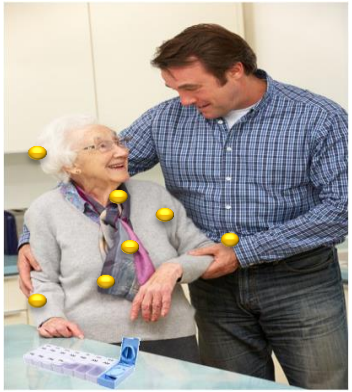
November 2 – 5, 2021

ML@ExtremeEdge of *Always-on* Intelligent Sensor Networks

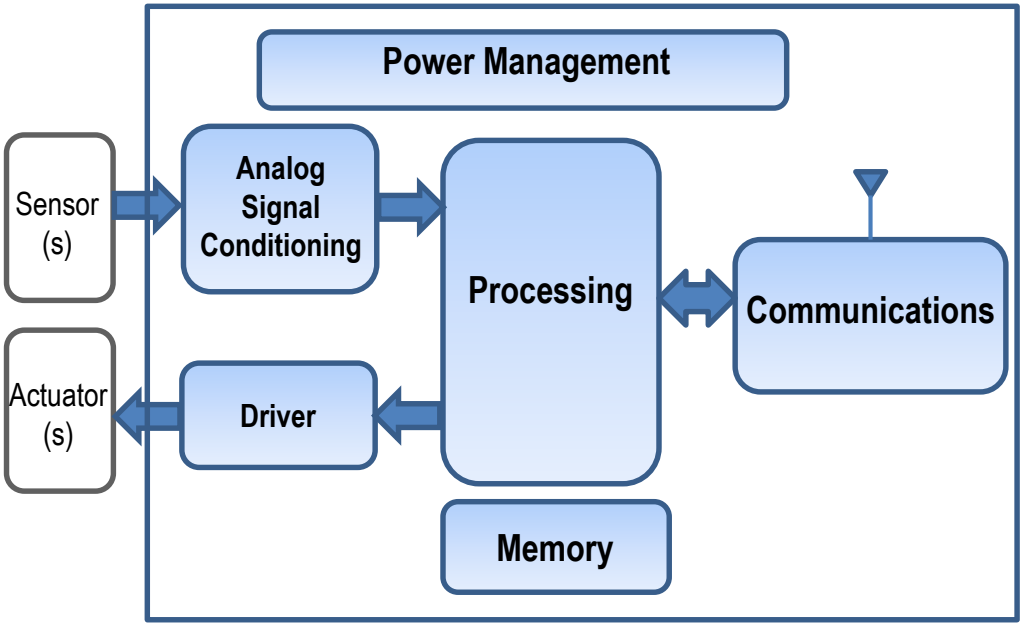
Mahesh Mehendale

Nov 2nd, 2021

Intelligent Internet of Things – Sensor Nodes



Extreme Edge: Sensor Node

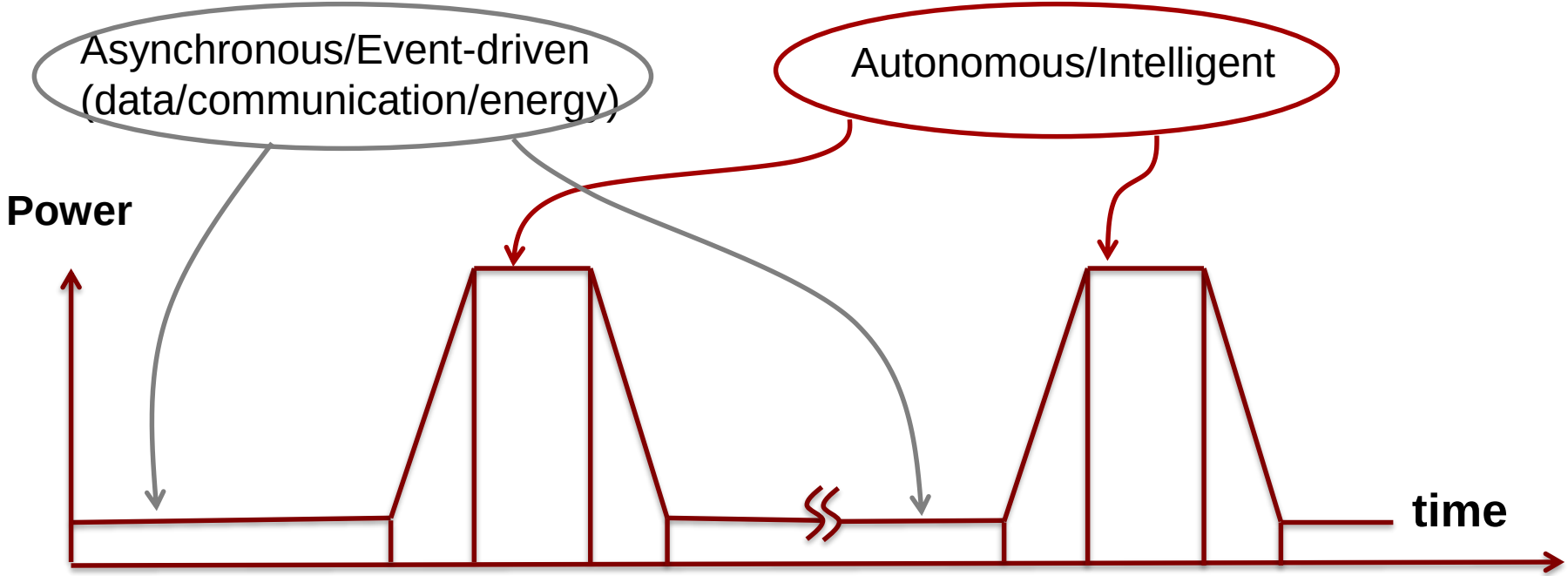


Key Requirements:

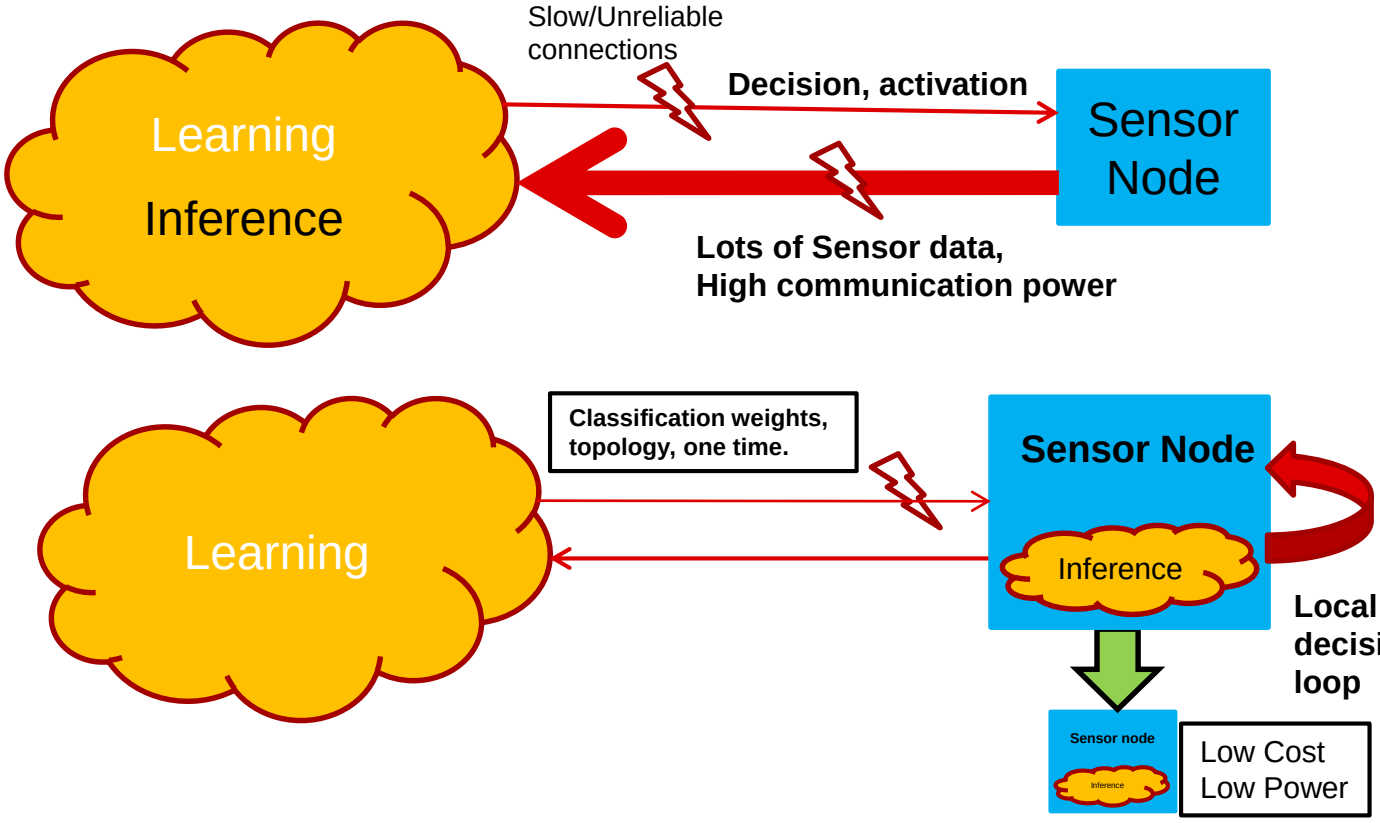
- Low cost
- Small form factor
- Ultra low power
  - Extended battery life of 10+ years
  - Cheaper/smaller batteries
- Modest performance requirement (in terms of sample rate and resolution)
- Integrated analog interface, power management, NVM storage, RF and digital processing in a single device: “analog friendly” technology node

