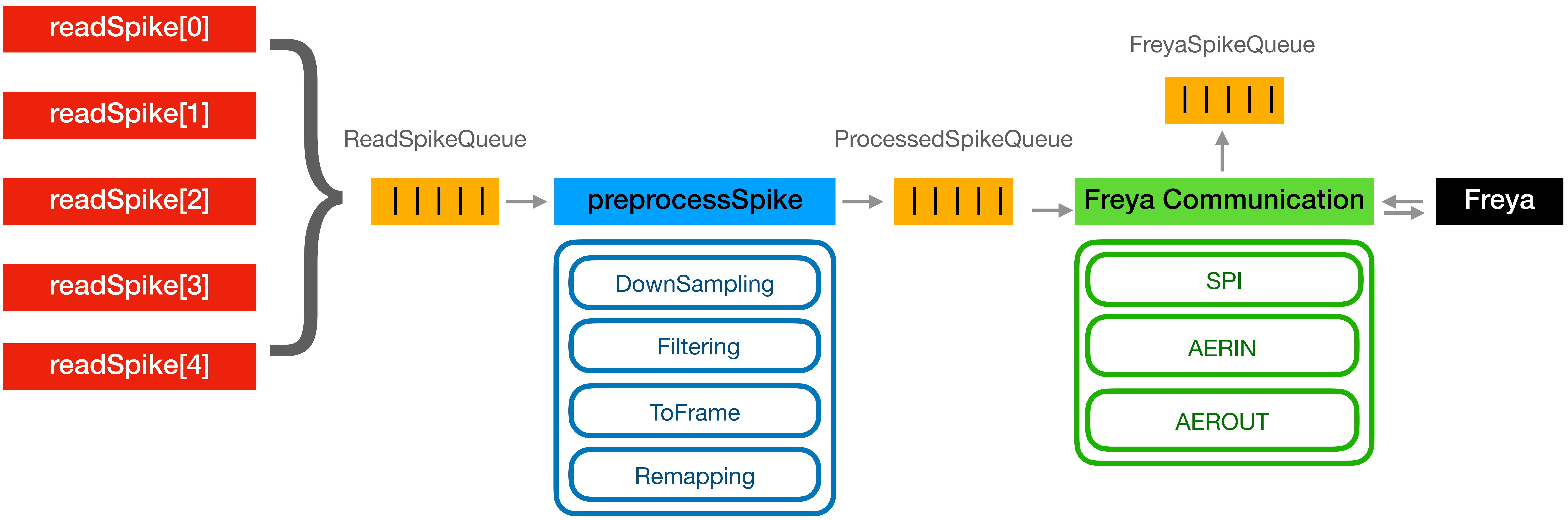
- Supply sequence
- Debug
- 3.3V to 5V level shifter
- AGND DGND seperate
- USB data & Supply





