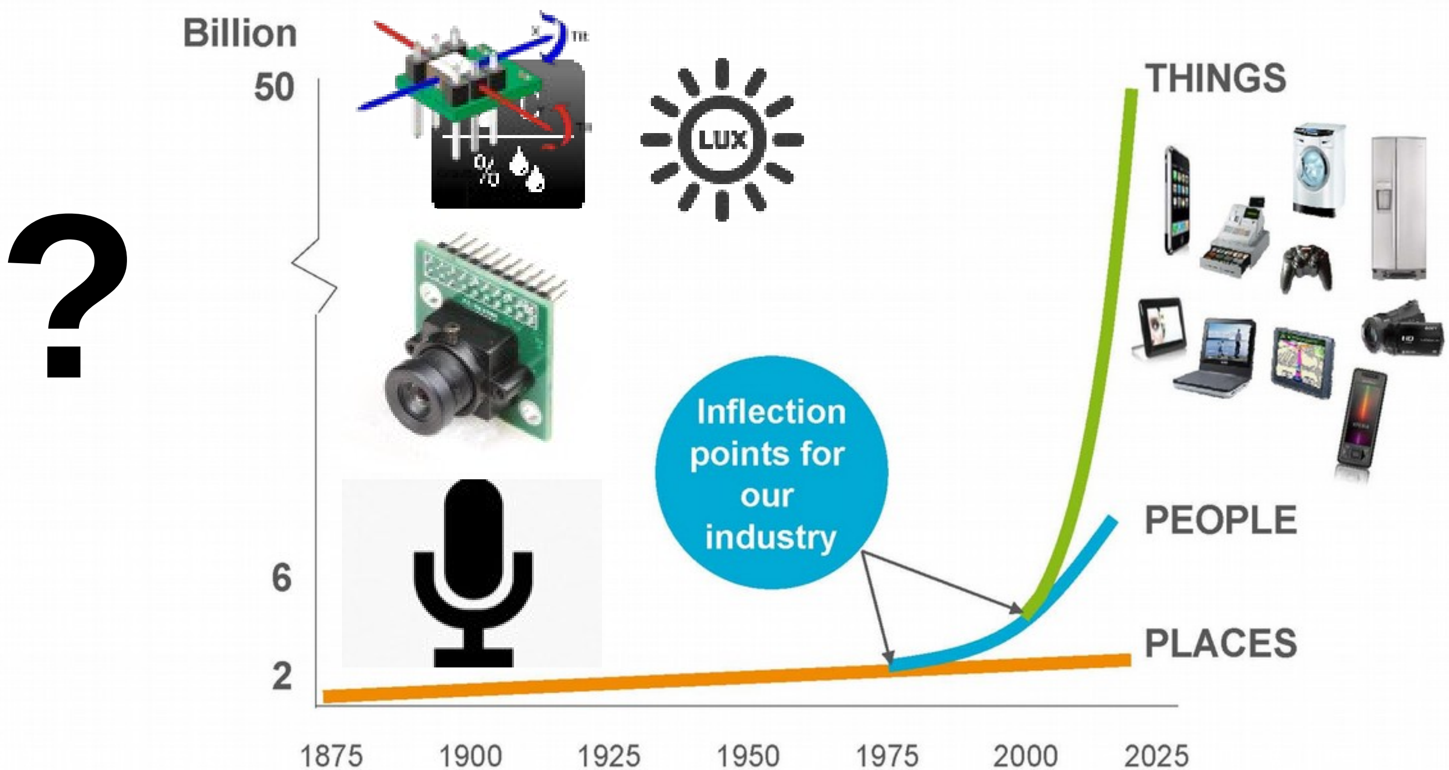
GAP8 IOT Application Processor

A PULP/RISCV BASED PLATFORM FOR NEAR-SENSOR ANALYTICS

Eric Flamand. CoFounder & CTO
Greenwaves Technologies

GREENWAVES TECHNOLOGIES

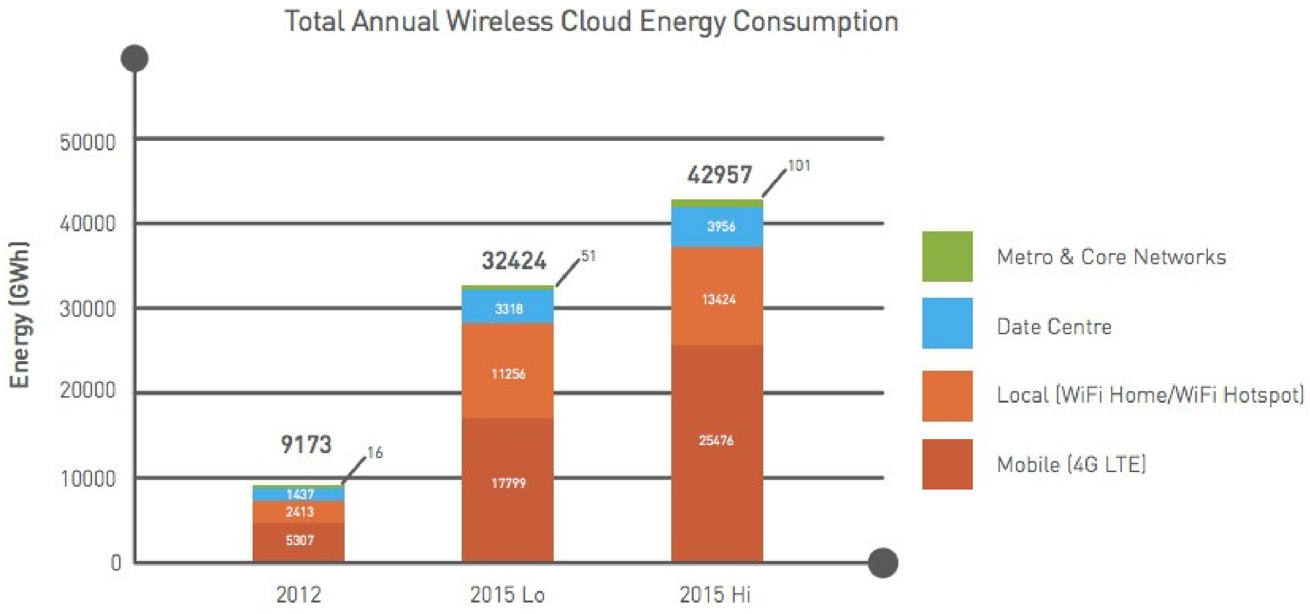
ANYTHING THAT BENEFITS FROM NETWORK CONNECTION WILL BE CONNECTED



Source: Ericsson

Cost of transporting these data over the air ?

Serial short reach link: best results around 0.5 pJ/bit
LTE: between 300 and 600 uJ/bit



Even assuming distributed computing is marginaly more efficient than centralized we win big if data volume to be exchanged over the air is srinked by several order of magnitude moving from quantitative data to qualitative data!

