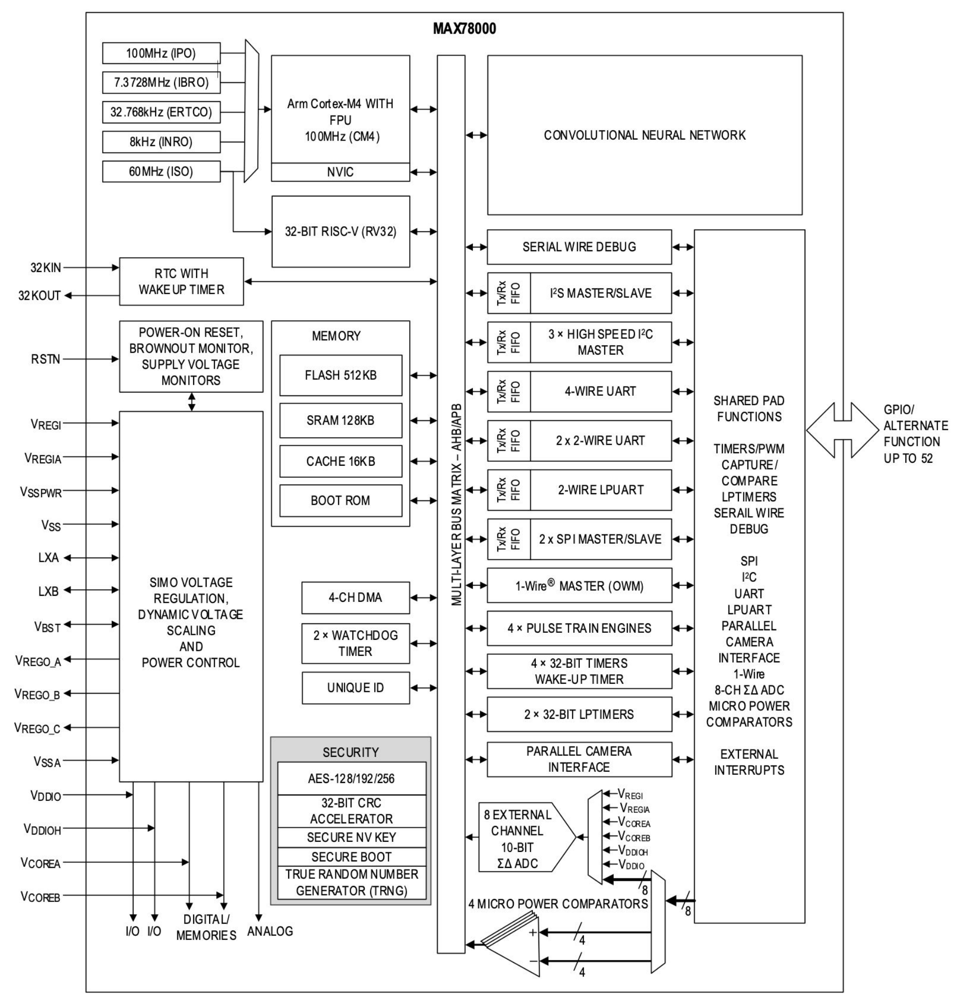
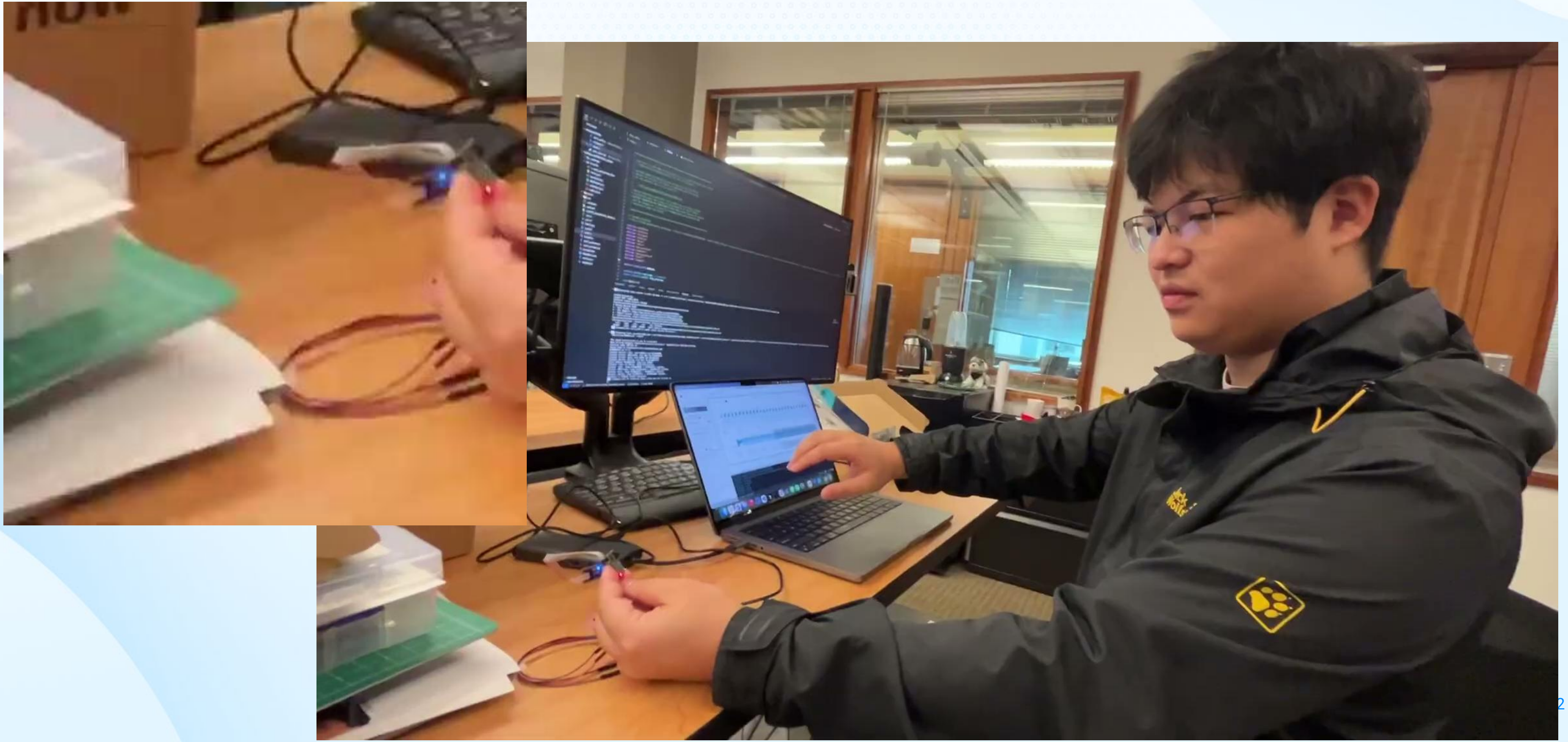
MAX78000