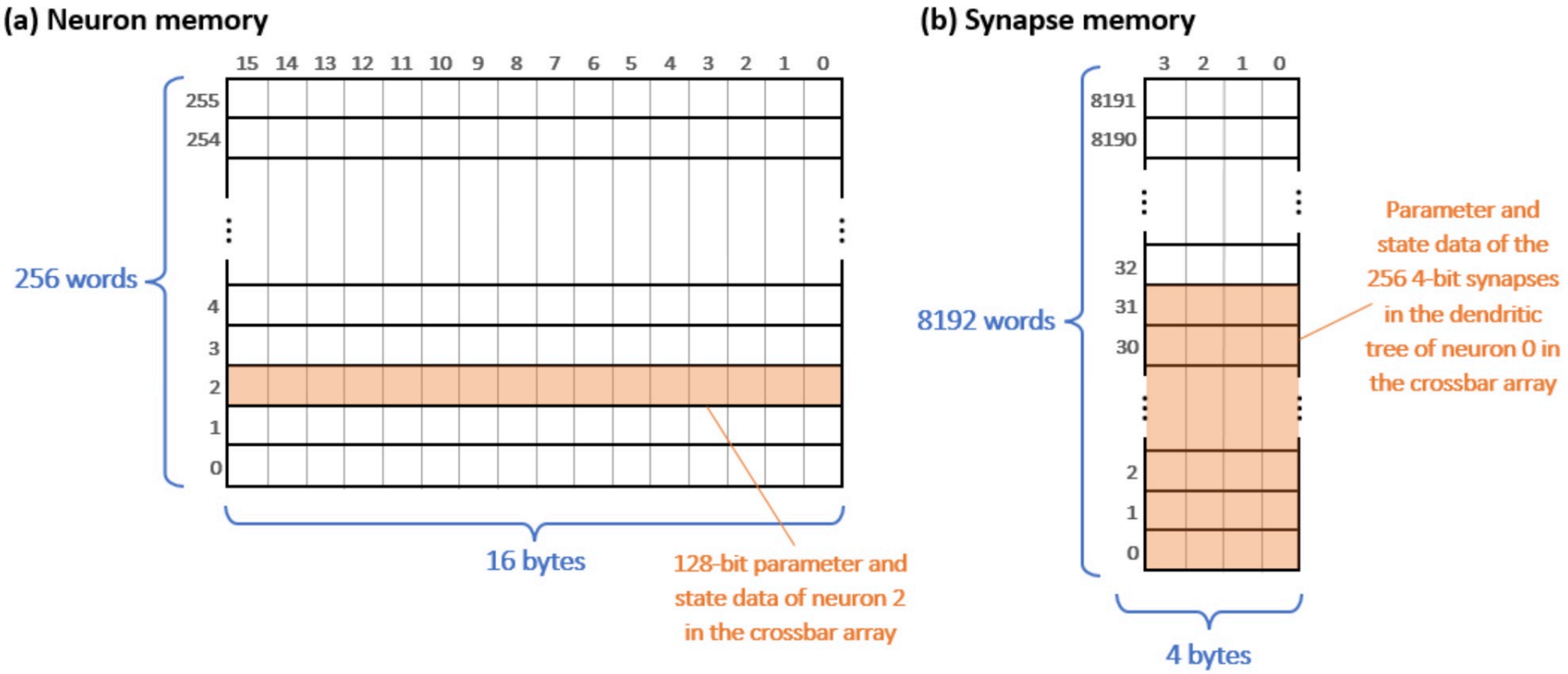
Software

Embedded Software RTOS



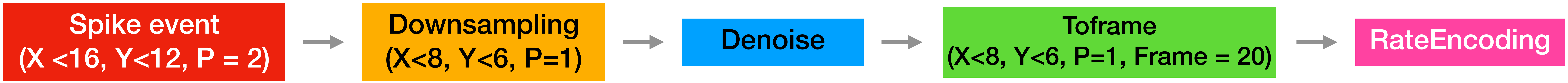
SNN to Freya

- IF (Intergrate and Fire Model)
- Signed 3bit integer for weight, 8bit for threshold, 8bit for h_mem
- SPI send weight for synapse SRAM
- Neuron spike event: Stimulates all neurons with the synaptic weight associated to pre-synaptic neuron *pre_neur<7:0>*.
- Neuron 256: input neuron+hidden neuron+output neuron<=256


(a) Neuron memory
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
255
254
4
3
2
1
0
256 words
16 bytes
128-bit parameter and state data of neuron 2 in the crossbar array
(b) Synapse memory
3 2 1 0
8191
8190
32
31
30
2
1
0
8192 words
4 bytes
Parameter and state data of the 256 4-bit synapses in the dendritic tree of neuron 0 in the crossbar array

Spike Preprocessing

- Neuron<=256 -> Downsampling
- Noisy -> Denoise
- Freya input 0/1 -> RateEncoding



Quantization

- Post Training Quantization

$$\begin{cases} V^{k+1} = V^k + J_{\text{in}} \text{ and } S = 0, & \text{if } V^k < 1 \\ V^{k+1} = 0 \text{ and } S = 1, & \text{if } V^k \geq 1. \end{cases}$$

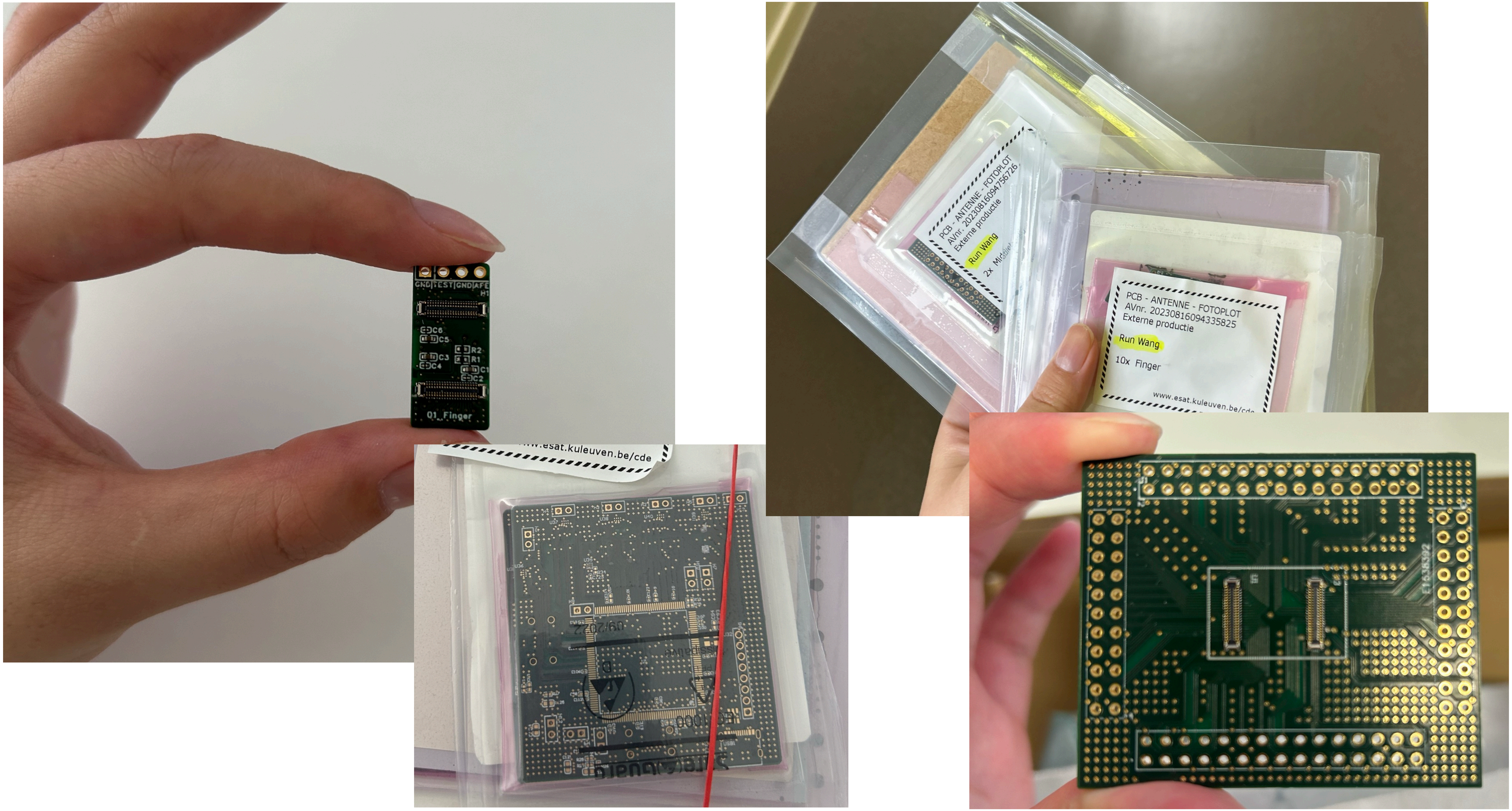
↓

$$\sum_{k=n}^{m} w_{i2h}^{(k)} \cdot J_{in}^{(k)} + w_{h2h} < \text{threshold}_{\text{h}}$$

