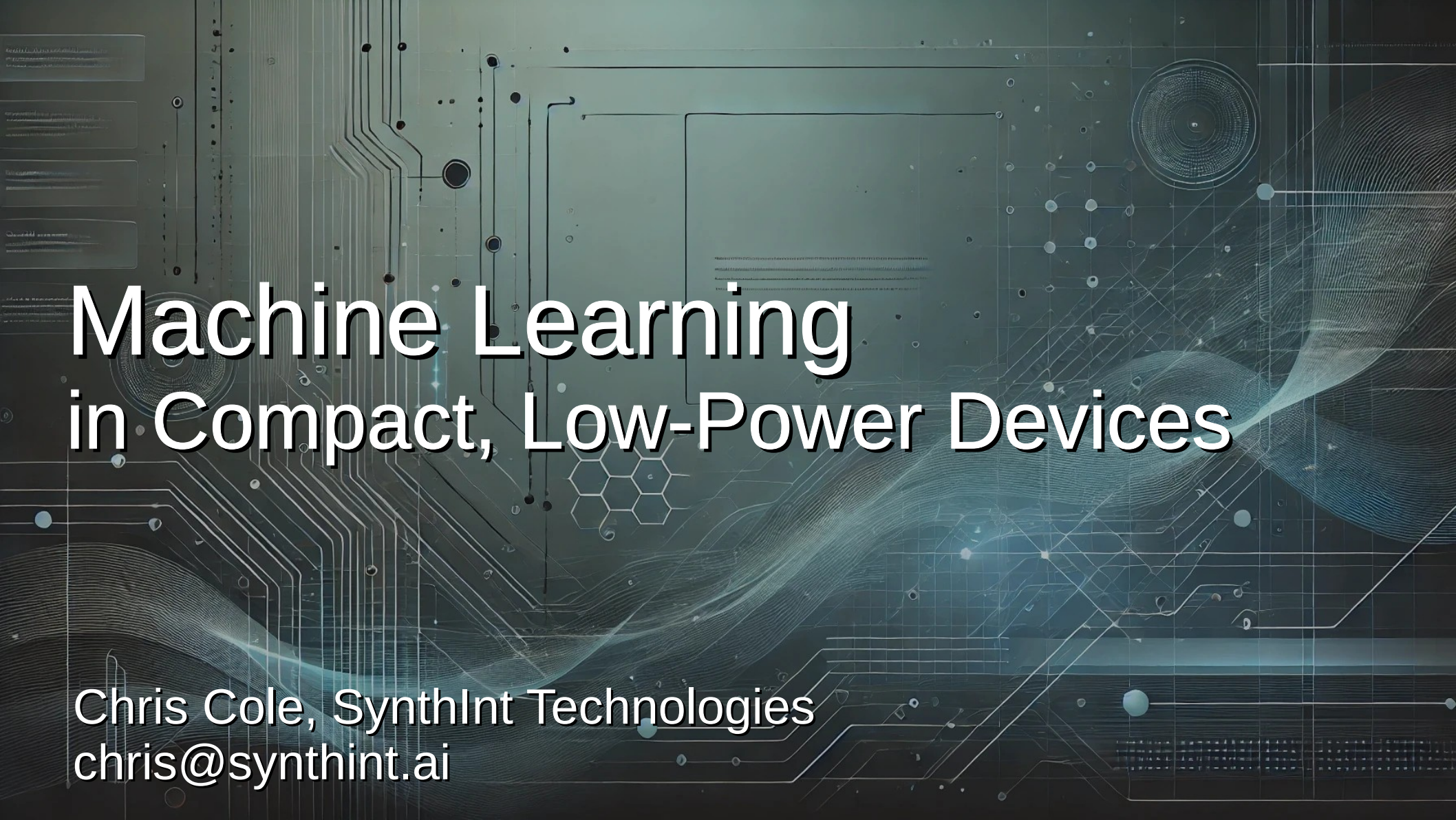
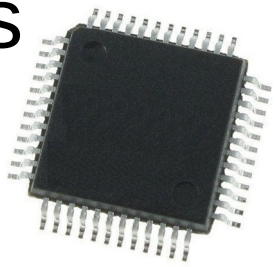


Machine Learning
in Compact, Low-Power Devices

Chris Cole, SynthInt Technologies
chris@synthint.ai

What if I told you that...

You can run an AI algorithm that predicts mechanical failure in a $0.40 chip?

