heterogeneous
hɛt(ə)re(ʊ)ˈdʒiːnɪəs

adjective
Diverse in character or content.

Hetero – from the Greek, meaning ‘other’
gen – a combining word that signifies “that which produces”
-ous - a suffix that has the general sense “possessing, full of” a given quality

Heterogeneous = A system full of diverse parts that produces much
The idea isn’t new, but

The time is now!
It’s not about my **GPU** is better than your **CPU**

Heterogeneous Computing is about:

**Performance, Cost & Power advantage**

*Move compute where you need it*
Today: Same SoC, separate memory architectures

Old techniques reaching breaking point & falling short of emerging needs

Designed as two separate worlds
- Doubling major SoC costs, i.e. memory
- Increasing latencies, in-efficiency & power consumption
More, more and Moore…

User experience & advanced software applications are demanding hardware upgrade

and industry’s commoditized transistors are in wrong arrangements to deliver
Let’s look at two practical examples

Just a few that are driving the need for heterogeneous devices

Automotive
Heterogeneous system
save systems costs

Internet of Things
Power Consumption vs
Bandwidth Costs
Automotive

- Transparency Market Research predict automotive electronic market $18 bn
- Just how many CPUs, GPUs & different OSs are there in a modern car?
- Will automotive industry sustain these increasing electronic costs?
- And security? So many attack surfaces
- How to deal with enhanced compute requirements in terms of power consumption and functional safety?

Heterogeneous system save systems costs

- In electronics (classic integration play)
- In reduced components, reduced cables & connectors, reduced weight leading to improved mileage
- In reduced logistics & spare parts
- In reducing production costs

<table>
<thead>
<tr>
<th>Engine management</th>
<th>Steering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission</td>
<td>Breaking, Fuel system</td>
</tr>
<tr>
<td>Battery management</td>
<td>Audio, infotainment</td>
</tr>
<tr>
<td>Doors, locks, lighting</td>
<td>Telematics &amp; many more</td>
</tr>
</tbody>
</table>
Internet of Things (IoT)

*Power consumption vs bandwidth costs, where to place the compute?*

- IoT businesses centre around running services → Opex must be low
- Edge device often try to optimize battery life, looking to off-load compute
- Streaming data can result in costs that do not necessarily scale well for business
  - Cloud companies typically charge for bandwidth, storage & processing
- Need silicon capable of balance resources & costs over the network
Heterogeneous benefits start with shared memory

Shared memory for CPU & GPU

- Increases performance
  - Eliminates copies
  - Reduces overheads (pass pointers)
- Lower latency from GPU to memory
- Power efficient design
- Enables architectural innovations
  - Move the processing to where you need it in the system (e.g., at edge, rather than all in the cloud)
- Scalable architecture
  - From edge to access point to data centre and through to mobile
- Allows quick offloading tasks to take advantage of the different architectures & capabilities of different hardware
Security & Safety

Is there a greater imperative in the industry today?

Security by separation through true hardware virtualization

- Sandbox functionality
- Mixed OSs on same silicon
- Isolate deterministic functionality
- Update without undermining existing software
- Avoid software contamination
- Accelerate time-to-market
Functional Safety – design and technology combo

- Safety requirements for IP Design (IP Vendor)
- Safety requirements for SoC Design (SoC Vendor)

SoC Safety goals

- SoC Safety concept

SoC Safety analysis

- SoC Integration
  - SoC Safety analysis report
  - Functional safety audit
  - FMEDA report
  - Safety Cases
  - Safety Plan
- SoC Verification
  - Validation of requirements
  - Verification plan
  - Verification & Testing
  - Verification plan
- SoC Design
  - Design Spec
  - Design Spec
  - Safety Plan
  - Safety Plan
- DIA: Development Interface Agreement
  - Concept
  - Concept
  - Concept
  - Concept

IP Design

- IP Design
  - Design Spec
  - Design Spec
  - Design Spec
  - Design Spec
- Verification & Testing
  - Failure analysis
  - Functional safety audit
  - FMEDA analysis
  - QA Audit

Validation of requirements

- Validation of requirements
- Validation of requirements

IP Verification and Testing (IP Vendor)

SoC: Safety Concept

- Safety assessment plan
- Concept analysis
- Safety Plan
- Concept
- Concept
- Concept
- Concept

Validation of requirements

- Validation of requirements
- Validation of requirements

SoC Integration, Verification and Testing (SoC Vendor)

CPU Safety concept

- HW/SW Verification
- IP Design
- Verifications and Testing
- SRS: Safety Requirement Specification
- Development Agreement
- DIA: Development Interface Agreement
- Validation of requirements
- Validation of requirements
- Validation of requirements
- Validation of requirements

IP verification and Testing (IP Vendor)
Customer project: MIPS distributed development

- Hazard analysis (HARA)
- Allocation of functional safety requirements to E/E system (Based on Applications)

Safety requirements, safety concept, hardware, software

Allocation of technical safety ASIL requirements to HW and SW products

ISO26262 Part 1, 2, 5, 8, 9

- ISO26262 Part 1 to Part 10

- SRS: Safety Requirement Specification
- TSR: Technical Safety Requirements
- DIA: Development Interface Agreement

© Imagination Technologies
Tokyo June 2017 13
SOI and ISO 26262

- Same letters – different order - perfect combo

- SOI delivers low power consumption
  - ADAS devices require compute redundancy in low power environment

- SOI is less sensitive to radiation
  - A key advantage to simplify complex designs and minimizing soft errors

- SOI eases mixed design due to higher dielectric isolation
  - This increases overall robustness and reliability of the design
**I6500 Warrior Class**

Heterogeneous Inside & Out

- Multi-threaded, multi-core, multi-cluster scalable CPU for heterogeneous compute
- Coherently implements CPU cores within a cluster (‘Heterogeneous Inside’)
- + Other CPU clusters, GPUs & accelerators on-chip (‘Heterogeneous Outside’)

The I6500 CPU at heart of heterogeneous coherent processing clusters in Mobileye’s next-generation EyeQ®5 SoC

The central computer performing sensor fusion for Fully Autonomous Driving (FAD) vehicles
The Secrets of the I6500 CPU

1. Picking up multiple threads
2. Sharing a common view
3. Maintaining cache coherency between processors
4. Mixing CPUs with dedicated accelerators
5. Reducing data transfer cost
6. IO Coherency Units
7. Designed with functional safety in mind
Conclusion

*Heterogeneous compute is coming*

Are your SoCs ready?

- Advanced driver assistance systems (ADAS)
- Autonomous vehicles
- Networking
- Drones
- Industrial automation
- Security
- Video analytics
- Machine learning