# Theme 4

## Platforms, Architectures and Demonstrators

Steve Furber and Geoff Merrett
PRiME Industry Day
London, 30 June 2015

www.prime-project.org

# What is Theme 4?

- Platforms and Architectures
  - A range of platforms to support **PRiME**'s research
  - The homogeneous SpiNNaker platform
  - Does a manycore architecture = (*n* * multicore)?
- Demonstrators (*and Applications*)
  - To showcase **PRiME** research

# Platforms & Architectures

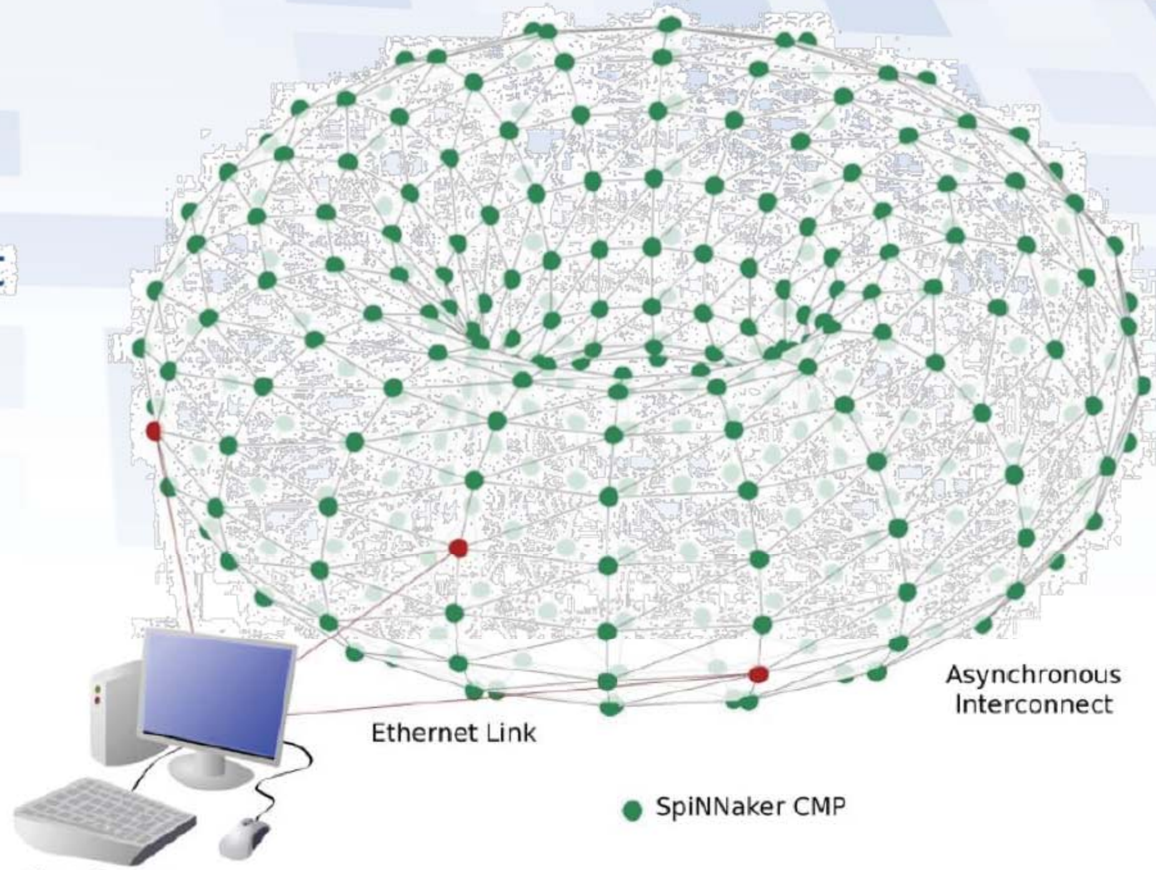
# Platforms and Architectures

- Heterogeneous Many-Core Platforms
  - Developed a Linux build for Cyclone V which meets **PRiME**'s OS requirements for the heterogeneous platform.
  - Integrating the heterogeneous platform with Theme 3 instrumentation for slack time measurement.
- Homogeneous Many-Core Platforms
  - Understanding of capability of existing COTS platforms
  - SpiNNaker: Implementing run-time mechanisms

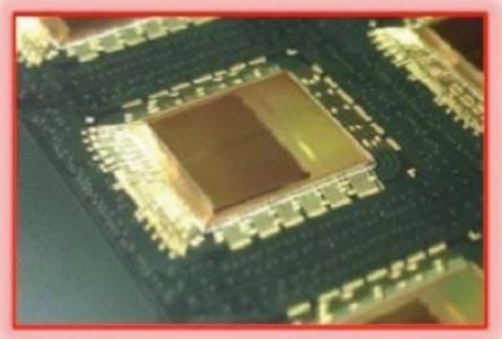



# The SpiNNaker project

- A million mobile phone processors in one computer
- Able to model about 1% of the human brain…
- …or 10 mice!