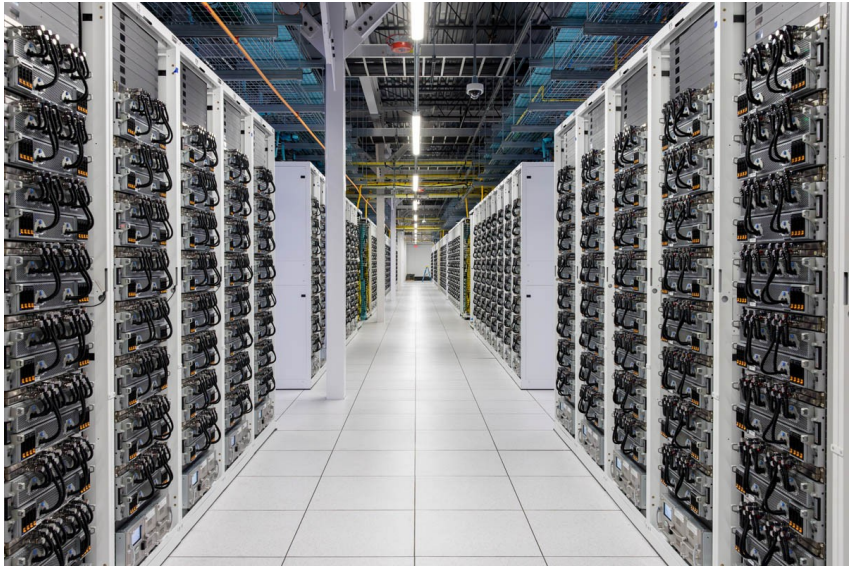


Object detection can be performed by a tiny device the size of your thumb?

A battery powered device can run an LLM and generate text in the palm of your hand?

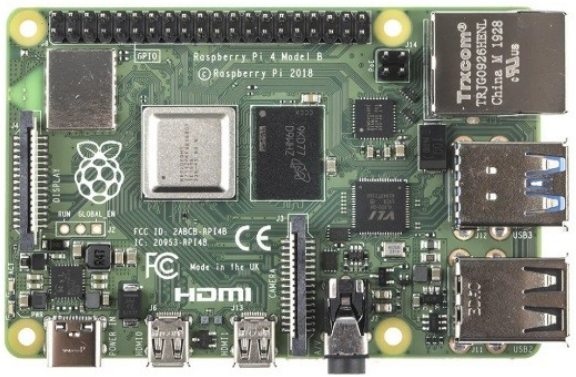
AI Requires a Lot of Compute. Right?

- xAI’s Colossus Cluster
- 100,000 Nvidia H100 GPUs
- Requires 150 MW of power
- Used to train Grok



So, How Is Embedded AI Possible?

- Reduce scope of problem trying to solve
- Reduce model size and hence amount of training data
- Use specific, highly efficient code
  - TensorFlow Lite, MS ELL, SynthInt NN
- Run on a low power, embedded system

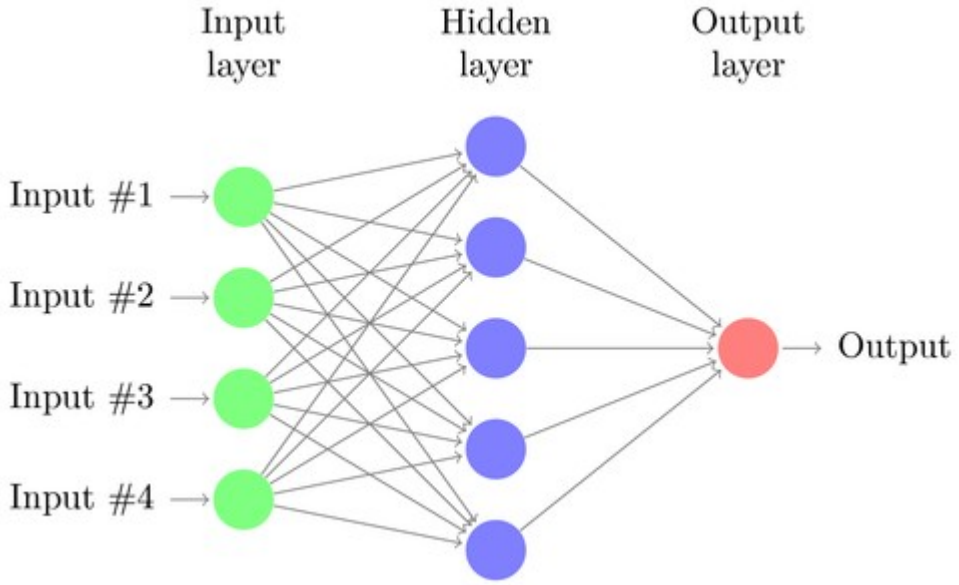


Why AI at the Edge?

- Reduced latency
  - Move computation as close to the data as possible
- Minimize network bandwidth requirements
  - Ability to process this data offline
- Enhance privacy
  - No need to upload data to the cloud

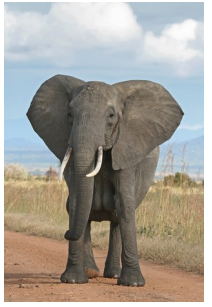
Structure of an ANN

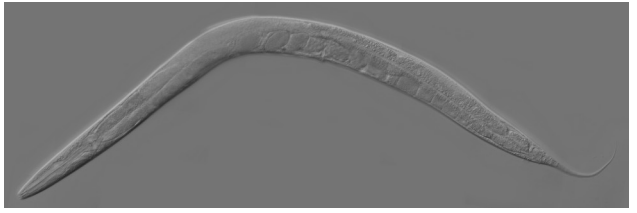
- Neurons have weights and activation functions.
- The order of the approximation is easily tuned through the architecture of the ANN.