So in practice ….
If we want data reach sensors

Move from (raw) data to meta data (abstract/pertinent)
Perform this transformation close to sensor
While fitting in a tight power and cost budget
And being seamlessly integrated to the Internet over the air

Three main sources of intensive data

- **Image**: *Raw input in the order of 100KB/s for a small sensor*
  - Scene classification
  - Posture analysis
  - Identification

  Output is a single index

- **Voice/Sound**: *Raw input in the order of 10KB/s per mic*
  - Recognition
  - Identification
  - Signature analysis

  Output is a single index

- **Vibrations**: *Raw input in the order of 10KB/s*
  - Preventive maintenance
  - Monitoring

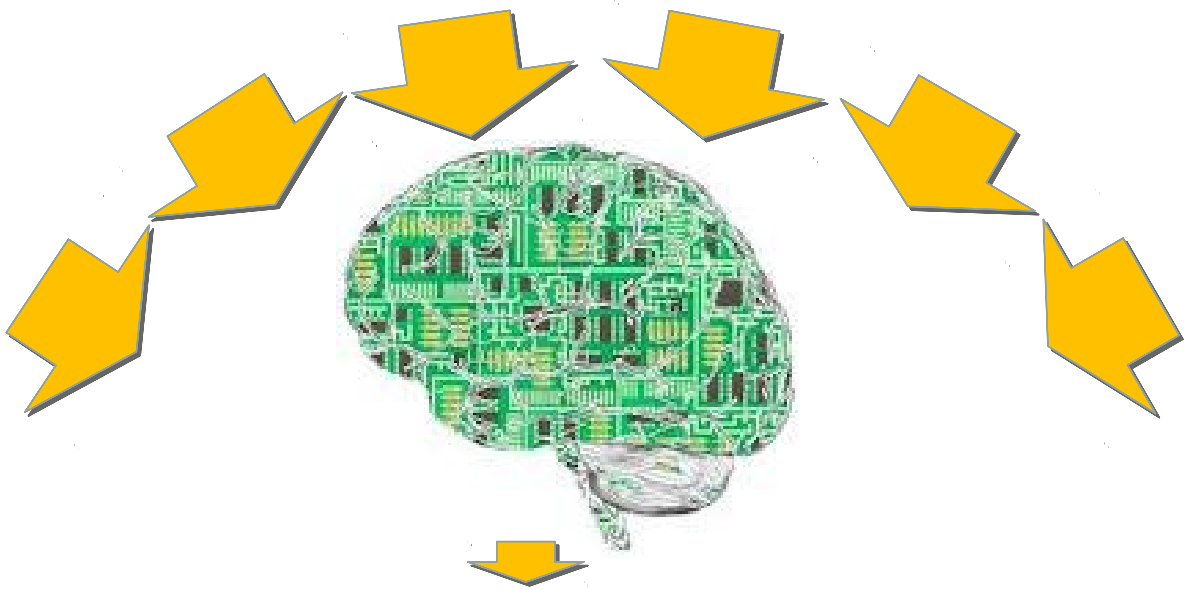
  Output is a single index or an alarm

Bandwidth is reduced by several order of magnitude

Once properly processed, common denominator is: extremely compact output (single index, alarm, …)

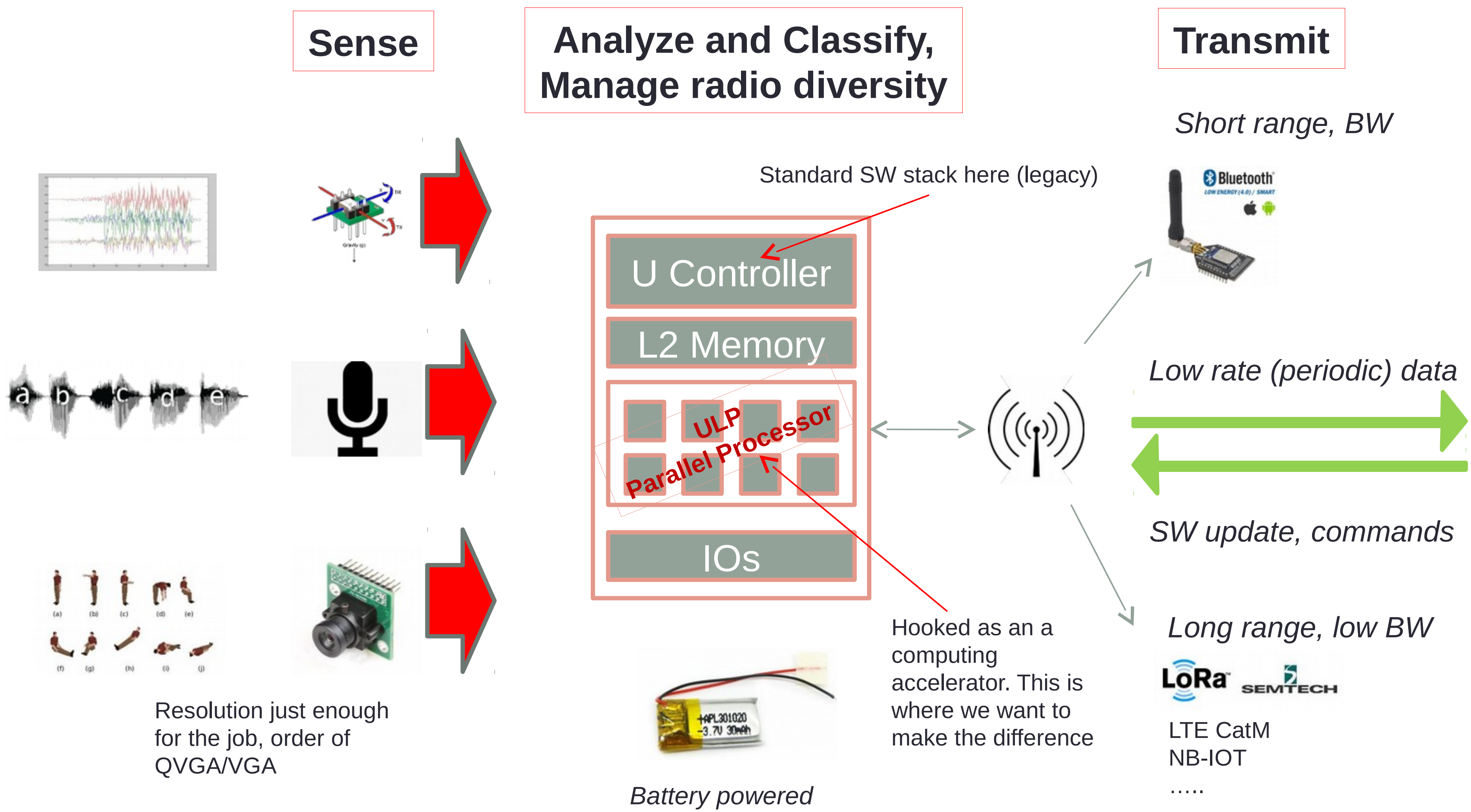
What we want to achieve

Giga/Mega Bytes per second of
incoming *raw data from sensors*



Few (Kilo) Bytes per second of ***outgoing****,*
heavily processed data @ minimum
Joule per operation

System level view



General pattern for content understanding

- Extract descriptors from raw data
  - 2D: Corners, blobs, HOG, DOG, …
  - 1D: LPC coefficients, Cepstral coeffs, …

➡ Usually highly parallel

- Use descriptors to classify data among representative families
  - Machine learning (CNN, SVM, Boost), Bayesian, ….