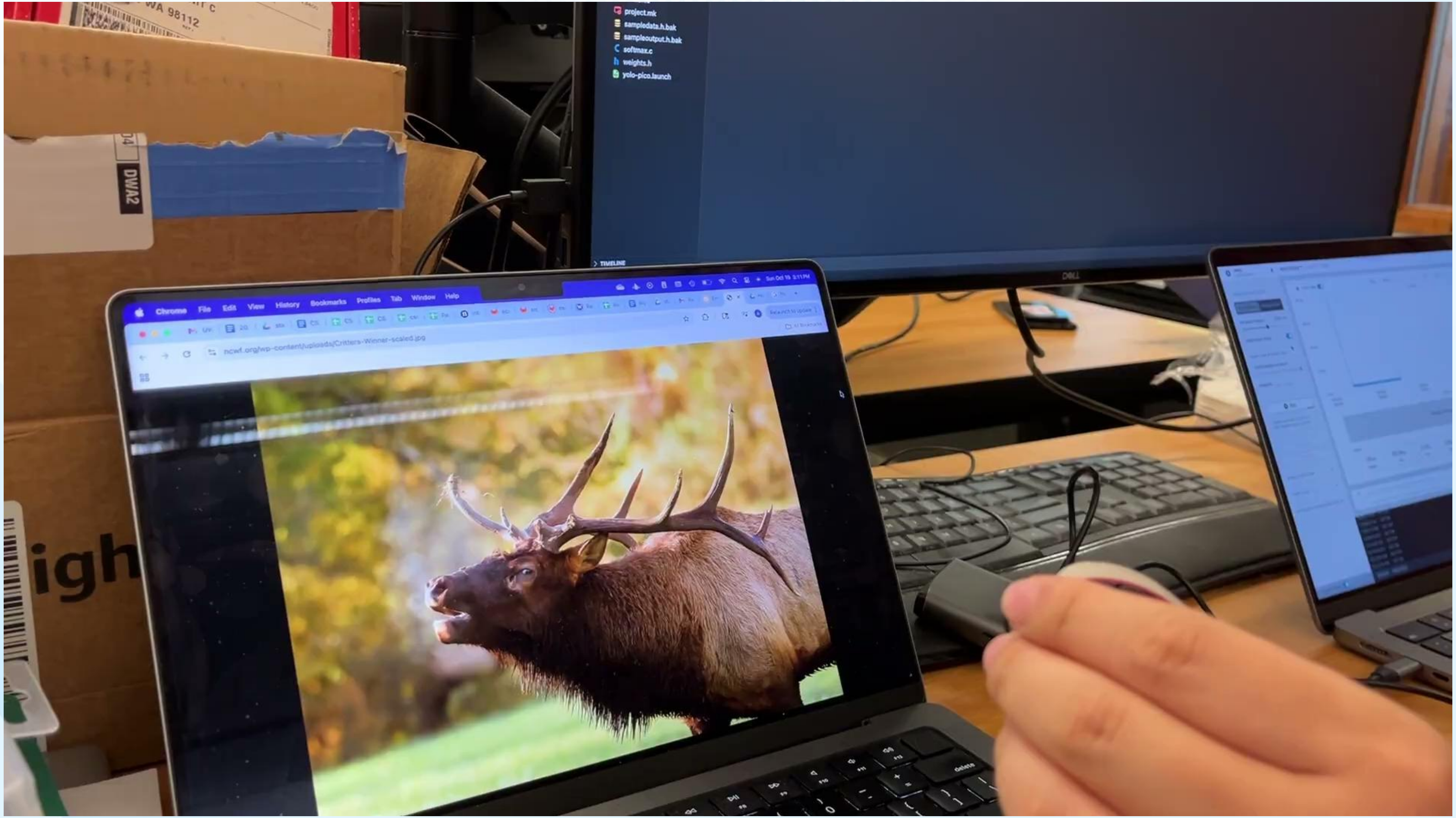
- $20 Low power MCU (datasheet)
- CNN accelerator (64 cores)
- Dual core
  - Arm Cortex-M4 Core
  - Low power RISC-V Core
- Oscillator Freq: 100Mhz - 8kHz
- Can be batteryless edge node with CNN capability with energy harvesting module.