Neural Network Sizes

- Sperm whale: 500 billion neurons
- African elephant: 257 billion neurons
- Human brain: 86 billion neurons
- Fruit Fly: 135 thousand neurons
- Roundworm: 302 neurons





- State of the art ANNs:
- 2020: 16 million neurons
- 2023: 1.8 trillion neurons in GPT4

- Typically, my ANN applications to date: < 200 neurons

Training vs. Inference

- Inference is much faster than training

- Example
  - May take about 20 minutes to train a 200 KB model (using a powerful desktop PC)
  - An MCU can inference this model in under 1 ms

Embedded Platforms: Jetson Nano

Quad-core Cortex-A57 @ 1.43 GHz

128-core Nvidia Maxwell GPU

4 GB LPDDR4

16 GB eMMC

LAN, USB, HDMI

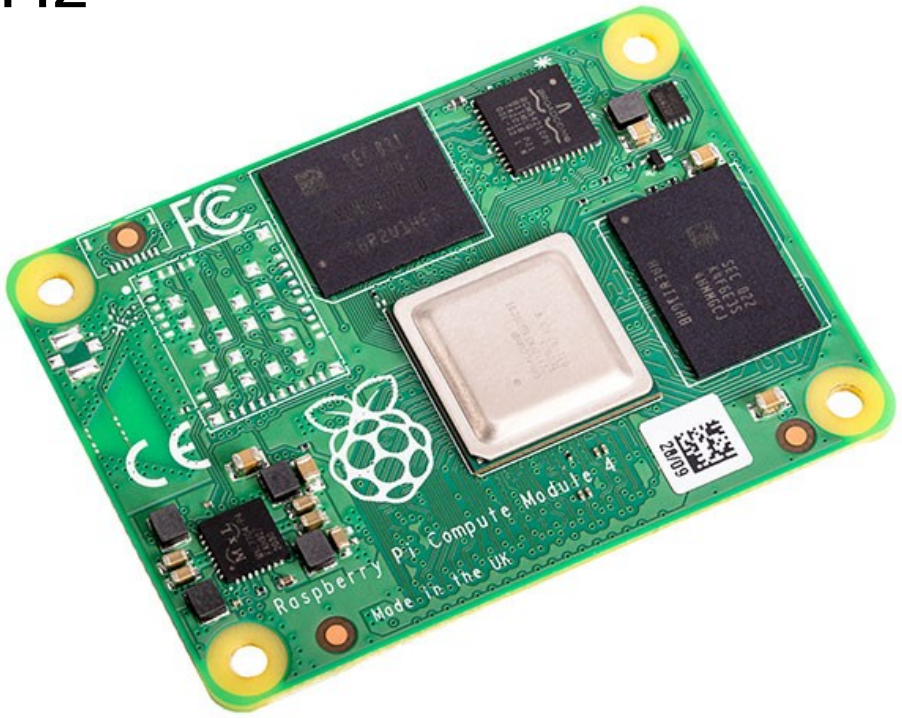
Embedded Platforms: RPI CM4

Quad-core Cortex-A72 @ 1.5 GHz

8 GB SDRAM

32 GB eMMC

LAN, USB, HDMI



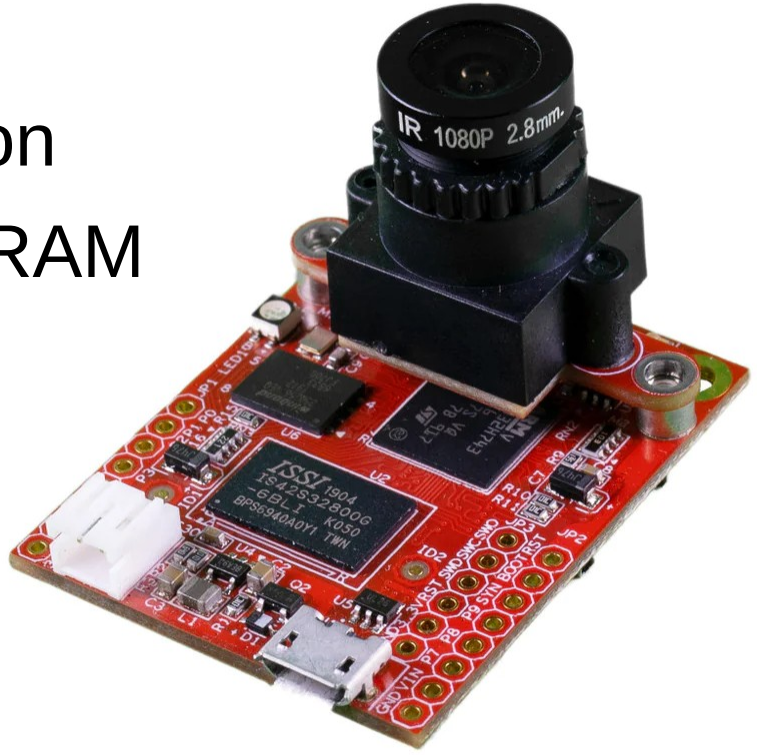
Embedded Platforms: OpenMV H7

480 MHz Cortex-M7 MCU

5 MP camera (2592 x 1944), IR option

2 MB flash, 1 MB SRAM, 32 MB SDRAM

SD Card to store model and data

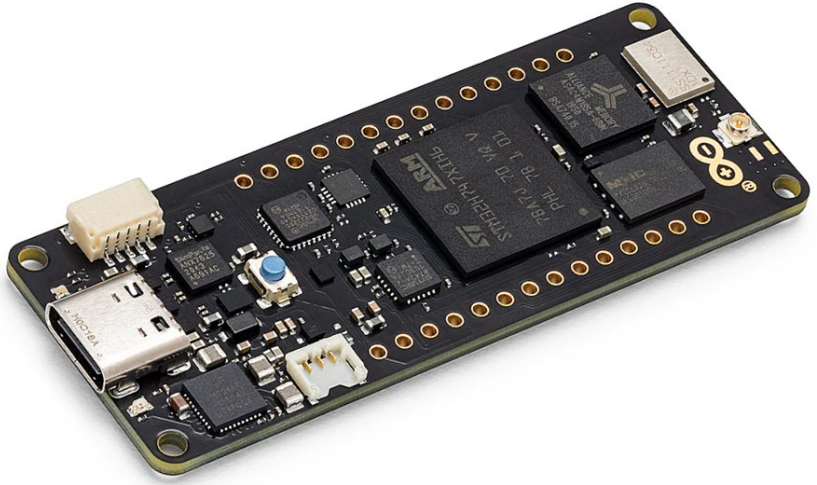


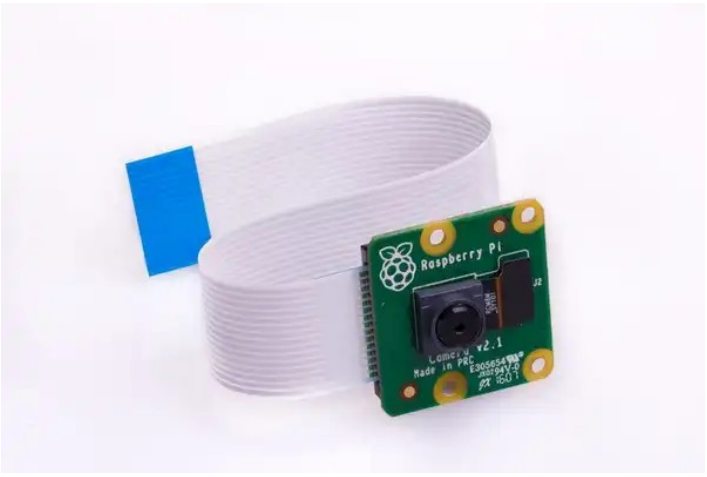
Embedded Platforms: Portenta H7

Arduino

STM32H747 (Cortex-M7 @ 480 MHz + M4 @ 240 MHz)

Edge Impulse, MicroPython, TensorFlow Lite





Embedded Platforms: Google Coral

i.MX 8M SoC Cortex-A53

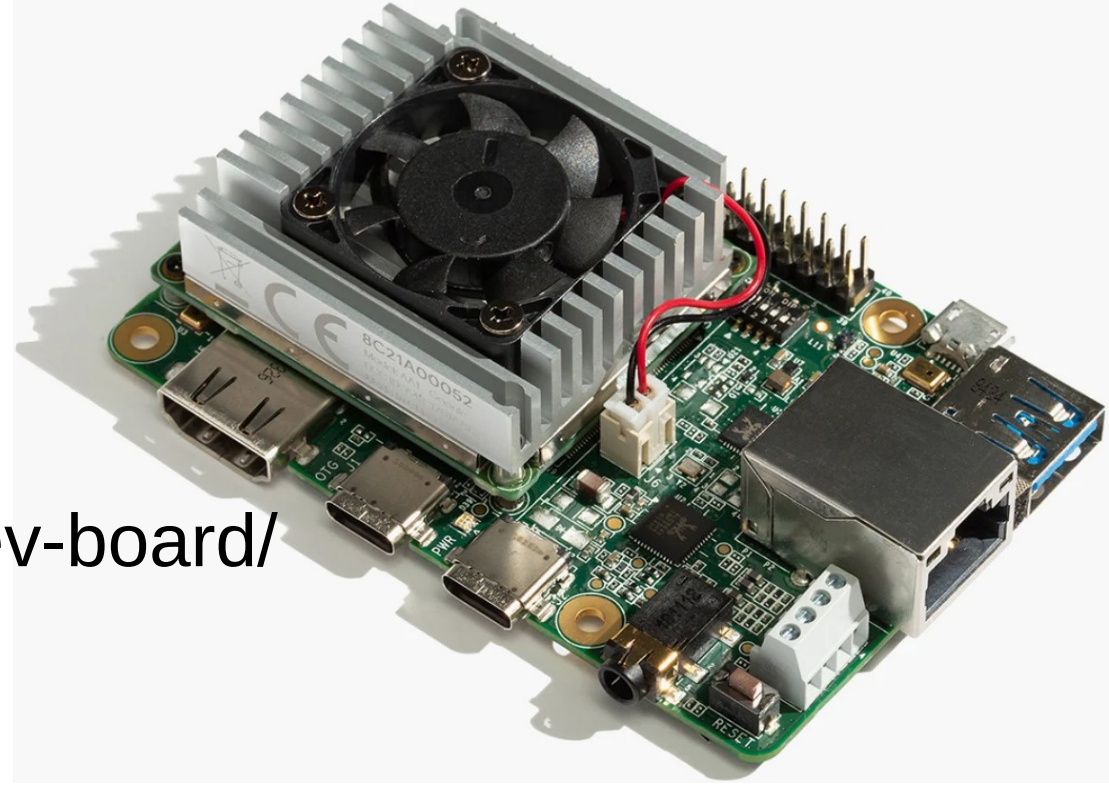
Google Edge TPU coproc.