n: Represents the time for any spike
m: Represents the next time spike after n

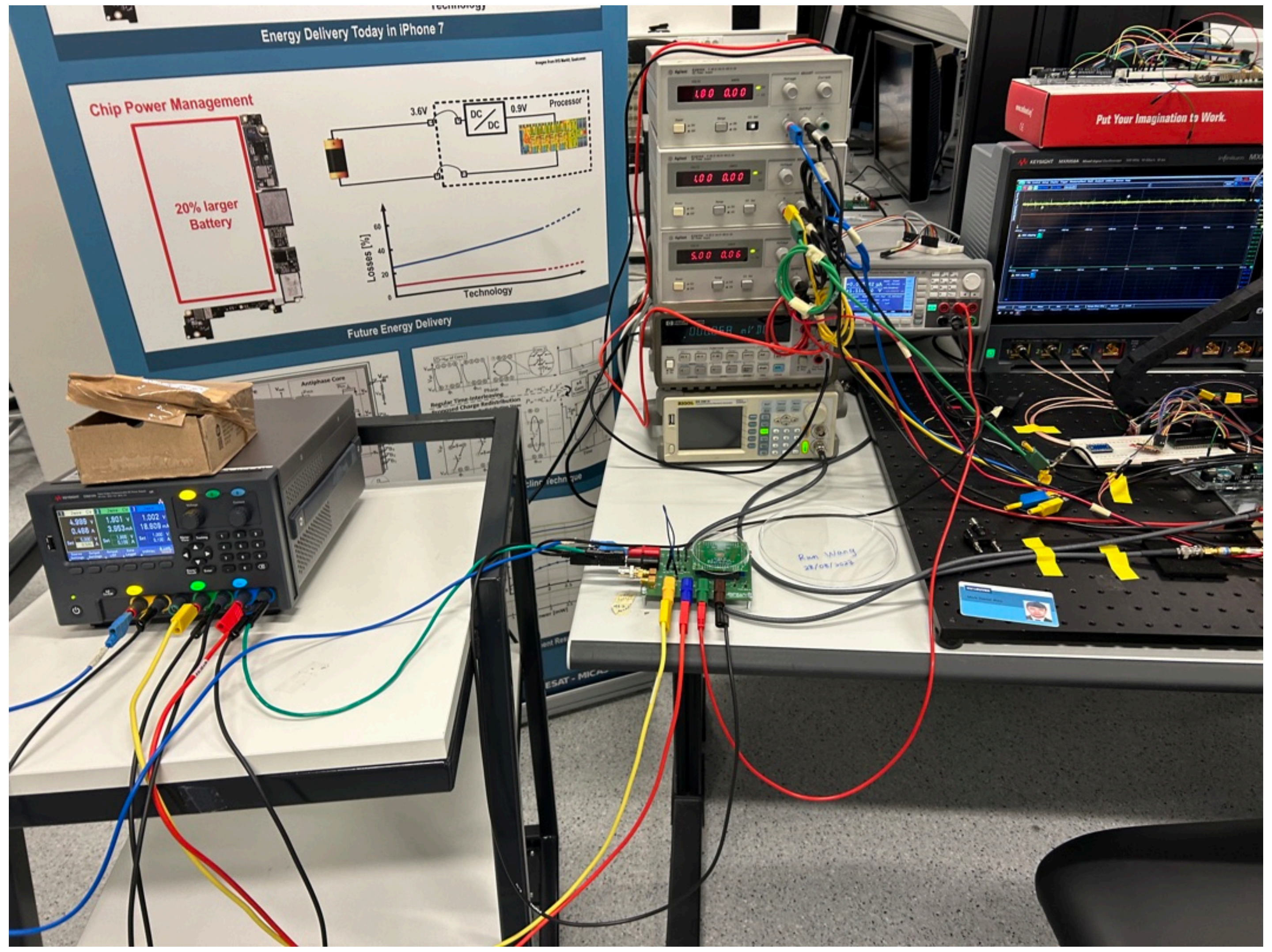
- Increase i2h, h2h, thr_h by same factor
- Increase h2o, thr_o by same factor