Demo (Face Detection)

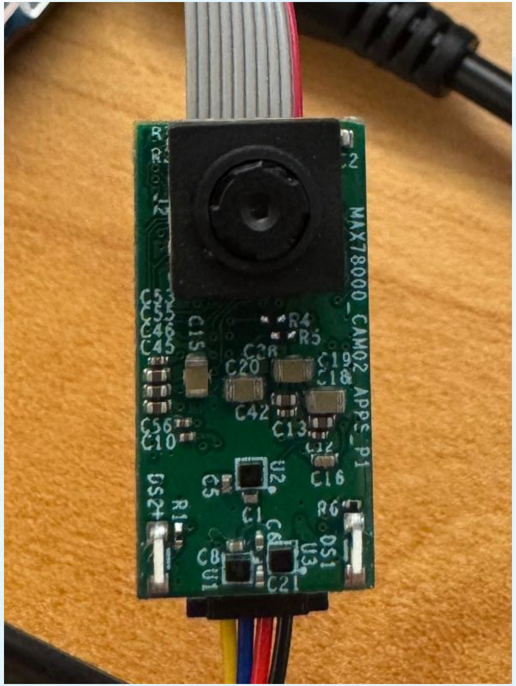


Demo (Elk Detection)


project.mk
sampledata.h.bak
sampleoutput.h.bak
softmax.c
weights.h
yolo-pico.launch
TIMELINE
Chrome File Edit View History Bookmarks Profiles Tab Window Help
ncwf.org/wp-content/uploads/Critters-Winner-scaled.jpg
DELL
DWA2