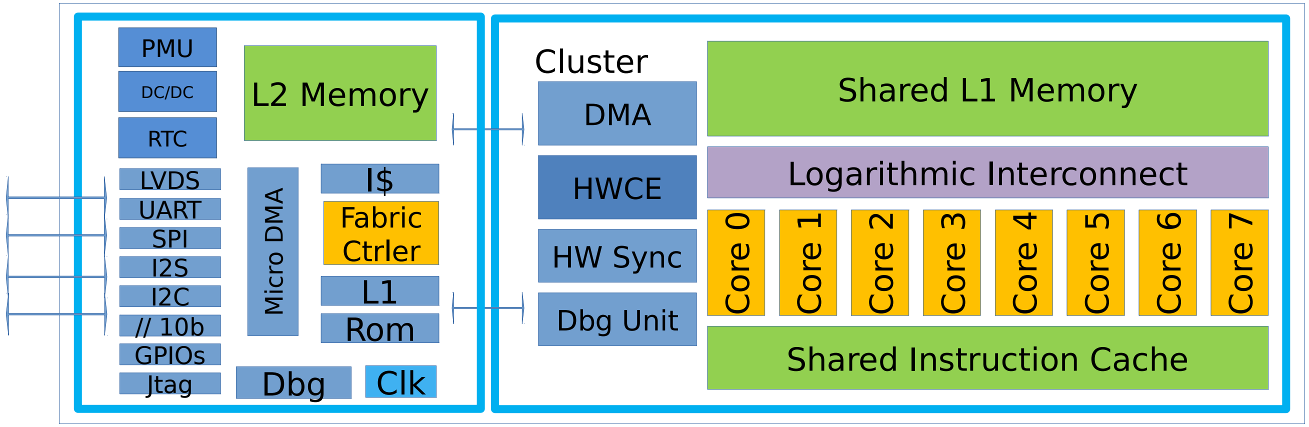
➡ Also highly parallel

GAP8: Ultra Low Power IoT Processor

Performances
- up to 12GOPS
- up to 0.4GOPS @ 1mW,
- up to 40MOPS @ 300uW
- 3 uWatt stand-by power consumption

Architecture efficiency
- Extended Risc-V ISA
- Low contention shared memory 8 +1 core clustered architecture
- Tight synchronization
- CNN based pattern matching engine (HWCE)

HW features
- Smart IOs
- Voltage regulator/DVFS
- RTC
- Secured execution



Low cost processor
- 55nm LP
- 0.5MB L2
- aQFN 84

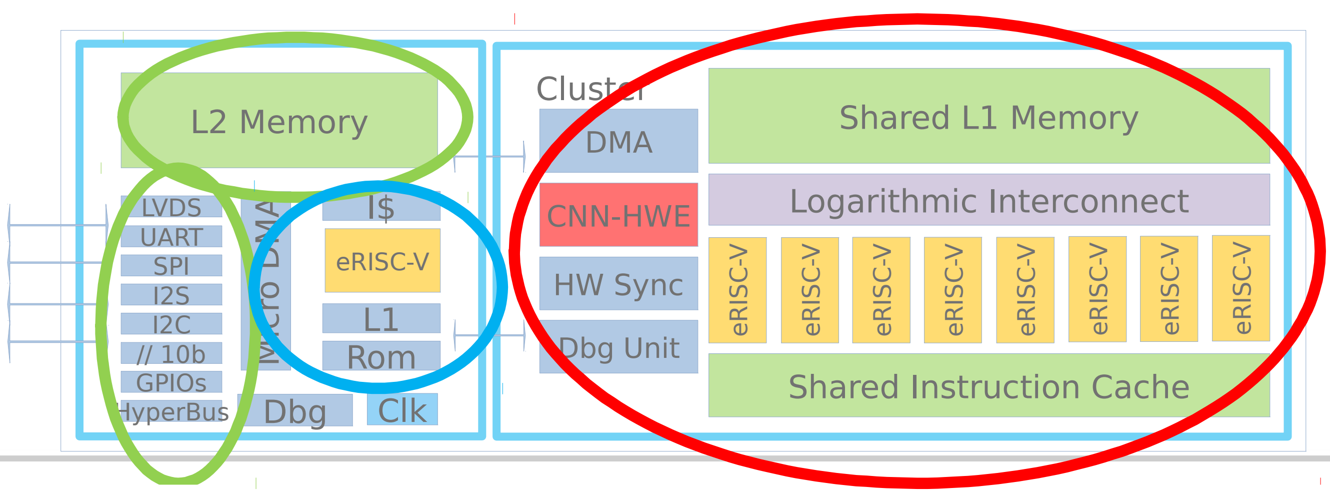
Leveraging open source projects
- Risc-V (Berkeley)
- PULP (ETHZ, UniBo)

Application affinity
- Dominant signal processing part
- Limited memory requirement
- Limited SW legacy

GAP8 has a unique energy efficiency across a very large range of computing power

GAP8 Hierarchical Architecture

monitoring	event qualification, protocol stack, system control	data analysis & classification, SW modem
Smart I/Os voltage regulator & RTC SRAM in retentive mode	extended RISC-V	extended RISC-V efficient 8 core parallelization HW synchronization shared instruction cache CNN HW engine
quasi stand-by	low computing power	high computing power
uWs	mWs	10 to 20 mWs
primary energy consumption		primary energy consumption

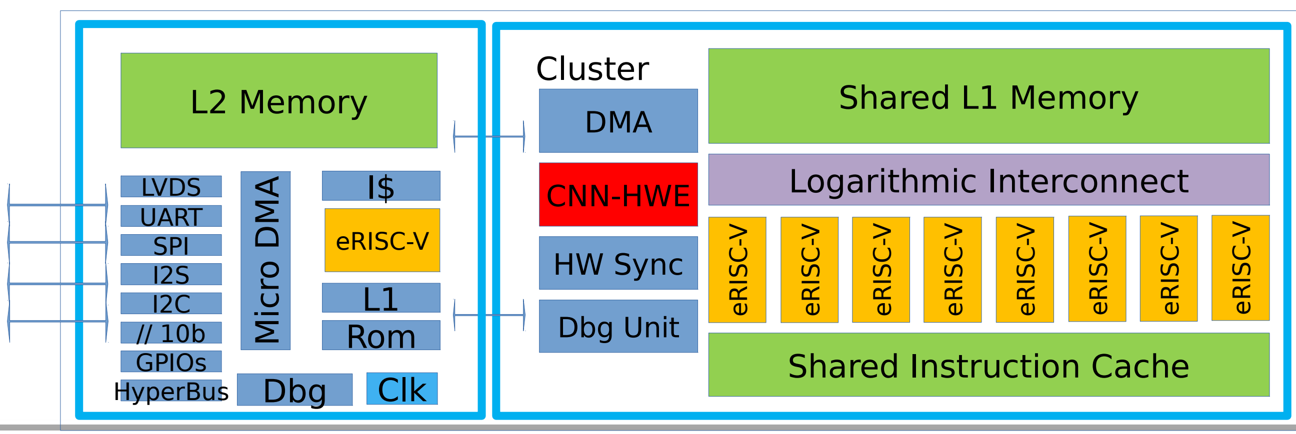


GAP8 architectural energy efficiency gains

data analysis & classification, SW modem

extended RISC-V → 3-5x
efficient 8 core parallelization → 1.4x
HW synchronization
shared instruction cache → 1.5x
CNN HW engine → 5x