The New Low Power Challenge

"always-on" with "event of interest" occurring infrequently

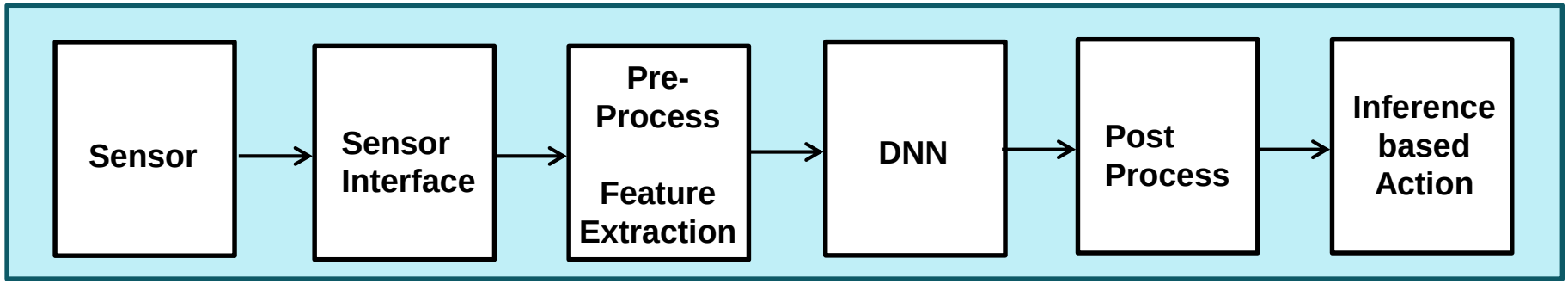


Why Intelligence at Extreme Edge?