8 GB eMMC

4 GB LPDDR

Wi-Fi, LAN, USB, A/V

https://coral.ai/products/dev-board/



Embedded Platforms: STM32 Disco

STM32H7@480 MHz

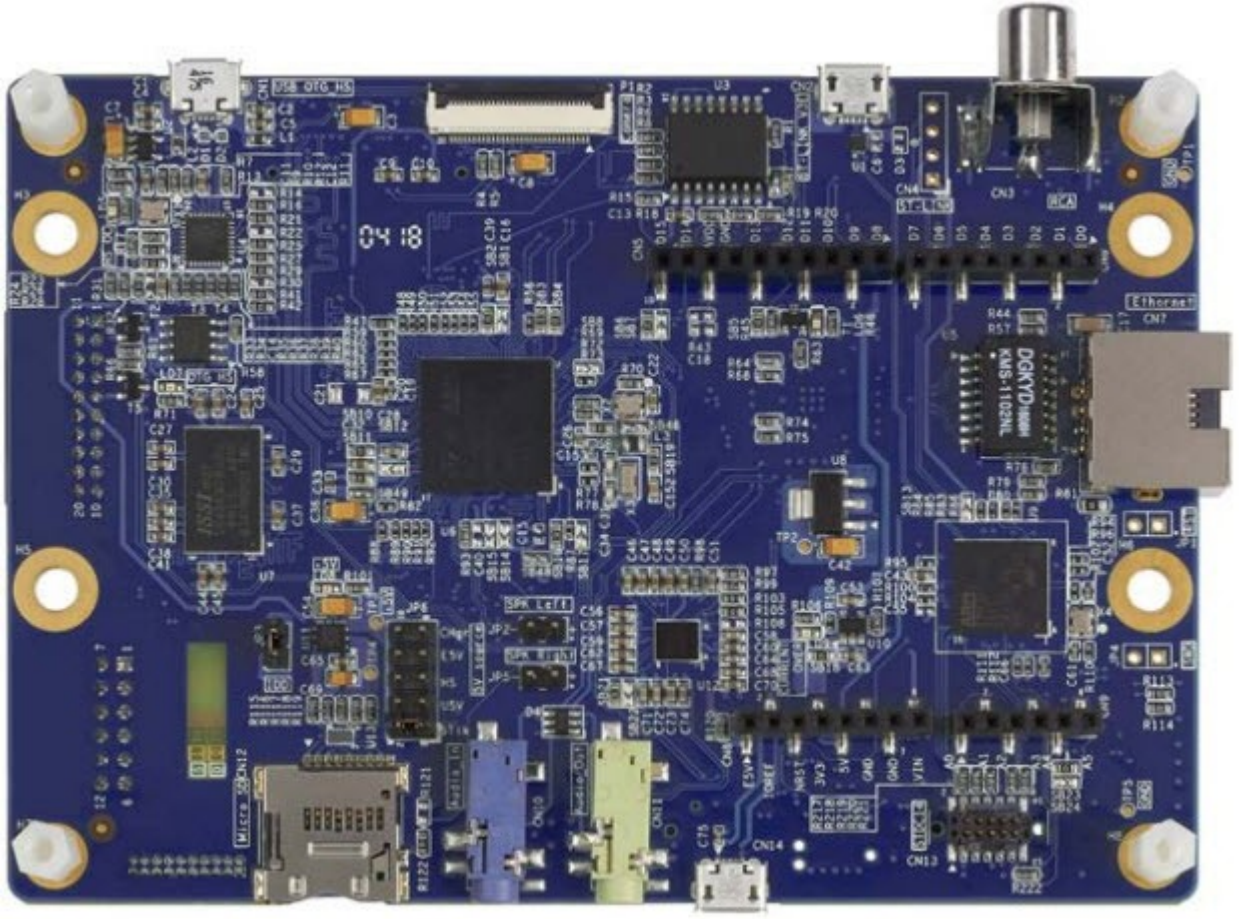
2 MB flash

1 MB SRAM

32 MB SDRAM

SD Card

LAN, USB, A/V

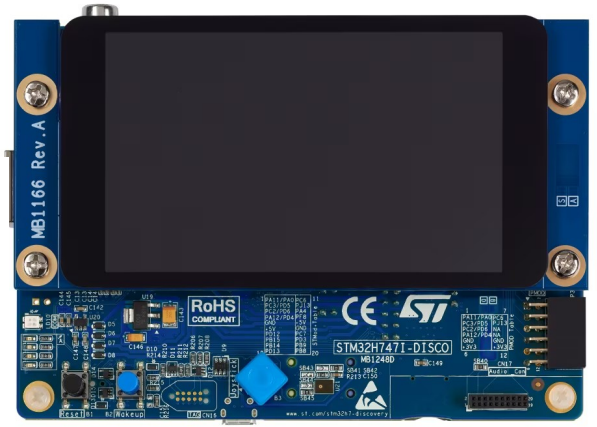


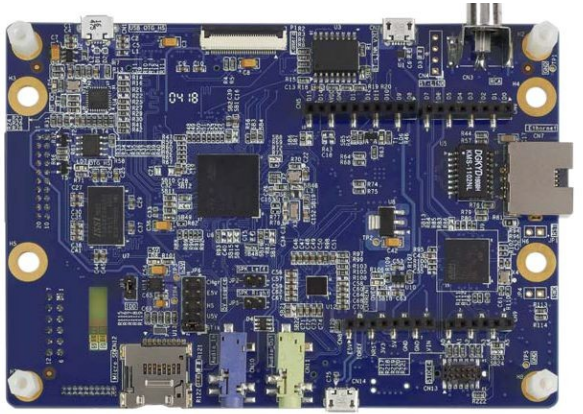
Embedded Platforms: STM32 Disco

OV5640 image sensor

5-Mpixel

8-bit color





Neural Processing Units (NPUs)

AI accelerators for microcontrollers

More energy efficient than CPUs and GPUs

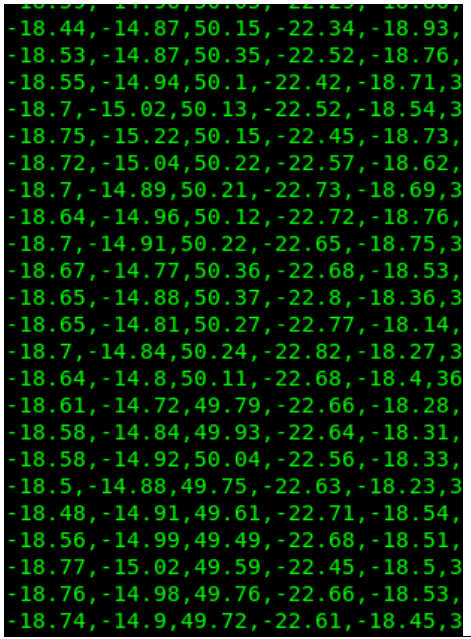
Edge computing capabilities

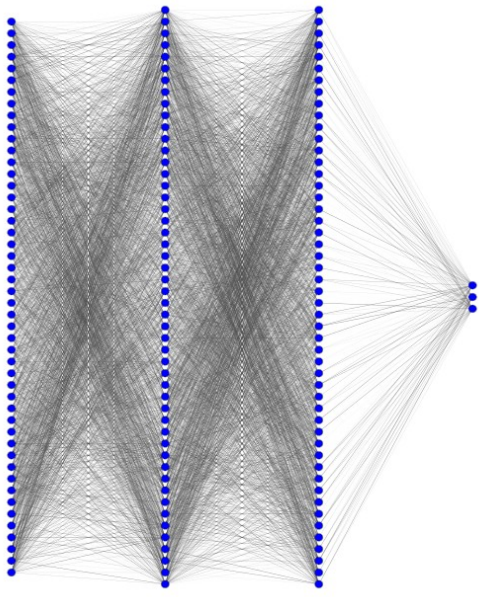
Embedded Applications (ANN)