overall, in practice on targeted algorithms, typically 20x



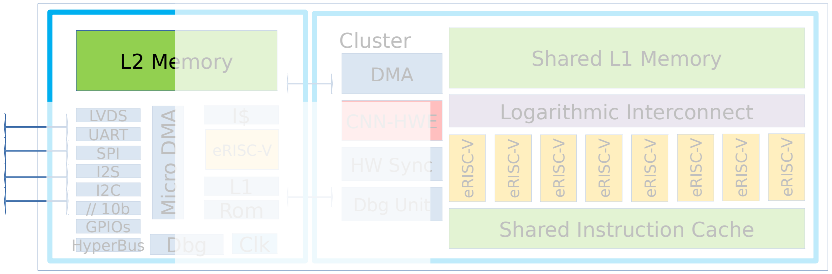
GREENWAVES TECHNOLOGIES

GAP8 Advanced Power Management

MCU sleep mode

uW range

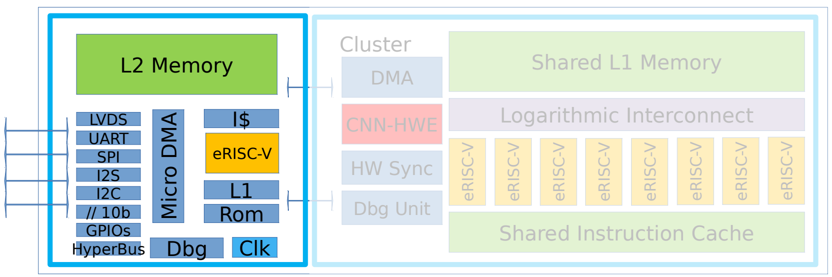
- Embedded DC/DC, low current
- Real Time Clock 32KHz only
- L2 Memory partially retentive



MCU active mode

1 mW range

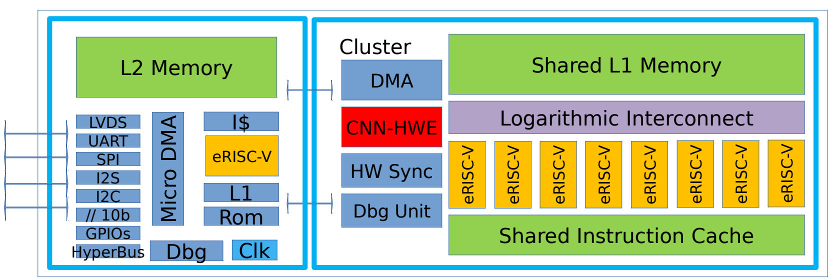
- Embedded DC/DC, high current
- Voltage can dynamically change
- One clock gen active, frequency can dynamically change
- Systematic clock gating



MCU + Parallel processor active mode

10-20 mW range

- Embedded DC/DC, high current
- Voltage can dynamically change
- Two clock gen active, frequencies can dynamically change
- Systematic Clock Gating

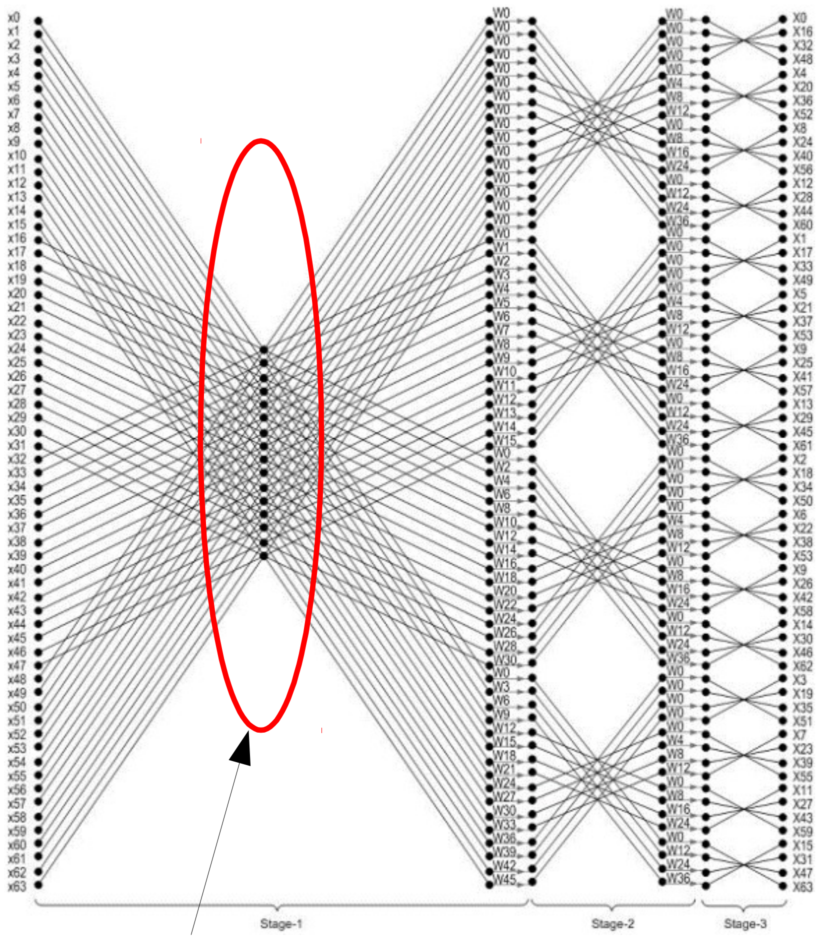


Ultra fast switching time from one mode to another
Ultra fast voltage and frequency change time

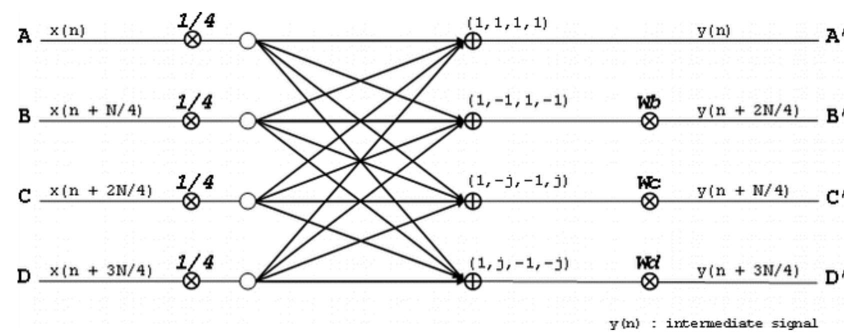
→ Highly optimized system level power consumption