- Low latency (real-time local processing)
- Reduced/no dependence on cloud
- Privacy enhancement
- Reduced load on already saturated network
- Energy efficiency for the node as well as the network

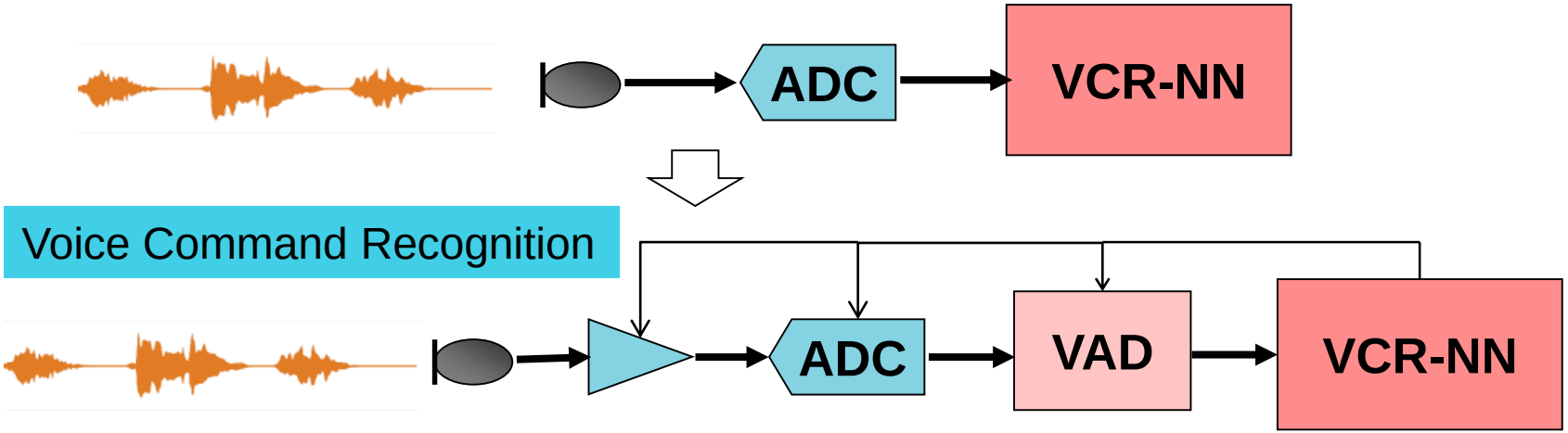
DNN based intelligence at the Extreme Edge



Accuracy ⇔ Performance ⇔ Power ⇔ Cost

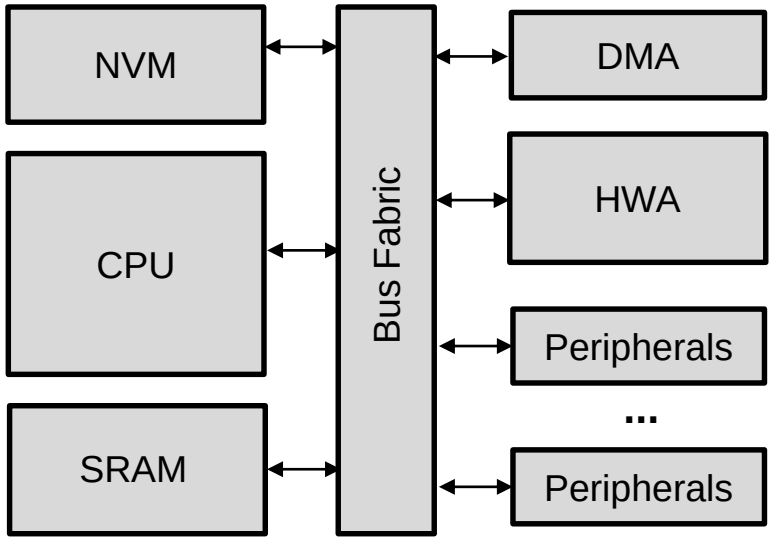
- System Level Optimization
- NN Algorithm Design
- NN Complexity Reduction
- NN optimized HW-SW architecture
- Circuit-level Optimization

System-level optimization



- Hierarchical Detection
- Context aware modulation of sample rate and bit precision
- Event driven sensing (asynchronous analog-to-digital conversion)