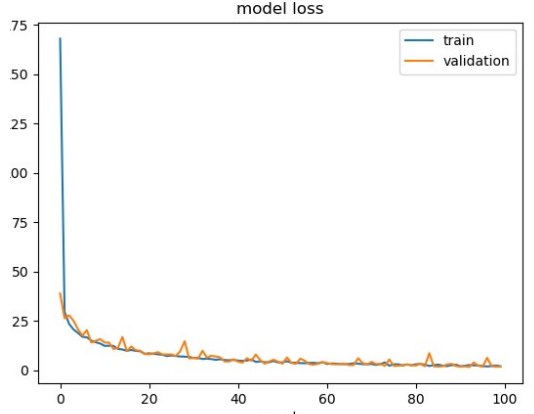
- Predictive analytics
- Object detection
- Biomarker recognition
- Handwriting recognition
- Speech recognition
- PID control loop tuning
- Adaptive motor control
- Robot gait control

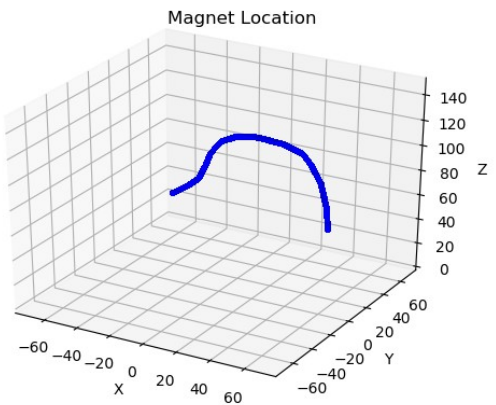
Medical Device Locator

- Neural Network to interpret X,Y,Z location of a magnet
- Sensor readings: 4x4 array of 3-axis magnetometers
- ANN: 48/50(ReLU)/50(ReLU)/3(Linear)









Embedded Applications (LLM)

- Conversational user interface (CUI)
  "Hey fridge, how many eggs do I have?"
- “Chat” with a small data set in a specific application
- Sentiment analysis

What is an LLM?