Qualitative data from real life applications

The work horse for radio, sound and vibration: FFT



Radix4 Butterfly

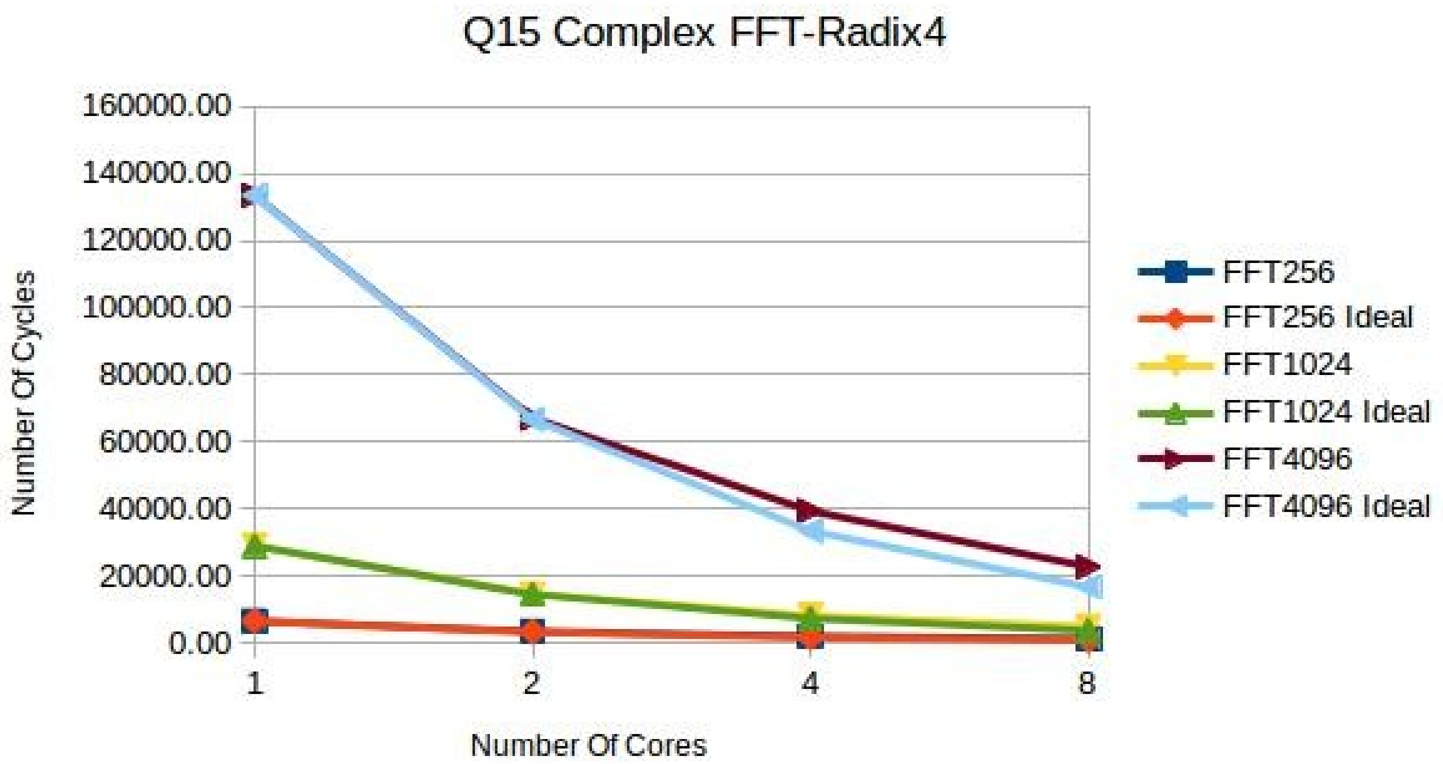


Key operations for performance

- Complex Multiplications
- Complex Rotations
- Post modified accesses
- Vectorial operations

All these butterflies are evaluated in parallel

The work horse for radio, sound and vibration: FFT



Number of operations (*,+,>>,Ld/St)

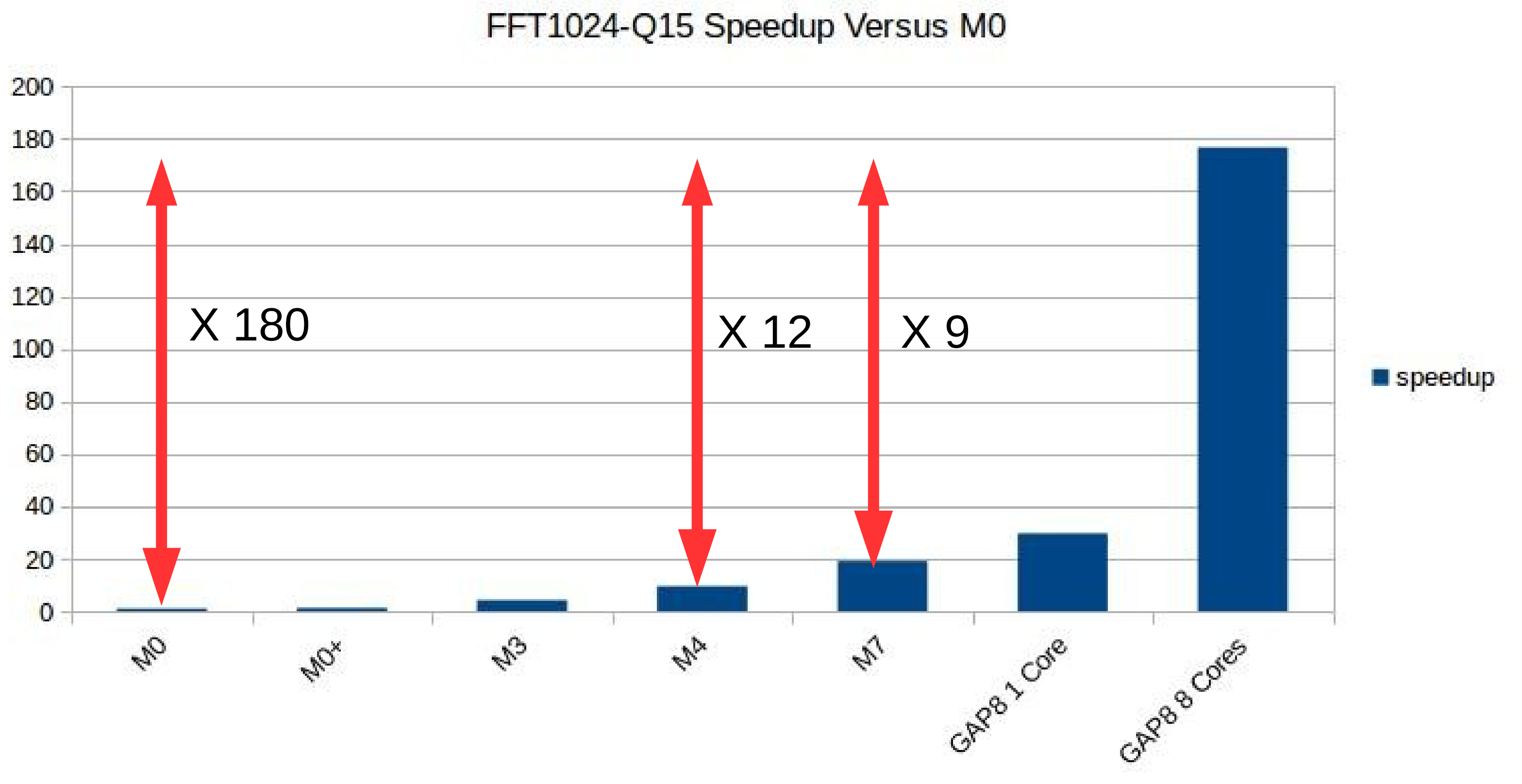
FFT 256	FFT 1024	FFT 4096
11264	56320	225280