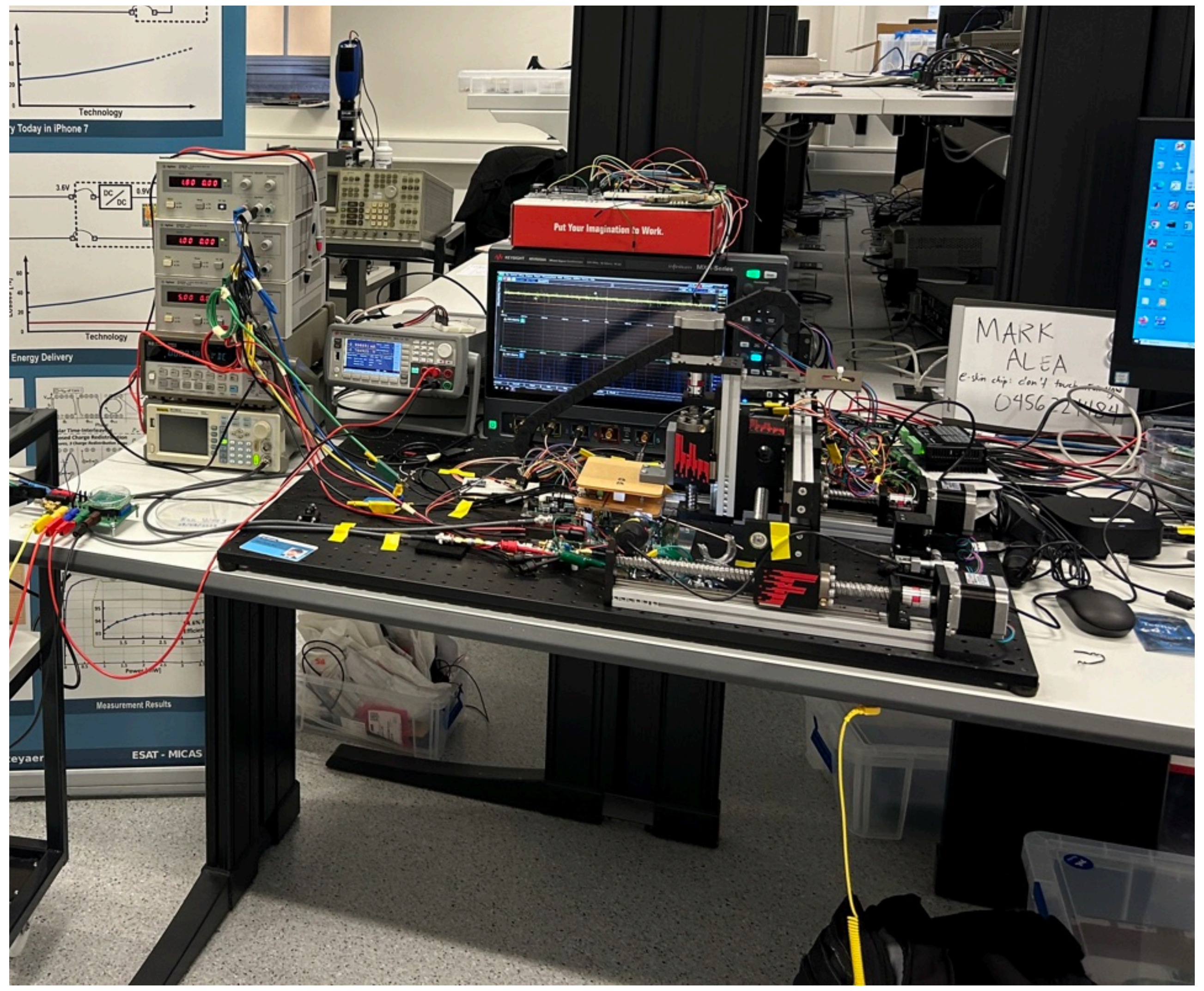
Result

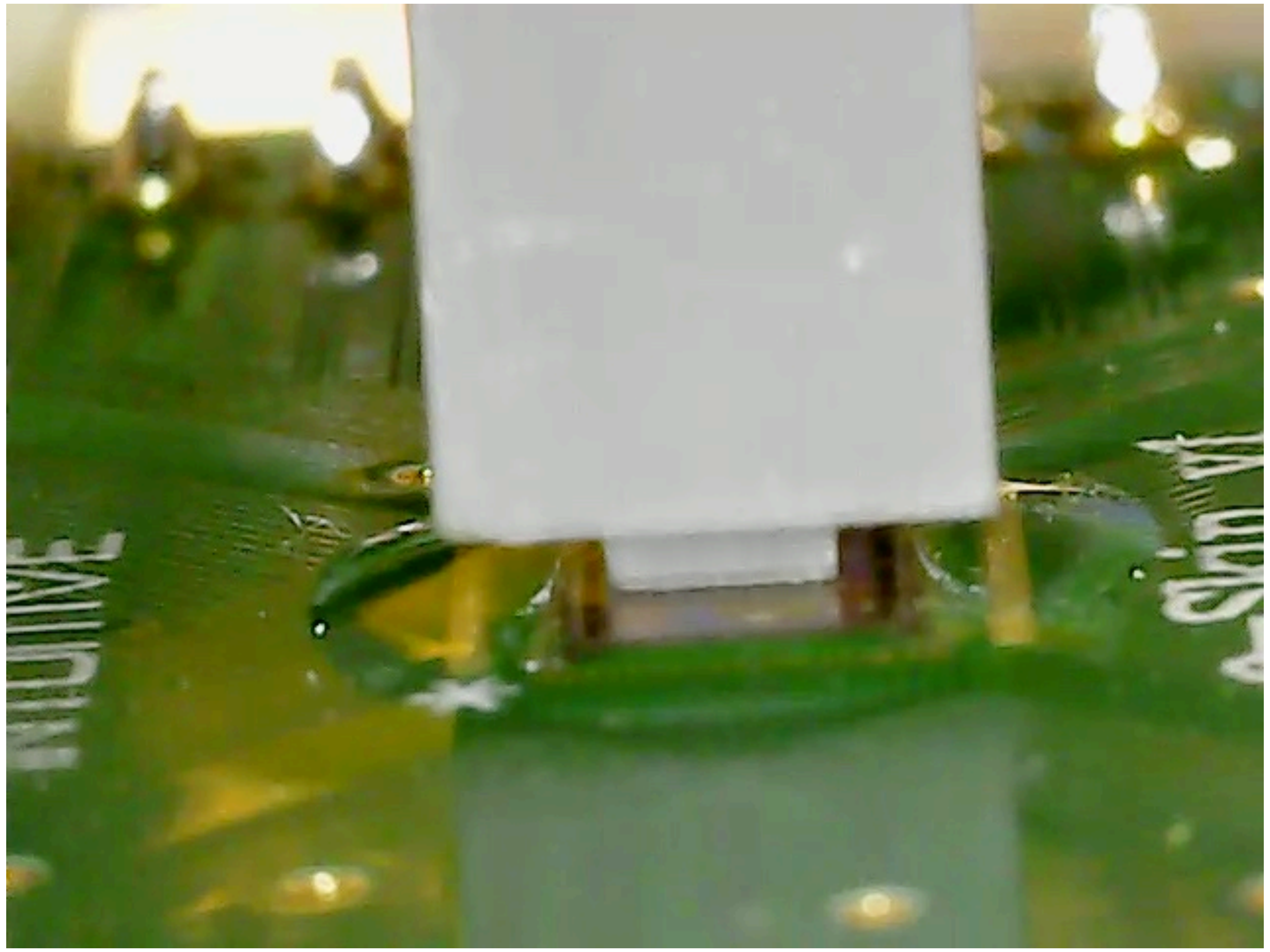

GND TEST GND AFE
01_Finger
PCB - ANTENNE - FOTOPLOT
AVnr. 20230816094756726
Externe productie
Run Wang
PCB - ANTENNE - FOTOPLOT
AVnr. 20230816094335825
Externe productie
Run Wang
10x Finger
www.esat.kuleuven.be/cde

Dataset

- Sensor PDMS Encapsulation
- Sensor Poling
- Frequency dataset 40 0.5Hz to 10 Hz