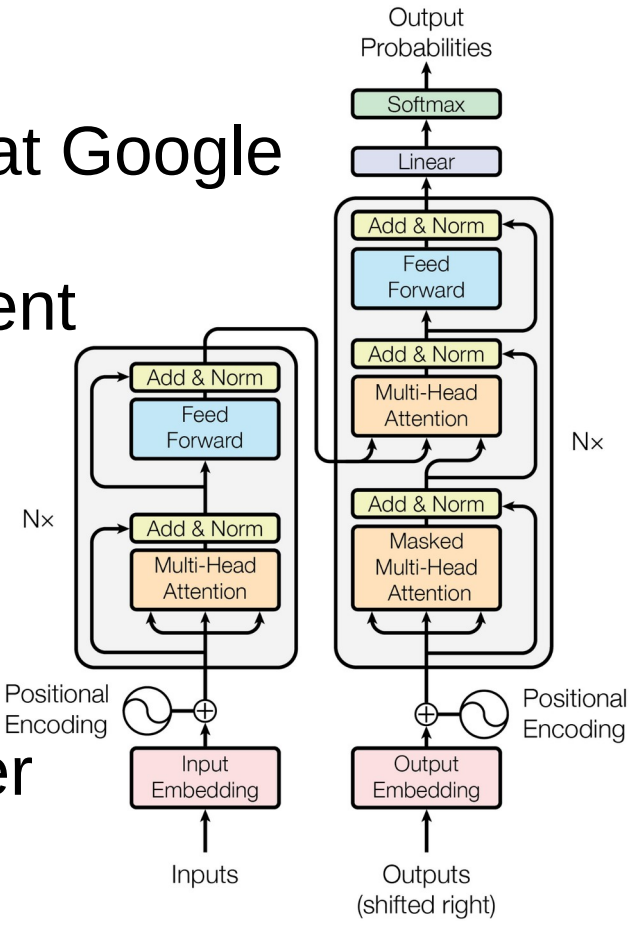
- LLM = “Large Language Model”
- Generative machine learning model that can comprehend and generate human language text

How Does an LLM Work?

- Sentences are split up into smaller units called **tokens**
- **Embeddings** turn the tokens into vectors of numbers
- Embeddings enrich tokenized data with meaning, allowing LLMs to comprehend **context** and **patterns**
- They are numerical representations of contextual similarities between words, and can be manipulated mathematically (king - man + woman = queen)

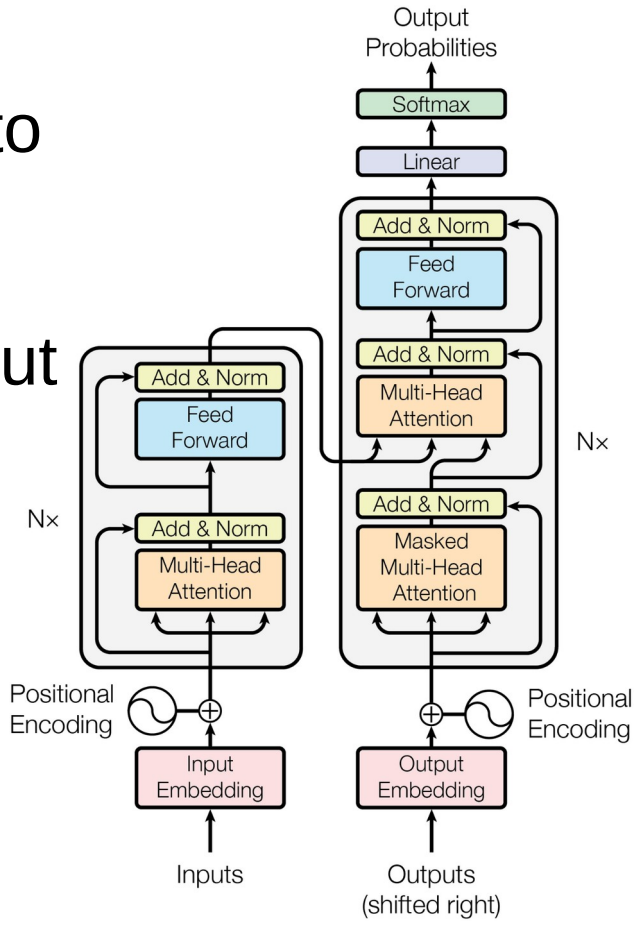
Transformer Architecture

- “Attention Is All You Need”, by 8 scientists at Google
- Human language is highly context-dependent
- Transformer able to learn context
- A mathematical technique called self-attention is used to detect subtle ways that elements in a sequence relate to each other