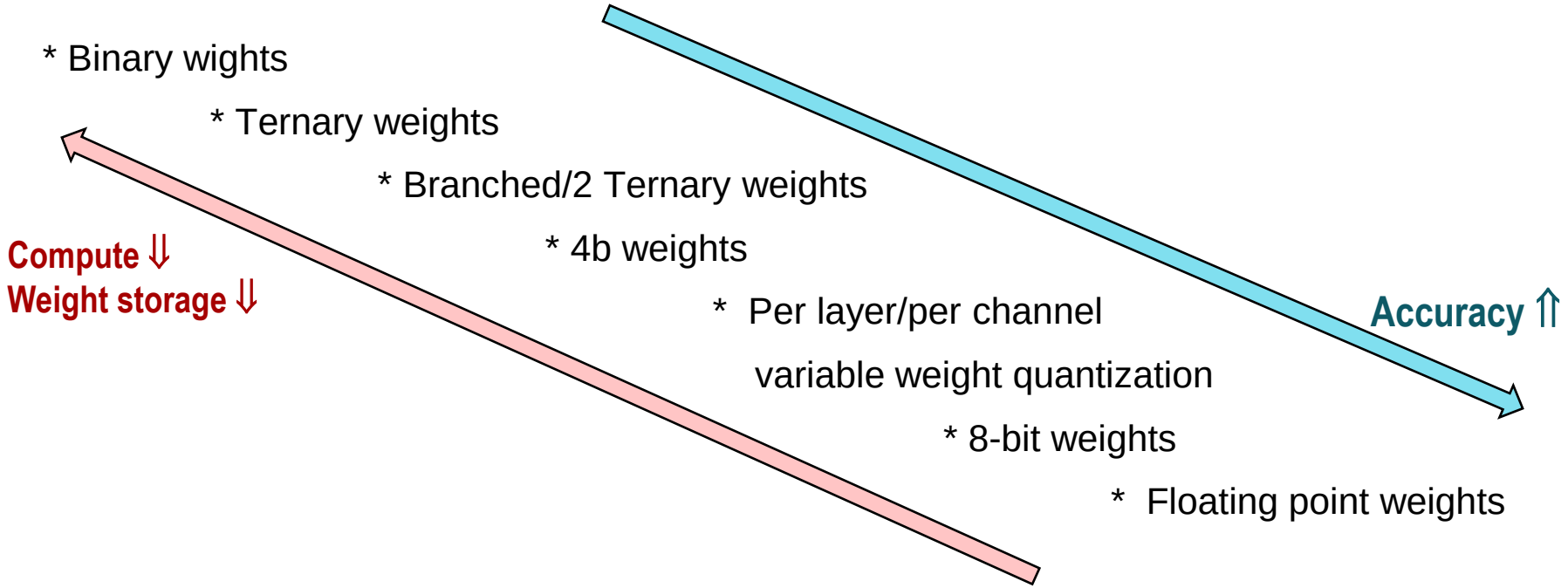
Low cost, Low Power DNN inference on end nodes



MCU Processing + Memory sub-system

- Device constraints:
  - All memory on chip – no external memory interface
  - Model parameters need to fit within available NVM (typically 4x of SRAM)
  - Peak SRAM requirements – need to be less than available SRAM
  - Compute complexity – need to meet real-time requirements and lower power on the target hardware architecture – as older technology nodes and low leakage processes drive modest clock rates
- NN algorithm (features, topology selection) and NN complexity reduction need to be co-optimized with hardware architecture – so as to meet desired accuracy and real-time performance.