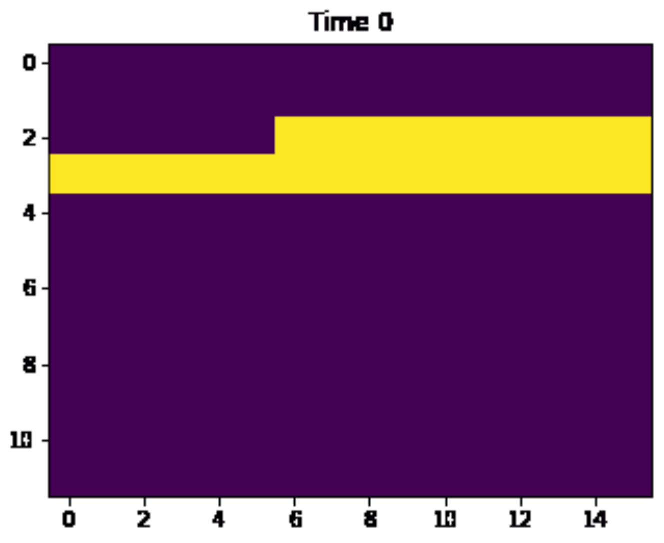


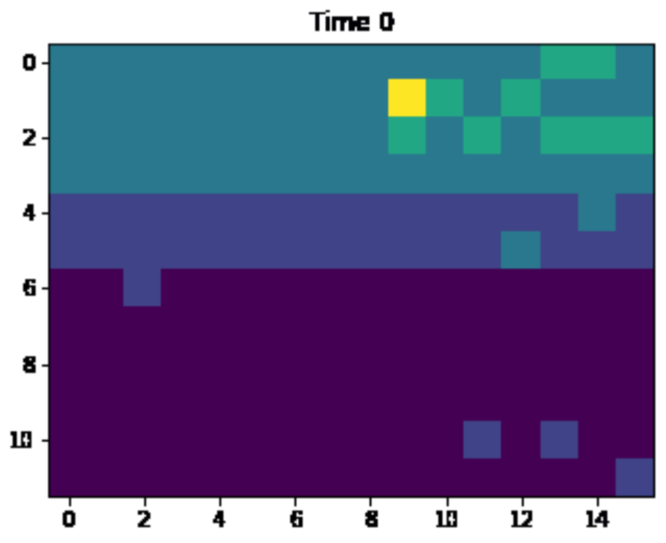


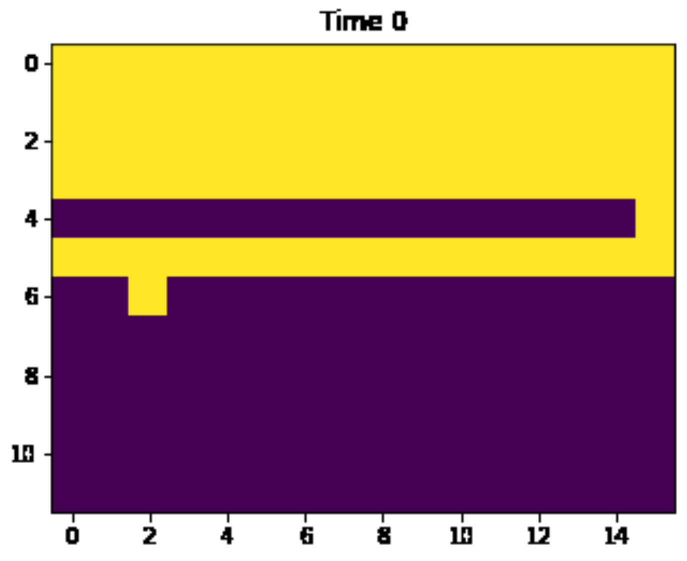


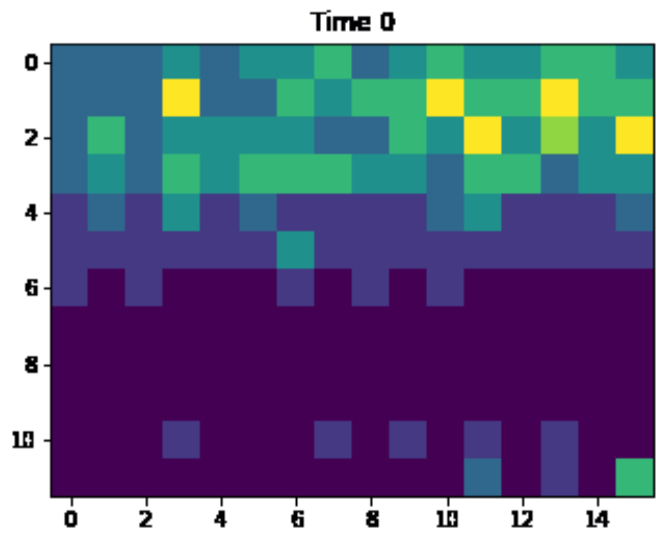
Frame Gif

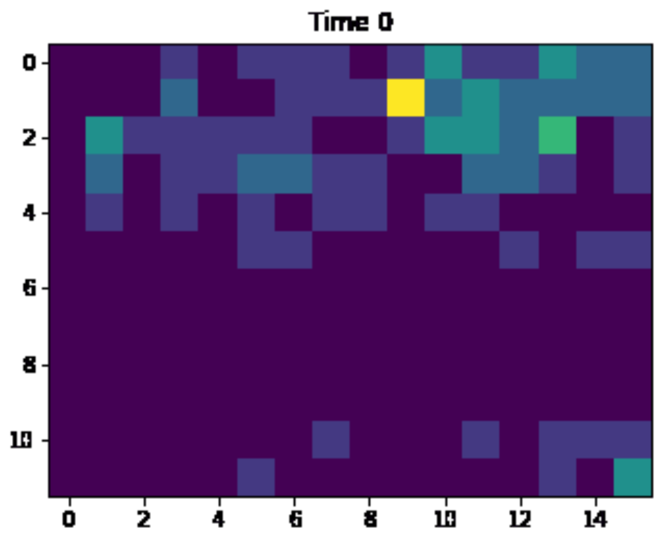
Class 0



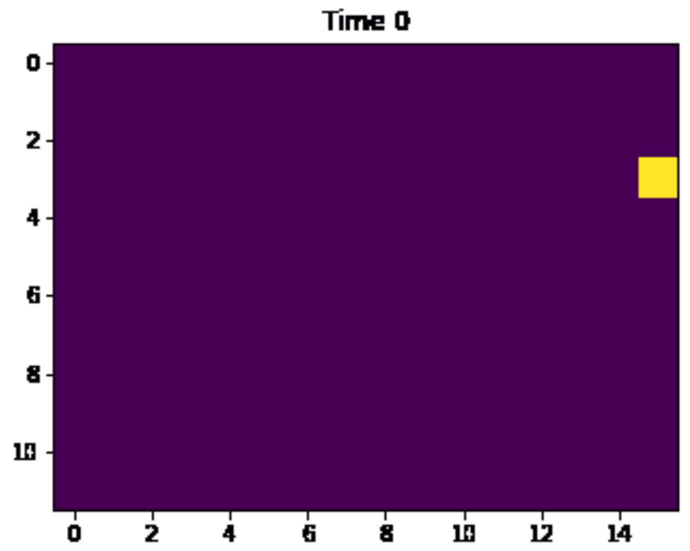


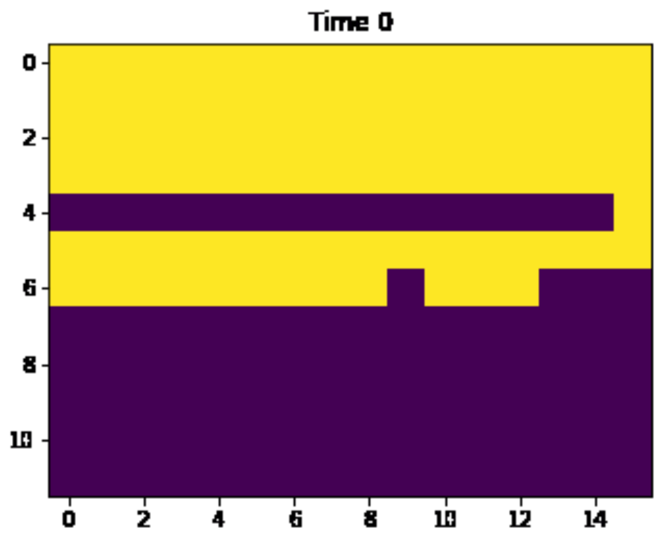


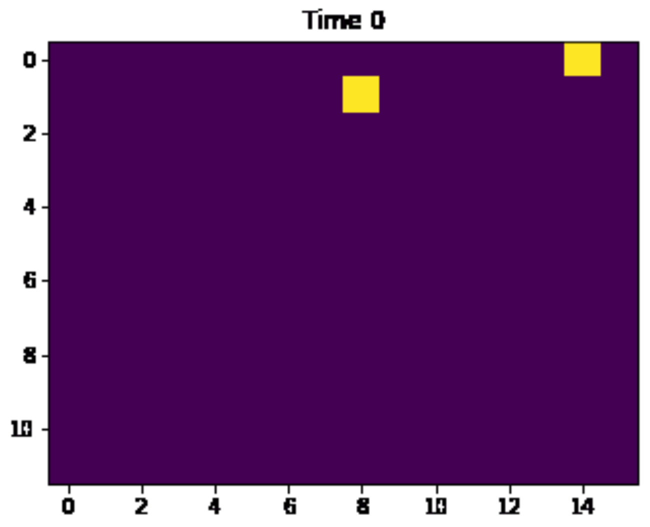


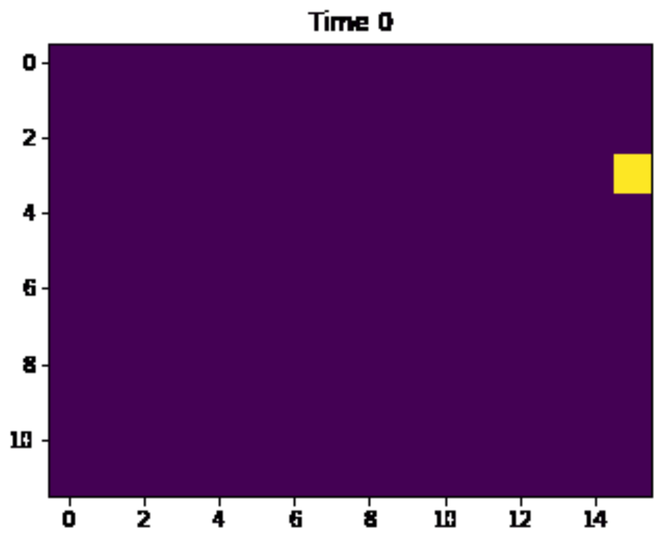


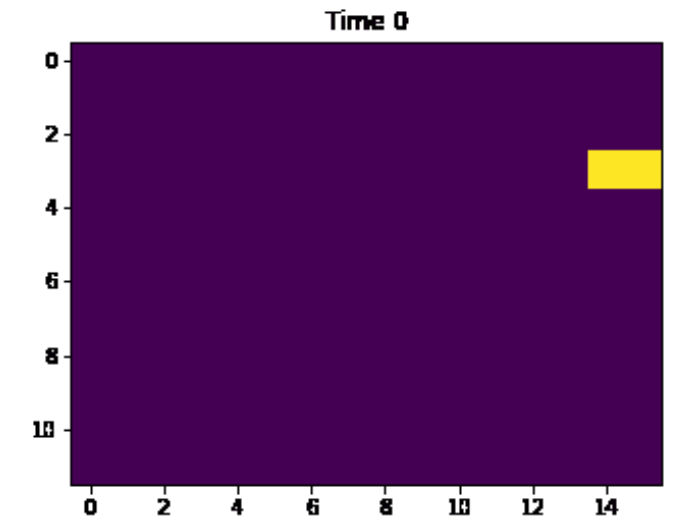
Class 1





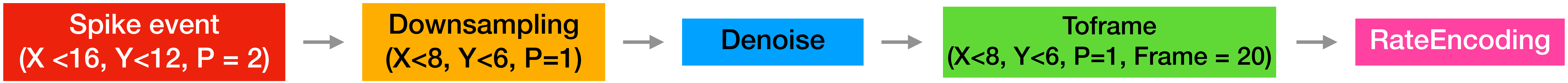






Preprocess flow Result

- 2 class 6 fold
  - IF neuron
  - input 48(6*8), hidden neuron 206
  - >90%



Quantization Result

- Post quantization
  - No rateencoding, most fold drop little
    - [1.0, 0.91,0.83, 0.83, 0.91,0.90] ->[0.91,0.91,0.75,0.91,0.5,0.81]
  - With rate encoding, drop a lot 90%->40%
    - After quantization, bias all zero