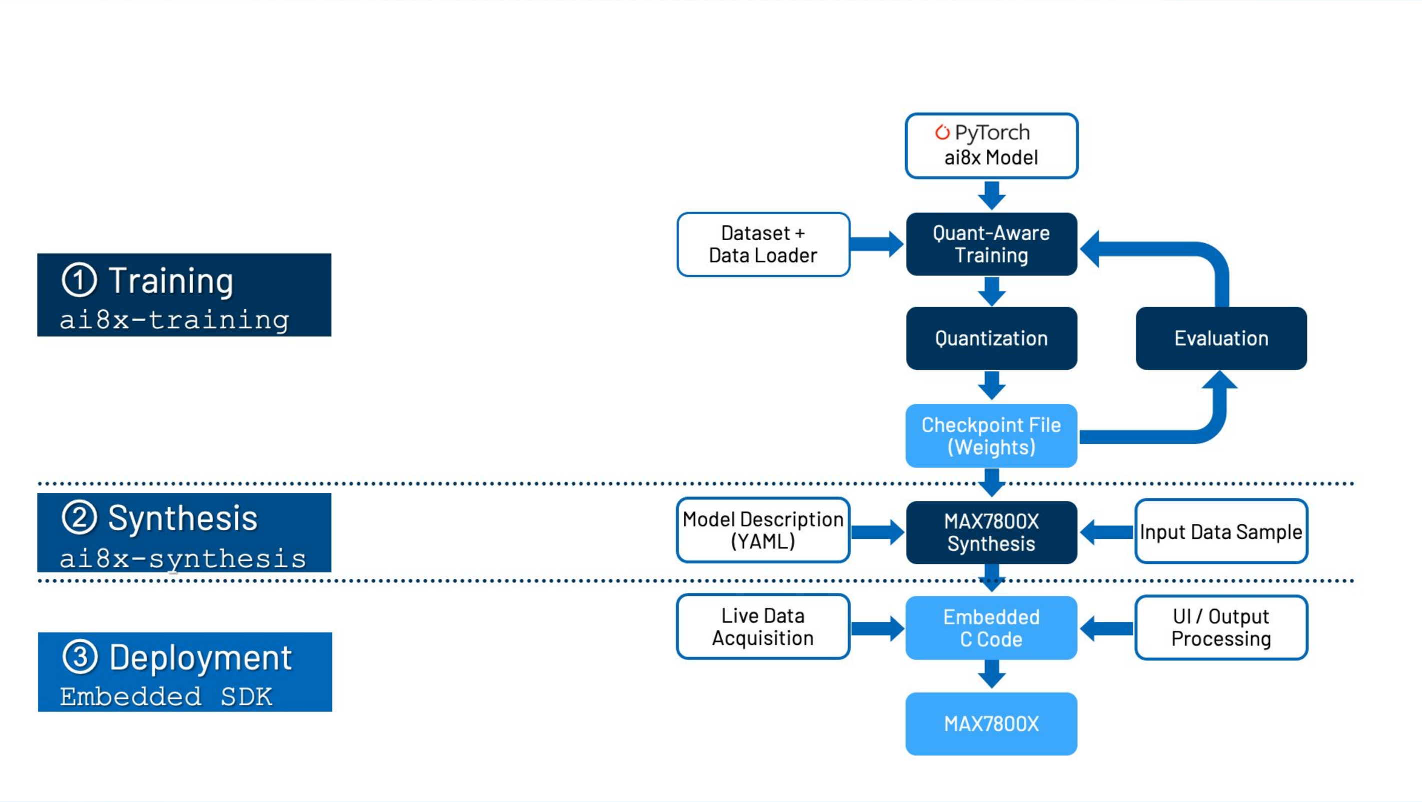
MAX78000 Limitations

- 442 KB model weight limitation
- 180x180 image resolution (RGB) (without CNN streaming mode)
- 224x224 image resolution (RGB) (CNN streaming mode)
- 512KB flash memory, 128KB SRAM
  - Firmware
  - Weight and Bias
  - Peripheral (camera, mic, etc) Buffer
- Can only do pooling before convolution
- Only supports fixed 1d and 2d kernels (1x1, 3x3)
- CNN accelerator only supports up to 32 layers (78002 is not limited)
- Manual processor and memory allocation for CNN



MAX78000 Cam02 Module

Using the MAX78000 for your own tasks



① Training
ai8x-training

② Synthesis
ai8x-synthesis

③ Deployment
Embedded SDK

PyTorch
ai8x Model

Dataset +
Data Loader

Quant-Aware
Training

Quantization

Evaluation

Checkpoint File
(Weights)

Model Description
(YAML)