Transformer Architecture

- Encoder (left side) maps input sequences to a sequence of continuous representations
- Decoder (right side) receives encoder output together with the output at the previous time step to generate an output



Training an LLM

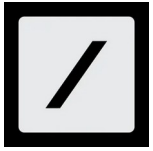
- Trained on tens of terabytes of data
- Curated data sets improve model quality
- Further trained via fine-tuning for a particular task

Popular LLMs

OpenAI ChatGPT

Anthropic Claude

Google Gemini



xAI Grok

Training Costs

GPT-3

- Cost $4.6m to train (45 TB compressed text)
- Using 1024x A100 GPUs, researchers calculated that OpenAI could have trained GPT-3 in 34 days

GPT-4

- Cost $63m to train (Altman indicates $100m)
- Estimated 90-100 days to train

Local LLM

LLMs can be very large

Smaller models are available that may be run on a desktop

Model size reduction through quantization

llama3.1:8b	4.7 GB
llama3.1:70b	39 GB
llama3.1:405b	228 GB

Embedded LLMs for MCUs

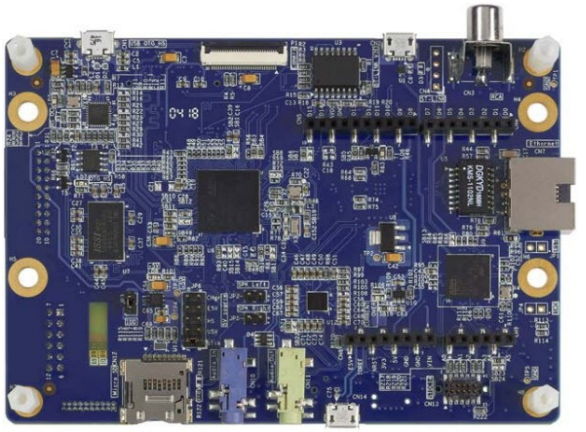
Small memory footprint (5 – 30 MB)

Low power consumption

Reduced training data required

Embedded LLM Demo

- Demonstrate an LLM running in an STM32H7 MCU
  480 MHz Cortex-M7, 2 MB flash, 1 MB SRAM, 32 MB DRAM
- Dynamically generates a random TinyStory
- Output via serial port
- MCU consumes 75 mA to run the demo



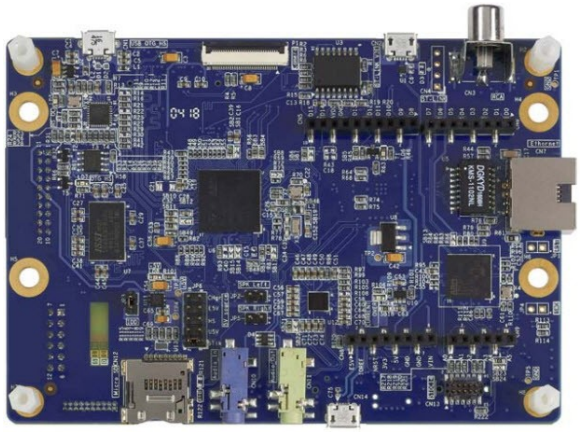
Embedded LLM Demo

```
One day, a boy named Tim went to the park with his mom. Tim was a very
popular boy, always happy and modest. His mom told him, "Tim, you must behave
when you play with your friends."While playing, Tim found a big box. He
opened the box and saw a pretty jewel. He picked it up and showed it to his
mom. "Look, mom! I found a pretty jewel!" he said. His mom smiled and said,
"Wow, Tim! That's amazing! You did a great job!"Then, Tim's friend Sue came
to the park. She saw the pretty jewel and asked, "Where did you get this?"
Tim showed her the pretty jewel. Sue was excited and said, "I found it! I am
so excited!" They played with the jewel all day, and Tim knew he did a great
job.

Once upon a time, there was a little boy named Timmy. Timmy loved to play
outside in the mud. One day, he got very dirty and needed to bathe. His mom
said, "Timmy, it's time to take a bath!" Timmy didn't want to stop playing,
but he knew he had to listen to his mom.After his bath, Timmy's mom gave him
a bottle of milk. Timmy drank the milk and smiled. He realized that he had a
good time playing in the mud. When he got out of the bath, he showed his mom
his clean milk and said, "I love you mommy!" His mom hugged him and said, "I
love you too, Timmy."
```

Training the LLM

- Obtain data (9 GB of text in the TinyStories demo)
- Train a model on the data using PyTorch
- Convert the model for use by the llama2.c code
- Model in this demo was quantized to 8-bit
- Resultant model sizes:
  4.6 MB (faster but less accurate)
  19.9 MB (slower but more sensible)



Limitations

- As models get smaller or overquantized, they lose quality
- “Hallucinations” generate incorrect output at high confidence
- Training bias
- Explainability

Legal Implications

- IP ownership of training data and results
- Copyright infringement
  - How does an AI implementation learn?
  - How do we learn?
- Liability