Number of Cycles running on 8 cores

FFT 256	FFT 1024	FFT 4096
1167	4842	22710

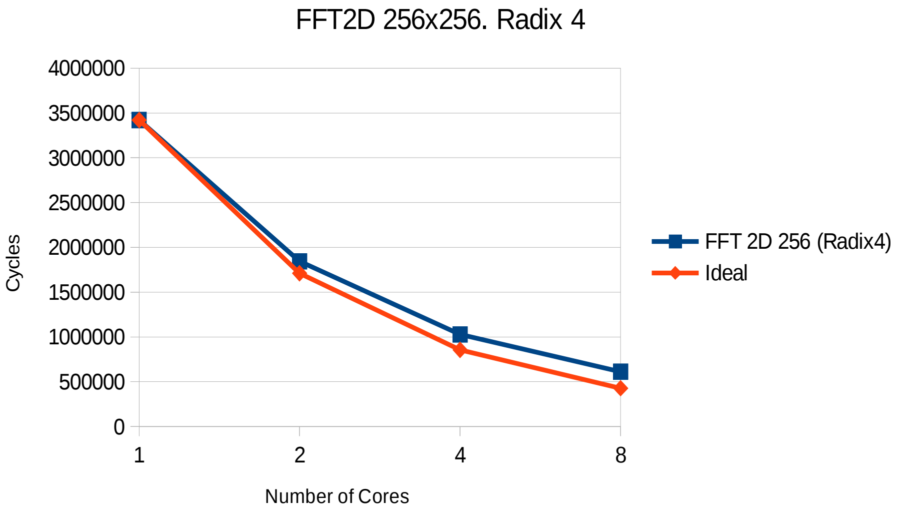
Ideal means perfect scaling, perf on N cores = Perf on 1 core / N

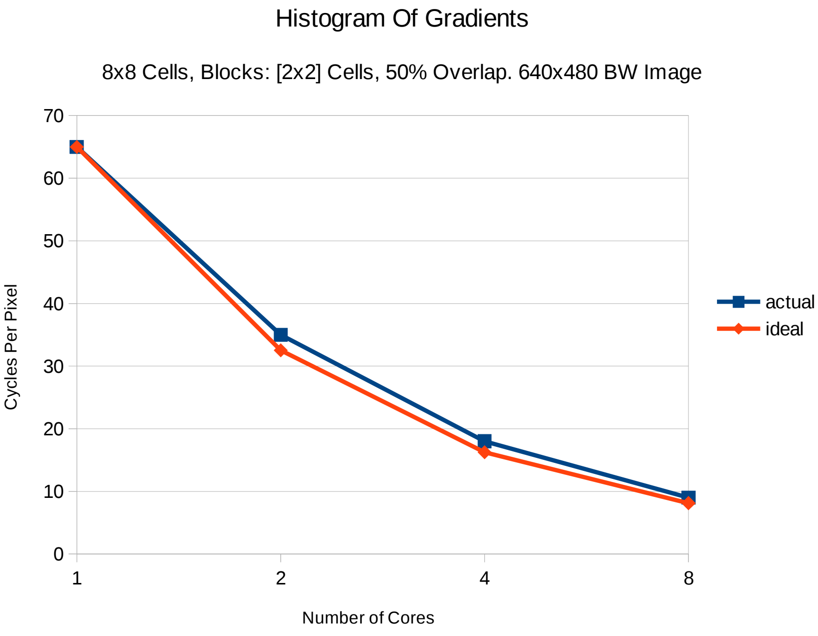
The work horse for radio, sound and vibration: FFT