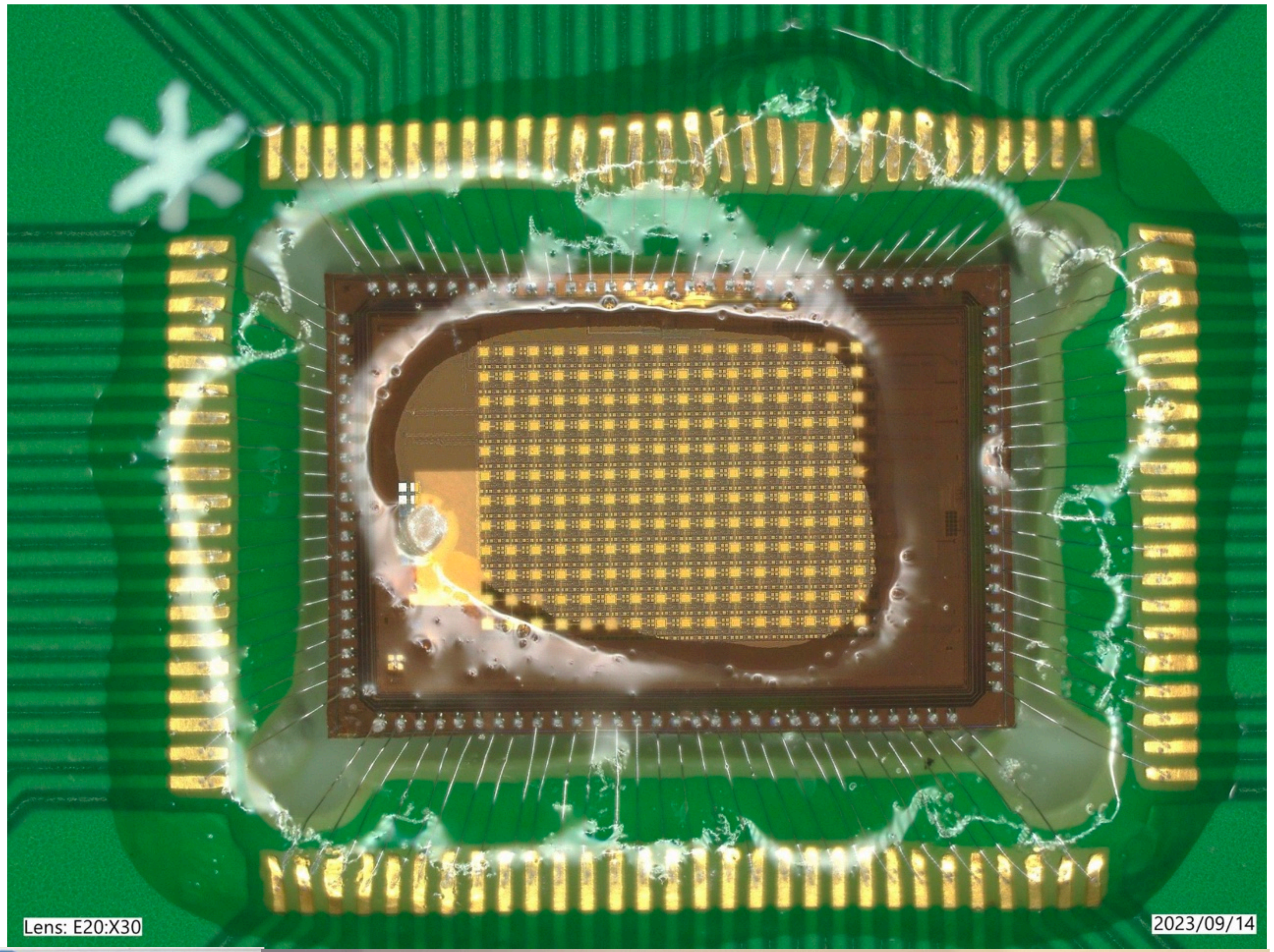
Quantization Discussion

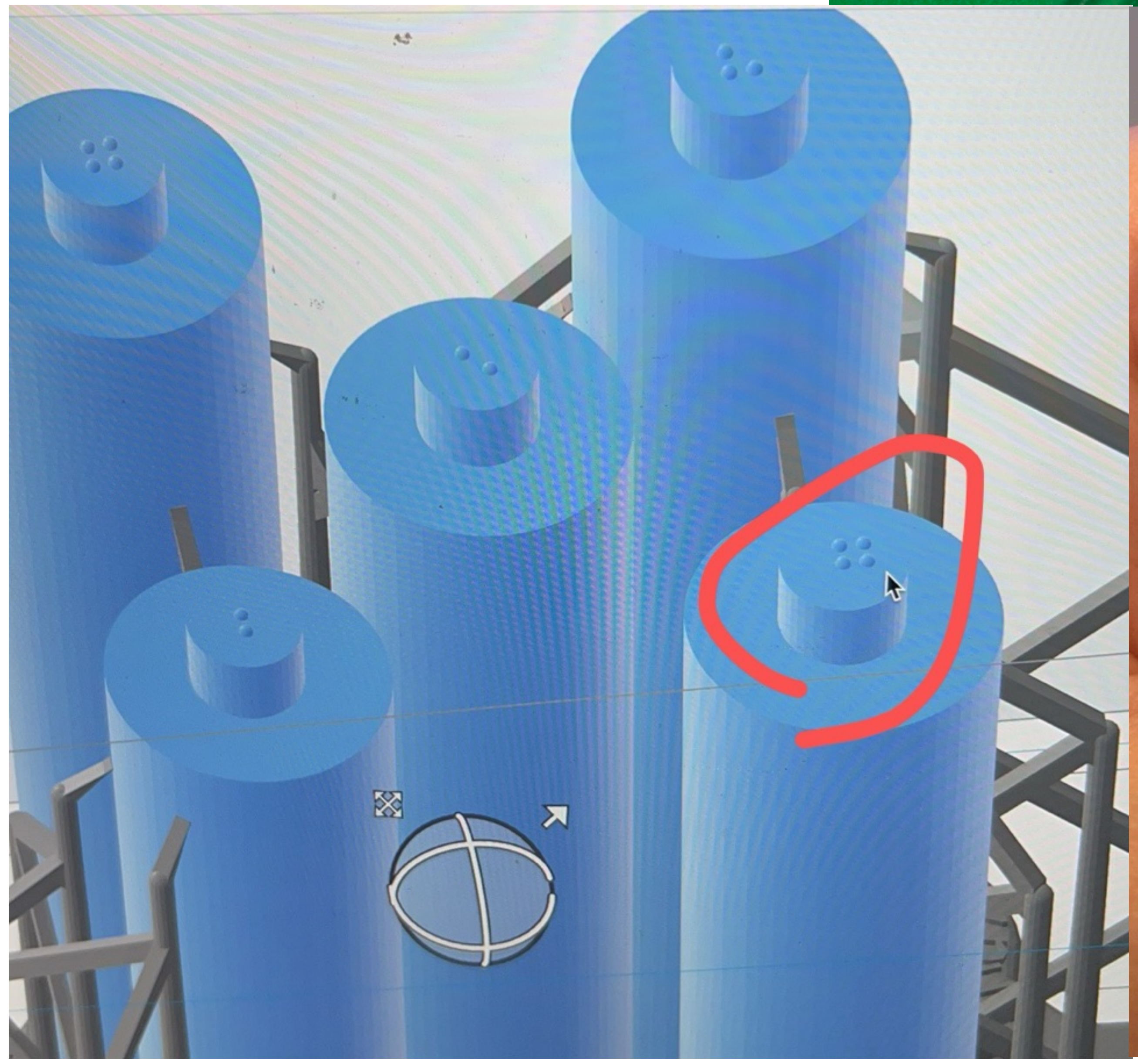
- Post quantization
  - Use statistical tools to remove outliers value
- Cannot implement quantization aware training
- ANN->SNN
- STDP

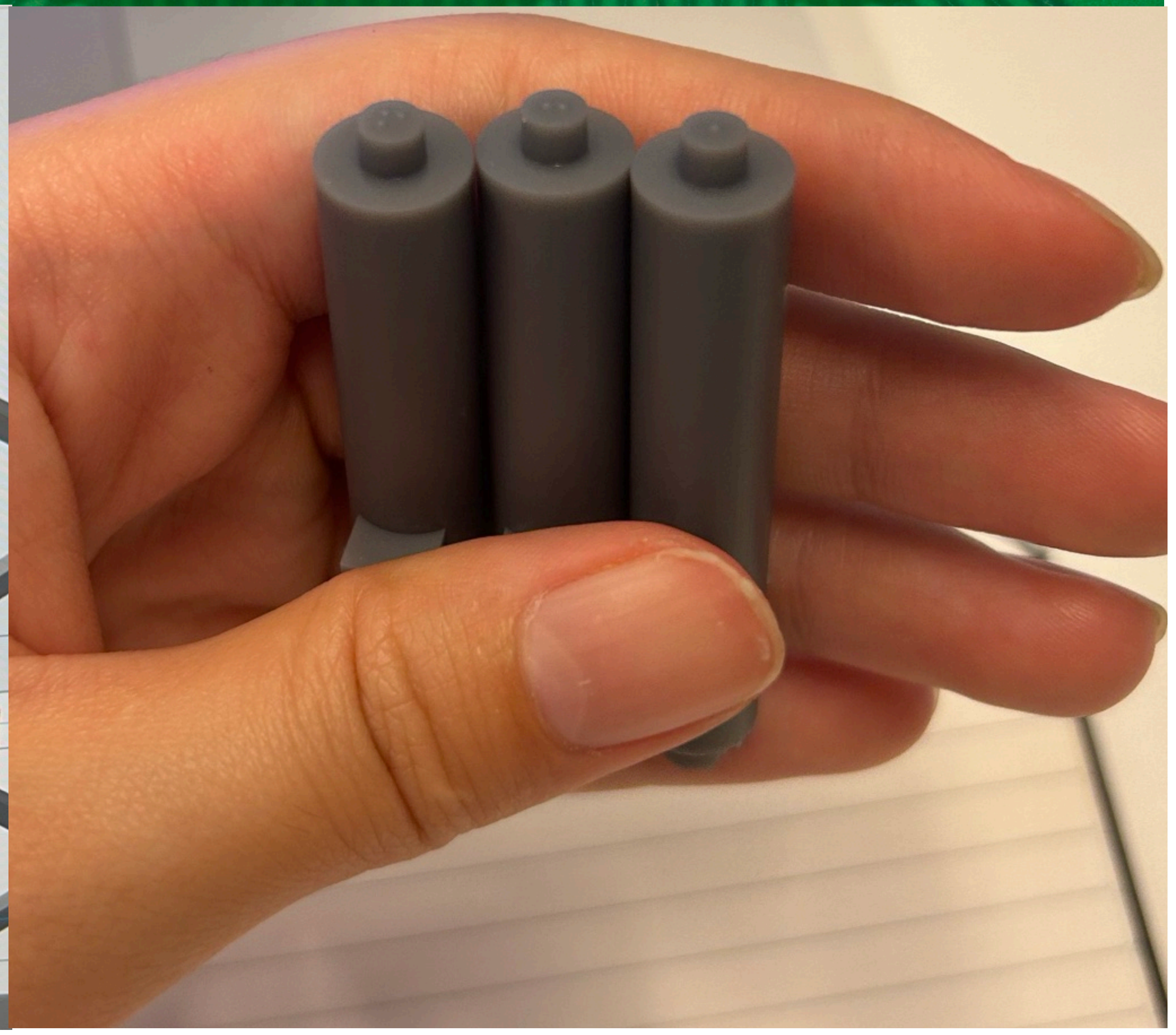
Discussion

Future Work

- Encapsulation Method Upgrade
- 3D model new task
- Hardware testing
- Embedded system implementation







Contribution

- Hardware design and in charge of production
- Software Freya Compatibility Design
- Encapsulation Method design
- 3D model design for future work