NN complexity reduction – aggressive weights quantization



Peak feature-map storage reduction

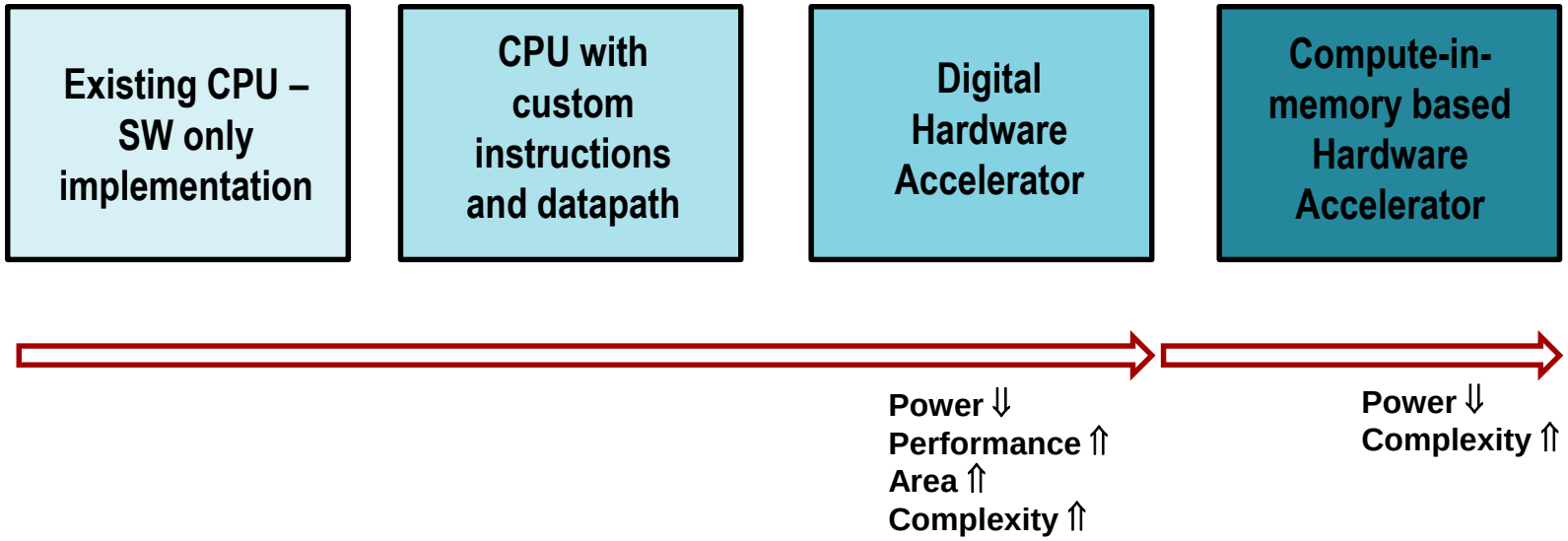
- Activations quantization e.g. 4 bit data
- Exploiting overlap between successive input feature maps