ARM FFT1024 Q15 Data are with CMSIS optimized library

Visual Localization: FFT2D + HOG



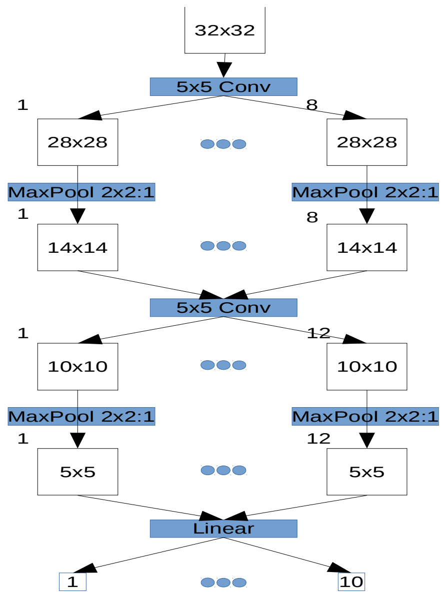


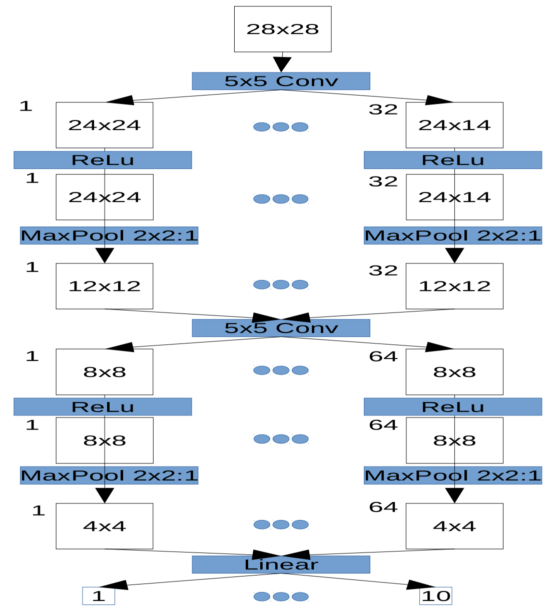
1 384 000 cycles per image

589 000 cycles per image

We need only 2 MHz per image

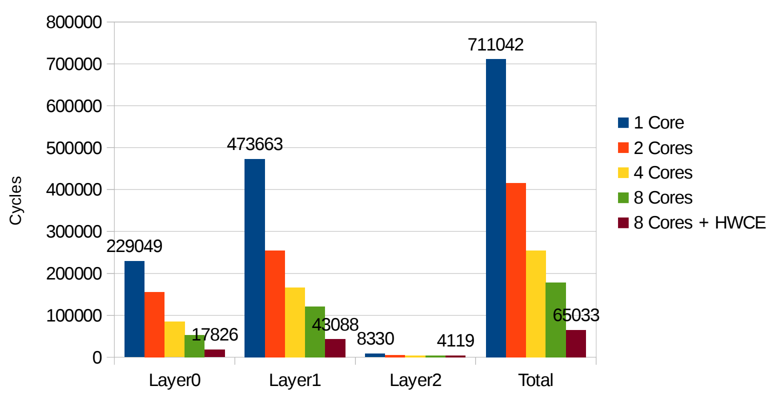
CNN based Image Classification

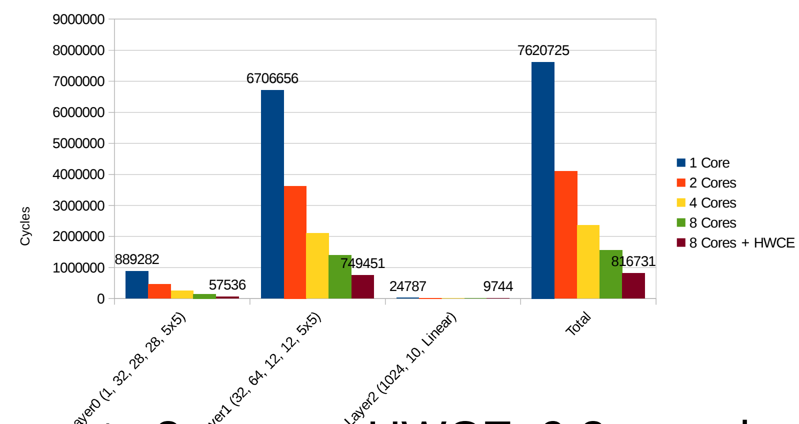




CIFAR10

MNIST





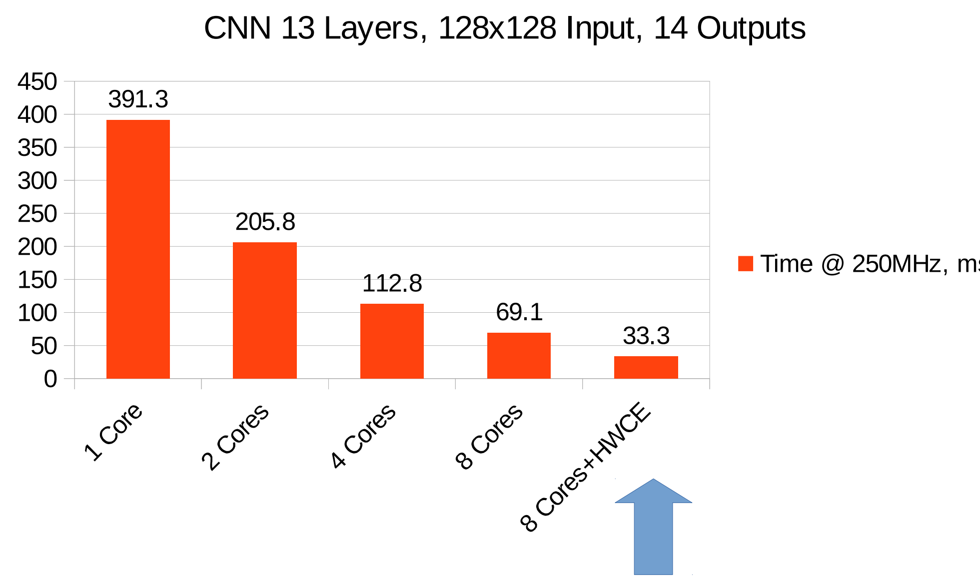
1 core to 8 cores + HWCE: 10.9 speedup

1 core to 8 cores + HWCE: 9.3 speedup

CNN based Image Classification

Global Normalization	*128 x 128*
Contrastive Normalization	*128 x 128*
Convolution 3x3	*1 x 128 x 128 => 32 x 126 x 126*
ReLU	*32 x 126 x 126 => 32 x 126 x 126*
MaxPool 2x2	*32 x 126 x126 => 32 x 63 x 63*
Convolution 3x3	*32 x 63 x 63 => 32 x 61 x 61*
ReLU	*32 x 61 x 61 => 32 x 61 x 61*
MaxPool 2x2	*32 x 61 x 61 => 32 x 30 x 30*
Convolution 3x3	*32 x 30 x 30 => 32 x 28 x 28*
ReLU	*32 x 28 x 28 => 32 x 28 x 28*
MaxPool 2x2	*32 x 28 x 28 => 32 x 14 x 14*
Linear	*6272 x 64 => 64*
ReLU	*64 => 64*
Linear	*64 x 13 => 13*

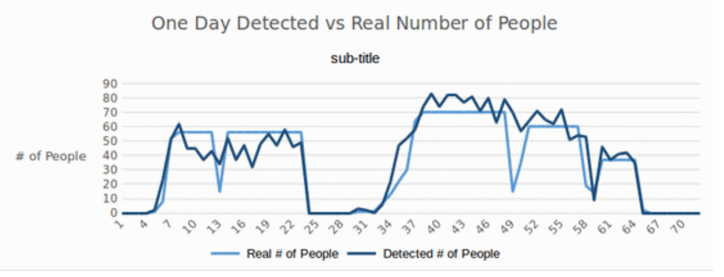
Trainable Par:	*421 263*
Neurons:	*1 511 904*