MAX7800X
Synthesis

Input Data Sample

Live Data
Acquisition

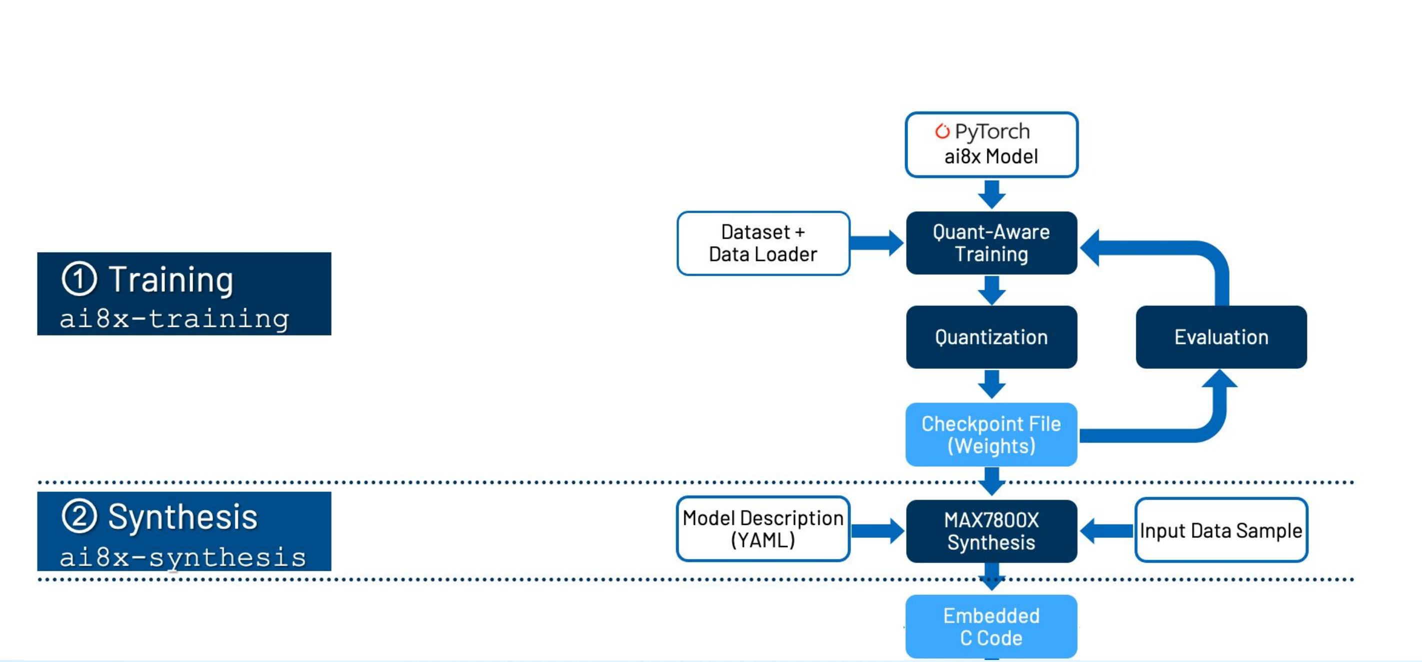
Embedded
C Code

UI / Output
Processing

MAX7800X

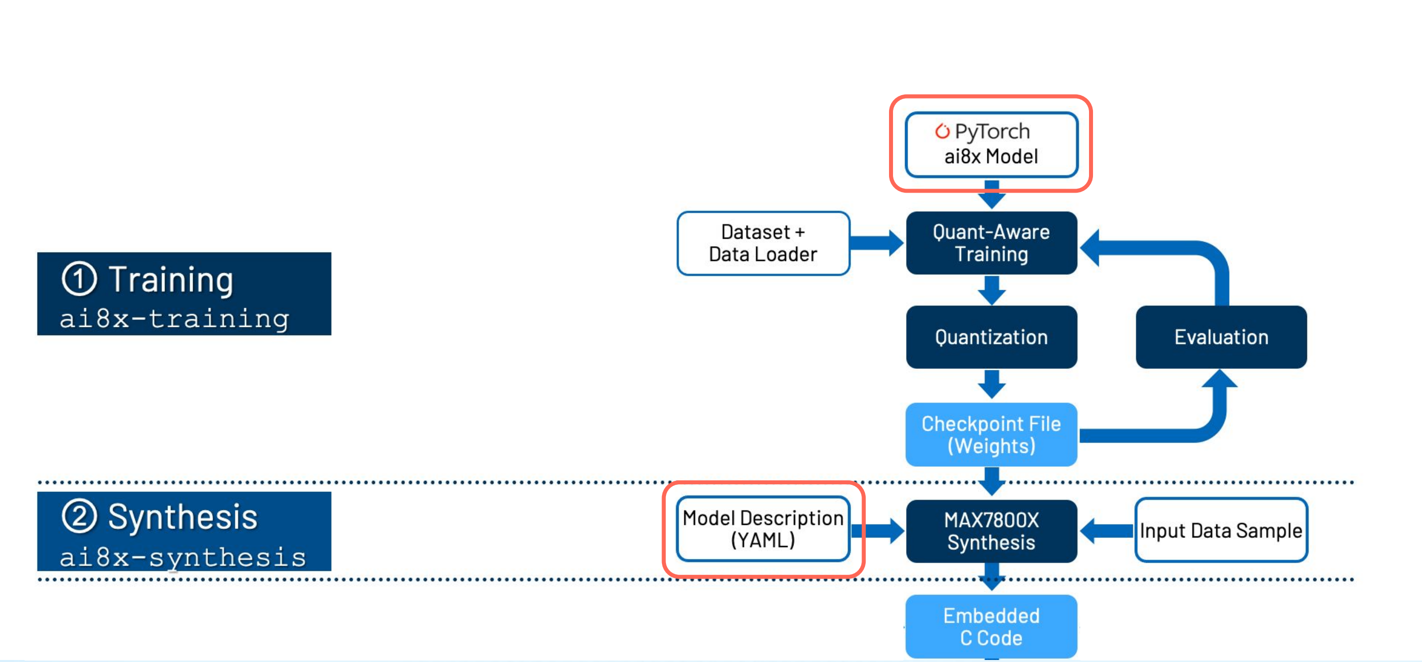
Model Creation (re-use existing architecture)

Framework: ai8x-training, ai8x-synthesis



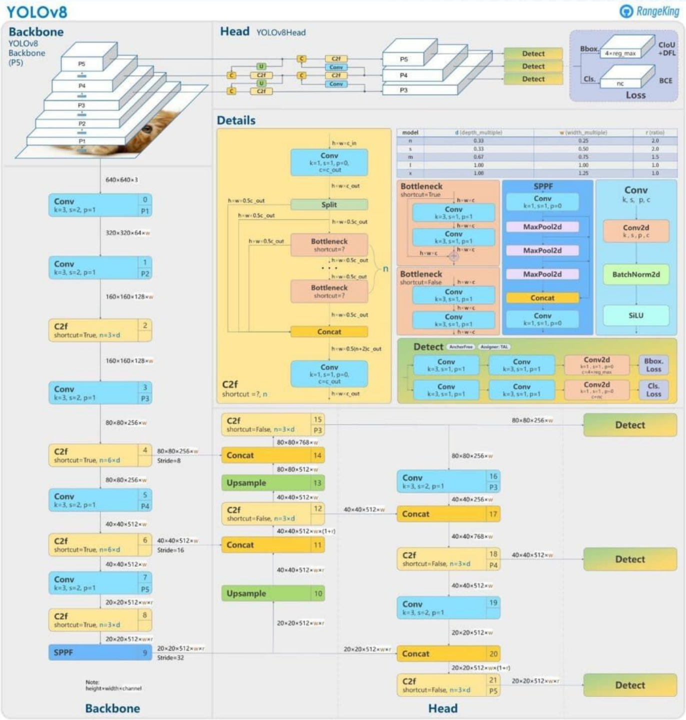
Model Creation (port a new architecture)