NN model augmentation techniques for exploring quality vs. complexity tradeoffs

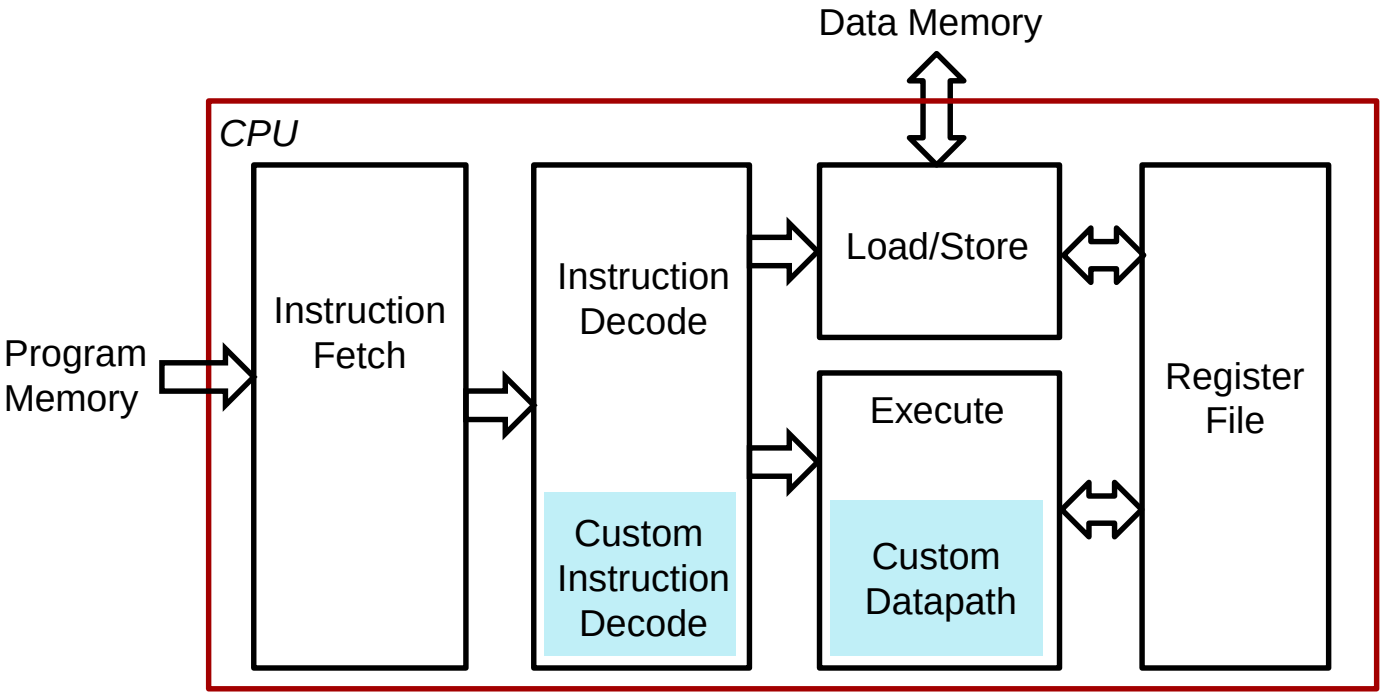
	FLASH	SRAM	Latency/Power
• Scaling number of channels	X	X	X
• Scaling input feature-map		X	X
• Adding layers	X		X

Need hardware aware integrated neural architecture augmentation + quantization framework – which drives “acceptable” accuracy at low power meeting real-time performance within resource constraints.

Target Hardware options



Generic Customizable RISC architecture



Baseline embedded RISC CPUs can be enhanced to:

- handle bit-level packed data (binary, ternary weights),
- support conditional add/sub/nop operations for multiplication with binary/ternary weights

LUT (lookup table) based pointwise convolution

Consider Pointwise Convolution: M input channels, N output channels, feature-map size of WxH

```
for  (i = 0 to H – 1) {
      for (j = 0 to W – 1) {
          for (m = 0 to (M/4)-1) {
              for (n = 0 to N-1) {
                  Y[n][i][j] += (X[4*m][i][j] * K[4*m,n] +
                                X[4*m+1][i][j] * K[4*m+1,n] +
                                X[4*m+2][i][j] * K[4*m+2,n] +
                                X[4*m+3][i][j] * K[4*m+3,n]) }}}}
```

When K's are binary (+1/-1) weights, the inner 4 term weighted sum compute can take up to 16 possible values.

If these values are pre-computed into a lookup table, the number of computations (ADD/SUB operations) can be reduced by ~4X from (M-1)*N to (M/4-1)*N

The same LUT approach can be used to reduce compute for Ternary weights as well.

Summary

- Always-on Intelligent Sensor node applications demand the “event-of-interest” detection to be done locally.
- This drives need to support low cost and ultra-low power DNN inferencing
- We need co-optimization across system, algorithm, architecture, design and circuit level techniques
- SW based implementation on standard CPUs is not adequate to meet latency, power requirements for many applications
- We need
  - Custom hardware – that can efficiently handle aggressively quantized NNs
  - Hardware aware integrated neural architecture augmentation + quantization framework – which drives “acceptable” accuracy at low power, meeting real-time performance within memory resource constraints

THANK YOU