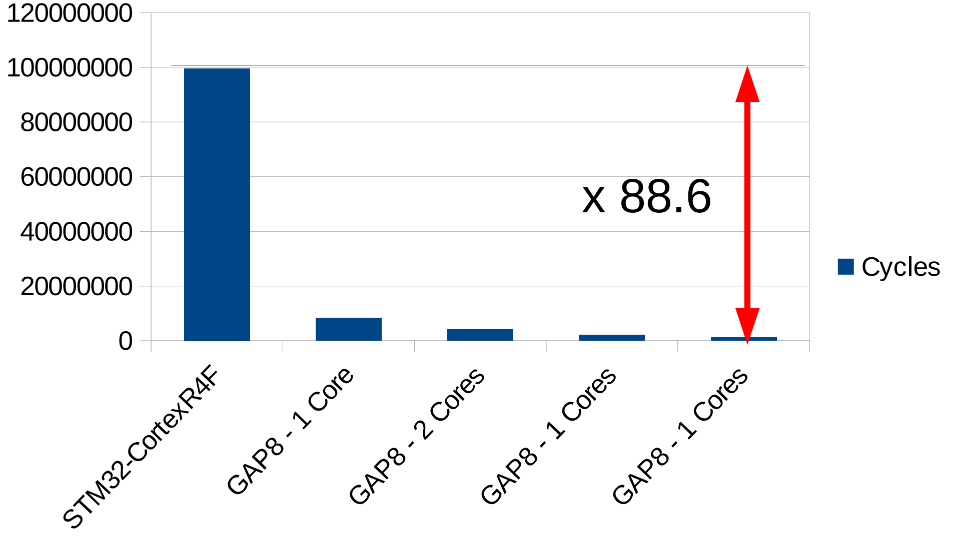


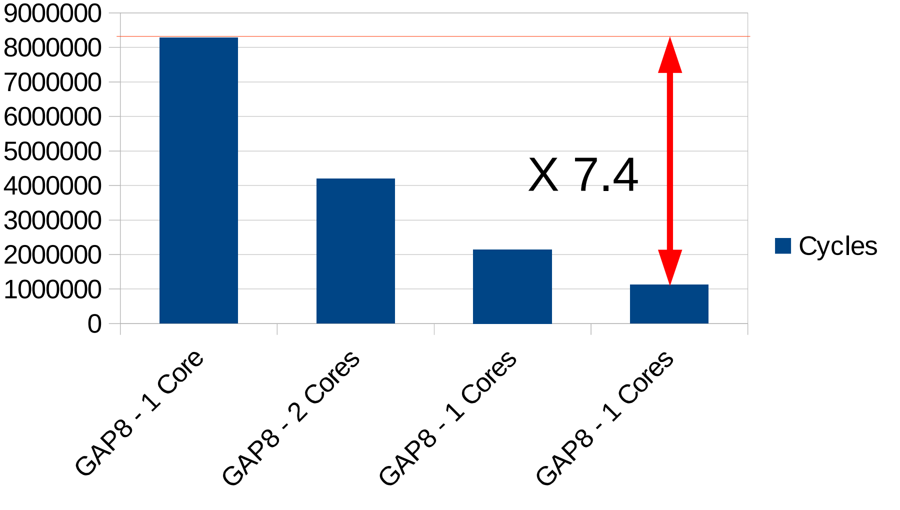
33ms per image

People Counting

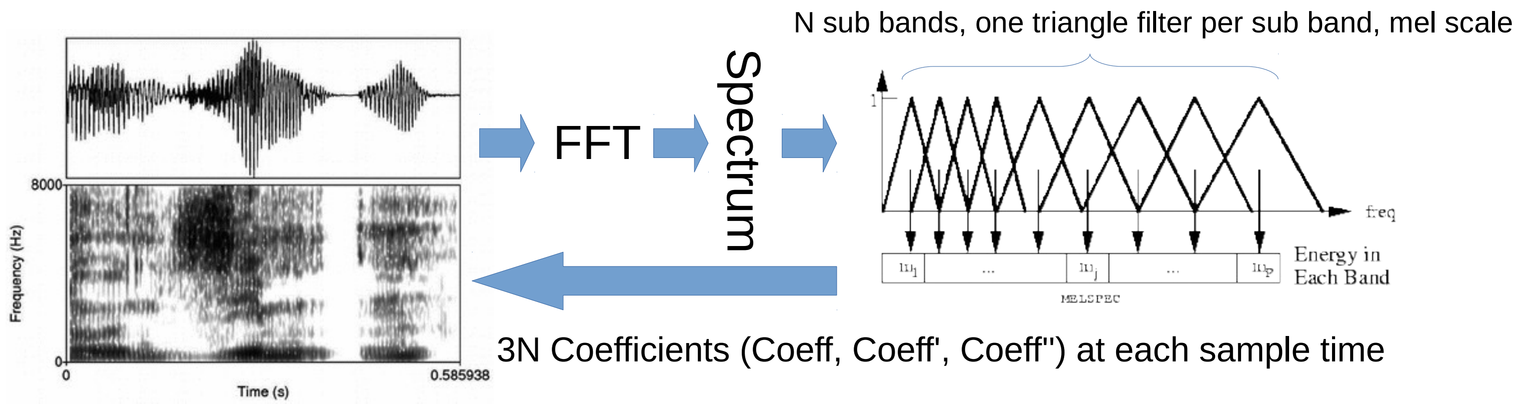
- Filtering + Difference of Gradient + SVM-RBF
- Open Space. Accuracy: approx 90%
- 1 Image every 3 minutes => 10 years on a battery

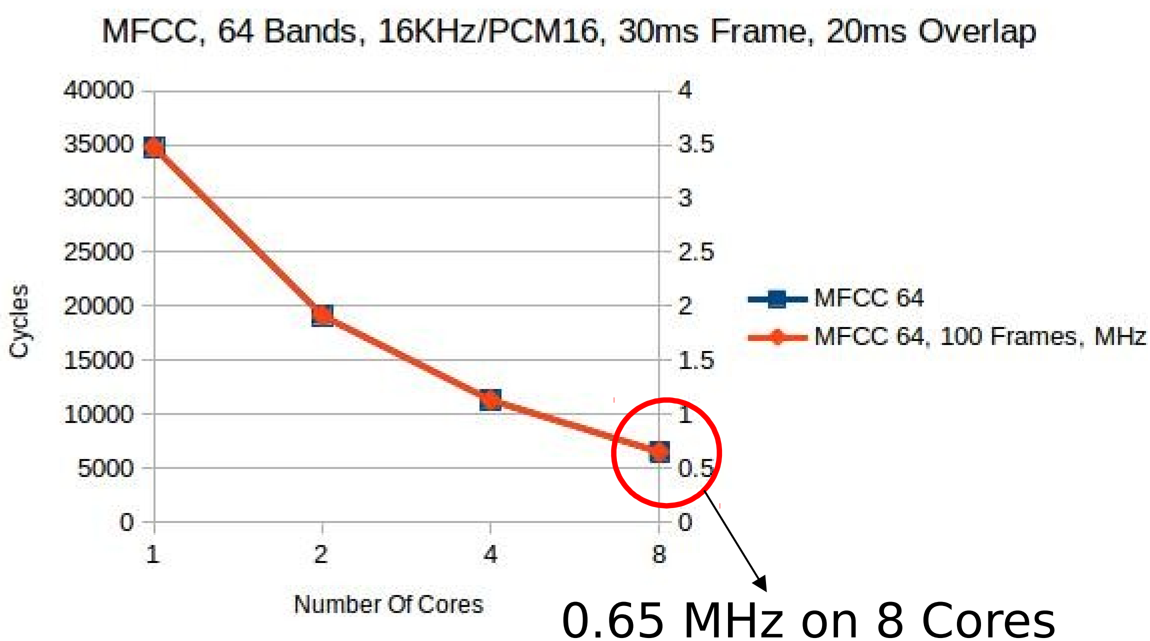




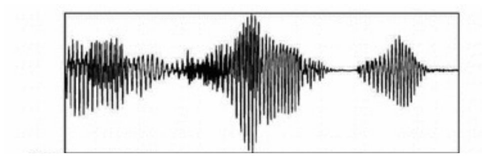


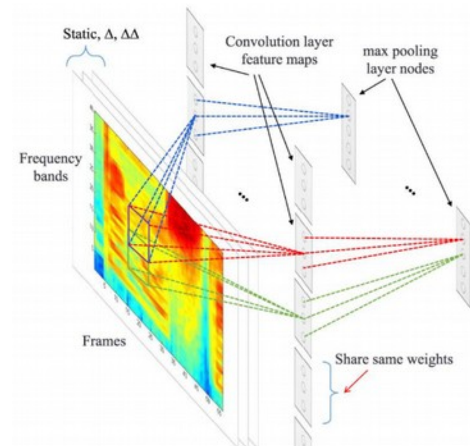
Audio Processing





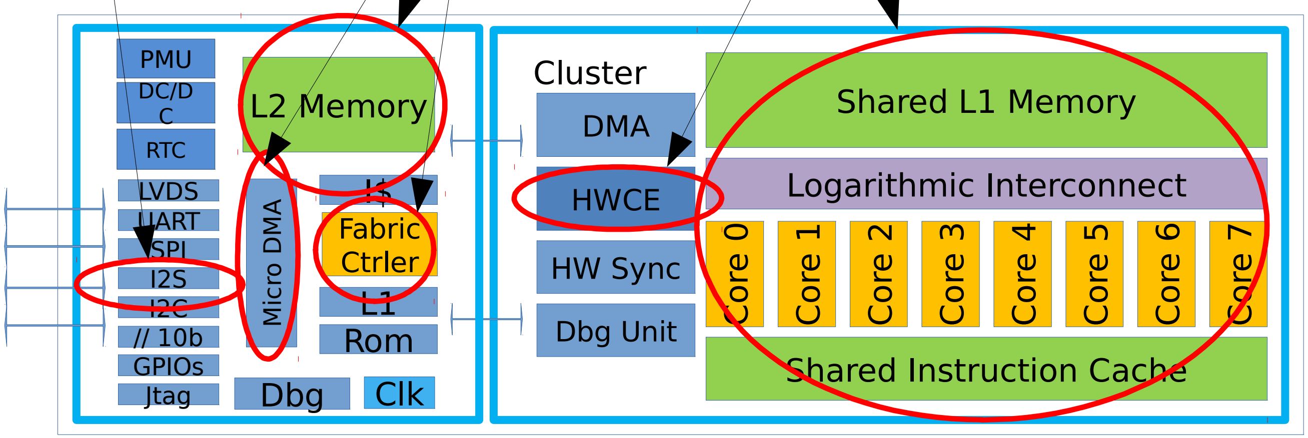
Hierarchical Power Processing?





Gradually turn on resources

Sound ? → Phonem ? → Word ?



Conclusion

- GAP8 bridges the gap between ultra low power MCU and multi-core processor:
  - Smart sensing on data rich sensors achievable within tinny power budget: uW in Idle, mW in micro controller mode, 5-20mW in number crunching mode : Few Mops to up to 12Gops
  - Low cost bill of material

- GAP8 agile power management architecture combined with IOT low duty cycling is a perfect fit for FDSOI process