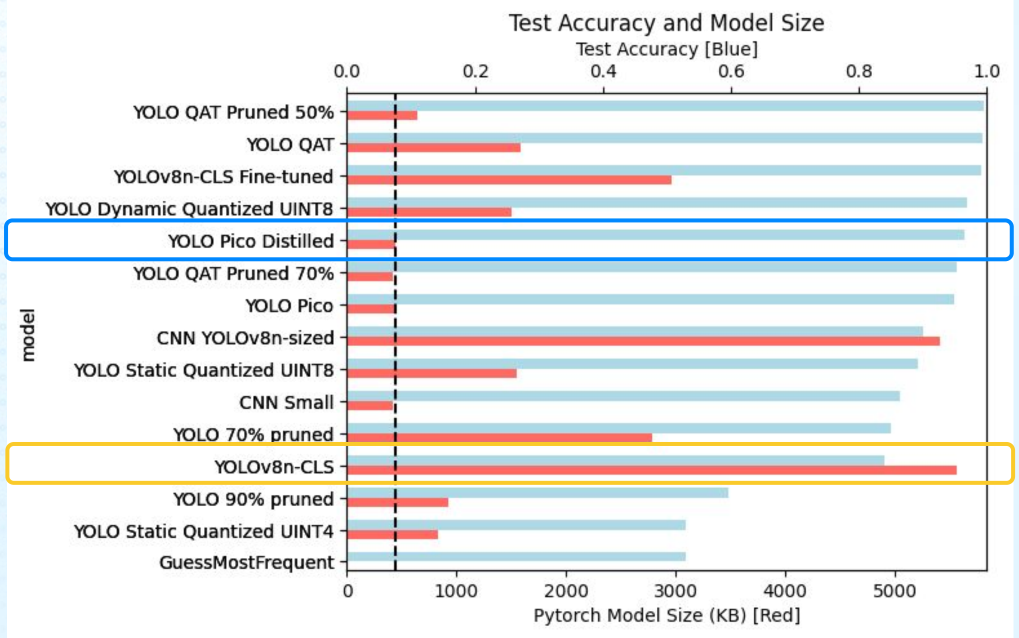
Framework (ours): YADES



Model Creation (YOLOv8p (ours))

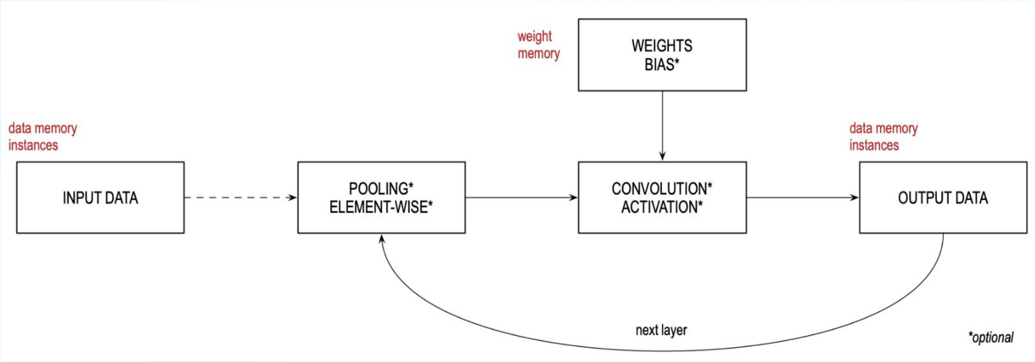
Framework (ours): YADES

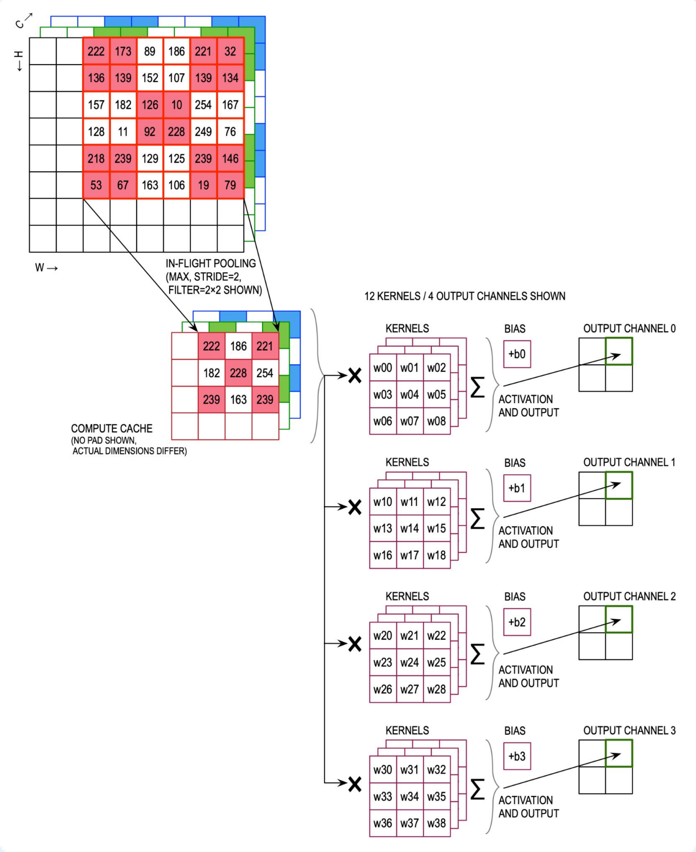




Model deployment (Memory Optimization)

- Streaming mode
  - Reduce memory footprint of inference.
  - Increase maximum input and embedding size. (180x180x4 -> 224x224x4)
  - Higher latency and power consumption
- “Ping-pong” memory usage
  - Non-streaming mode only.

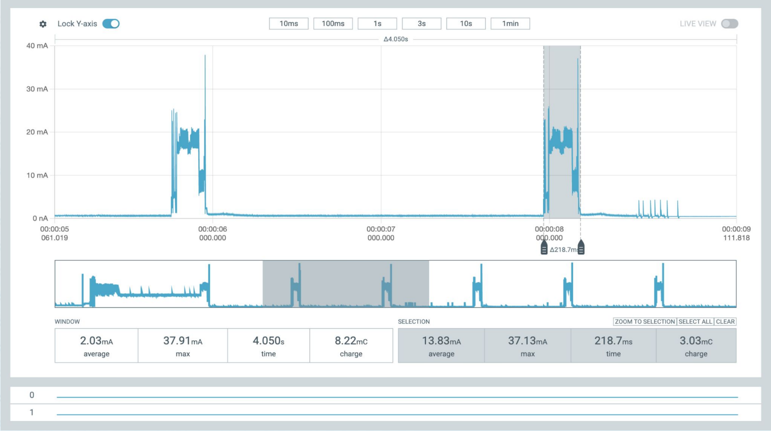




Model deployment (Power/Latency Optimization)

Power

- Undervoltage
  - Stability
- Shutdown unused peripherals
- MCU frequency vs latency vs power

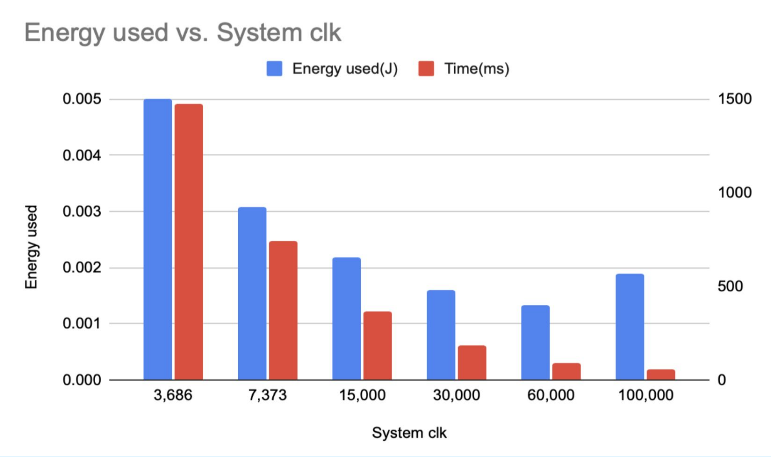


WINDOW

2.03mA	37.91mA	4.050s	8.22mC
average	max	time	charge

SELECTION

13.83mA	37.13mA	218.7ms	3.03mC
average	max	time	charge

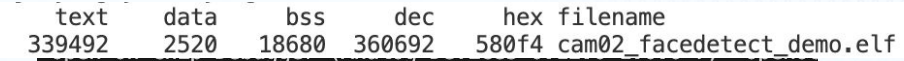


Latency

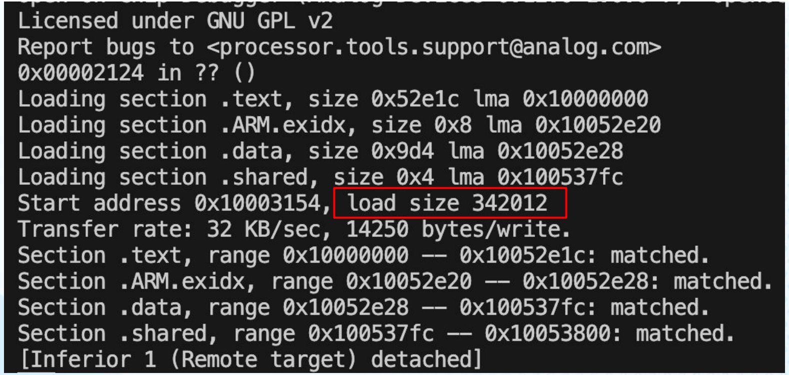
- Parallel execution
  - Example: loading weight to CNN while taking the picture.

Outlier: 30kHz clk lead to 180s inference time and consumes 0.487J

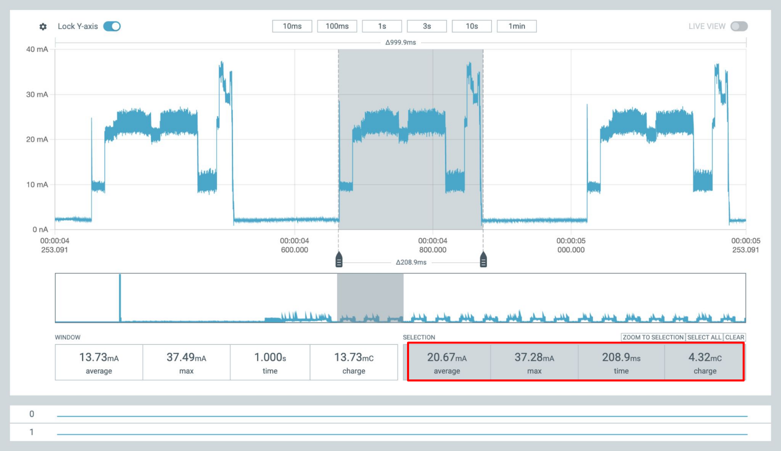
AI85FaceIdNet vs YOLOv8pico (ours)



```
   text    data     bss     dec     hex filename
 339492    2520   18680  360692   580f4 cam02_facedetect_demo.elf
```



```
Licensed under GNU GPL v2
Report bugs to <processor.tools.support@analog.com>
0x00002124 in ?? ()
Loading section .text, size 0x52e1c lma 0x10000000
Loading section .ARM.exidx, size 0x8 lma 0x10052e20
Loading section .data, size 0x9d4 lma 0x10052e28
Loading section .shared, size 0x4 lma 0x100537fc
Start address 0x10003154, load size 342012
Transfer rate: 32 KB/sec, 14250 bytes/write.
Section .text, range 0x10000000 -- 0x10052e1c: matched.
Section .ARM.exidx, range 0x10052e20 -- 0x10052e28: matched.
Section .data, range 0x10052e28 -- 0x100537fc: matched.
Section .shared, range 0x100537fc -- 0x10053800: matched.
[Inferior 1 (Remote target) detached]
```

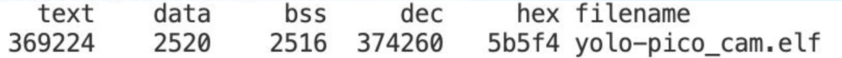


WINDOW

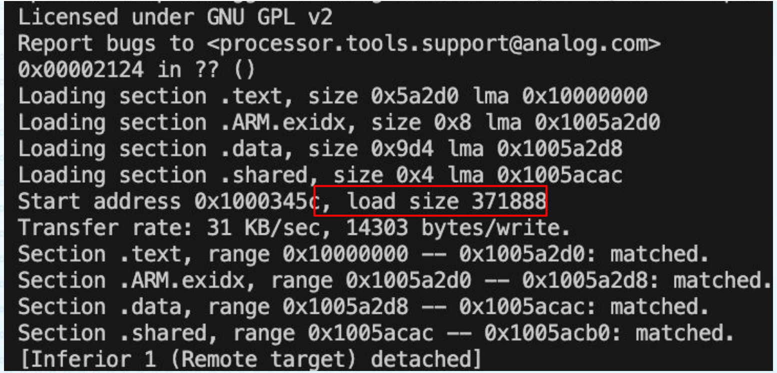
13.73mA average	37.49mA max	1.000s time	13.73mC charge

SELECTION

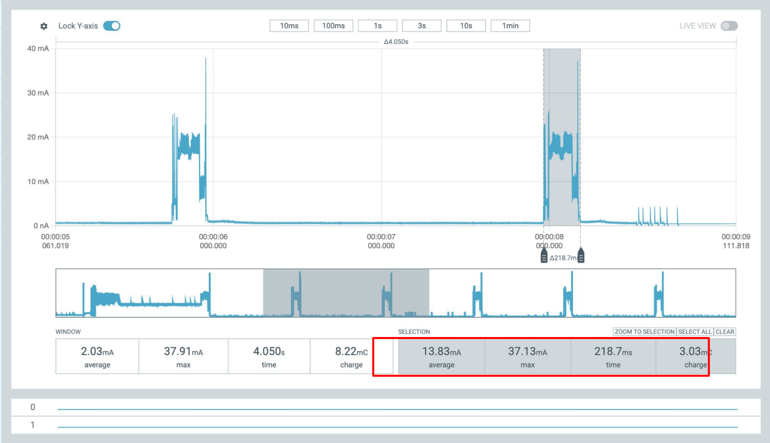
20.67mA average	37.28mA max	208.9ms time	4.32mC charge



```
   text    data     bss     dec     hex filename
 369224    2520    2516  374260   5b5f4 yolo-pico_cam.elf
```



```
Licensed under GNU GPL v2
Report bugs to <processor.tools.support@analog.com>
0x00002124 in ?? ()
Loading section .text, size 0x5a2d0 lma 0x10000000
Loading section .ARM.exidx, size 0x8 lma 0x1005a2d0
Loading section .data, size 0x9d4 lma 0x1005a2d8
Loading section .shared, size 0x4 lma 0x1005acac
Start address 0x1000345c, load size 371888
Transfer rate: 31 KB/sec, 14303 bytes/write.
Section .text, range 0x10000000 -- 0x1005a2d0: matched.
Section .ARM.exidx, range 0x1005a2d0 -- 0x1005a2d8: matched.
Section .data, range 0x1005a2d8 -- 0x1005acac: matched.
Section .shared, range 0x1005acac -- 0x1005acb0: matched.
[Inferior 1 (Remote target) detached]
```



WINDOW

2.03mA average	37.91mA max	4.050s time	8.22mC charge

SELECTION

13.83mA average	37.13mA max	218.7ms time	3.03mC charge

Future Works and Ideas

- Custom PCB with higher energy efficiency coming
- Energy harvesting module integration
- Lora module integration
- Distributed ML encoder node
  - Instead of YOLO-pico, deploy a encoder/filter on the node.
  - Reducing transmission size and frequency.
  - Send data of interest to central hub with greater compute power.
  - Preserve privacy since embeddings could be hard to decode.
- Sensor fusion encoder
- Load balance? Nodes with more energy harvested do more computation.