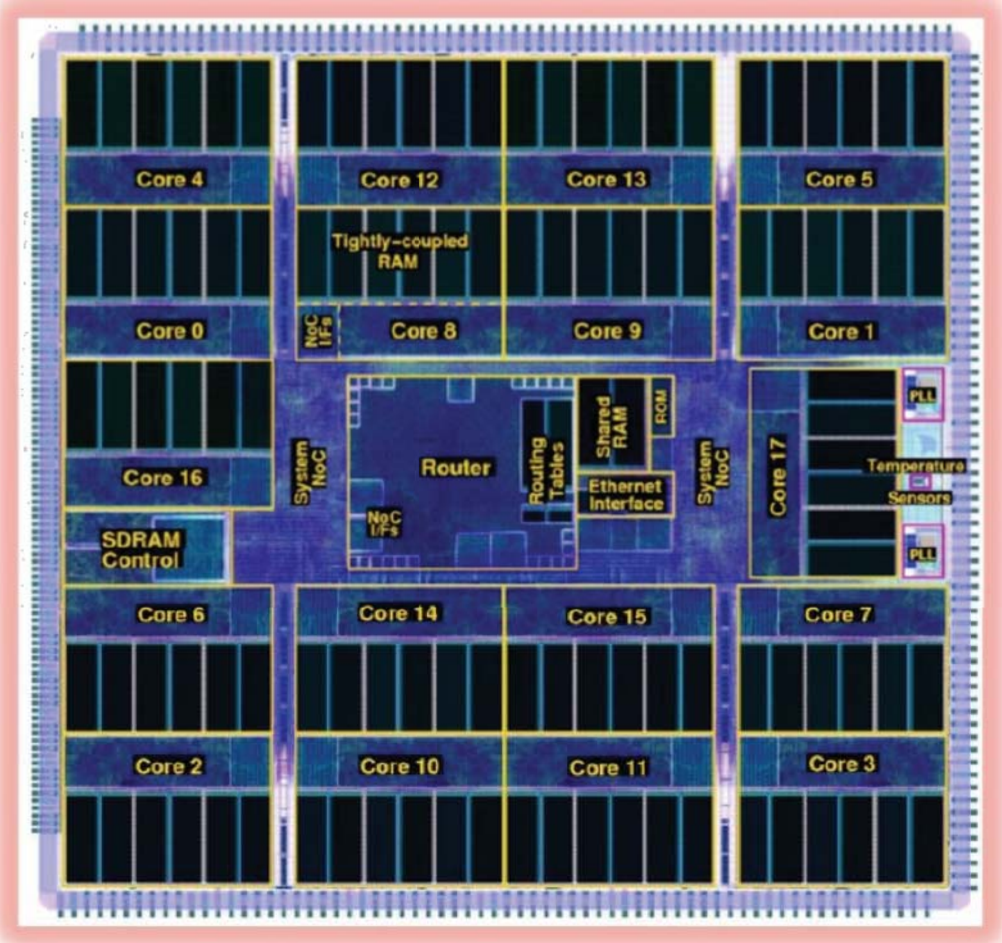


# The SpiNNaker project




Mobile DDR SDRAM interface

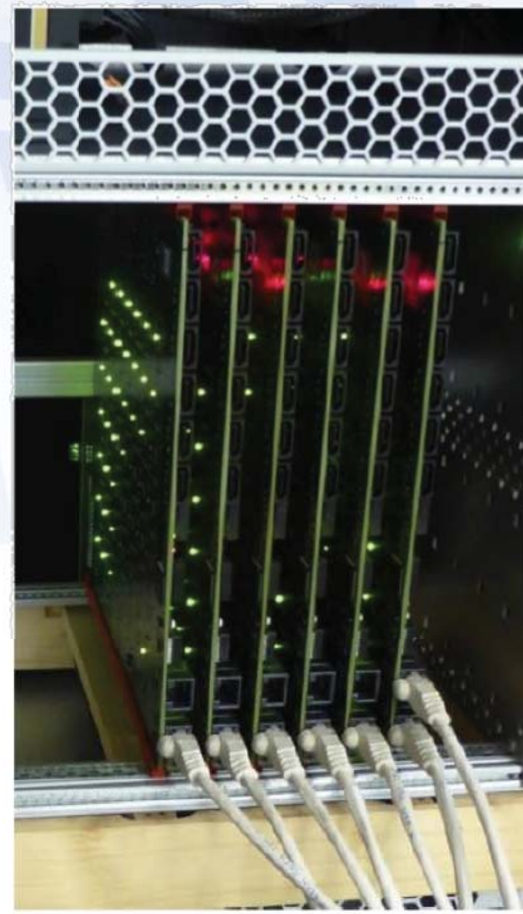
Multi-chip packaging by UNISEM Europe



# The SpiNNaker project

SpiNNaker
Biologically Inspired Massively Parallel Architectures

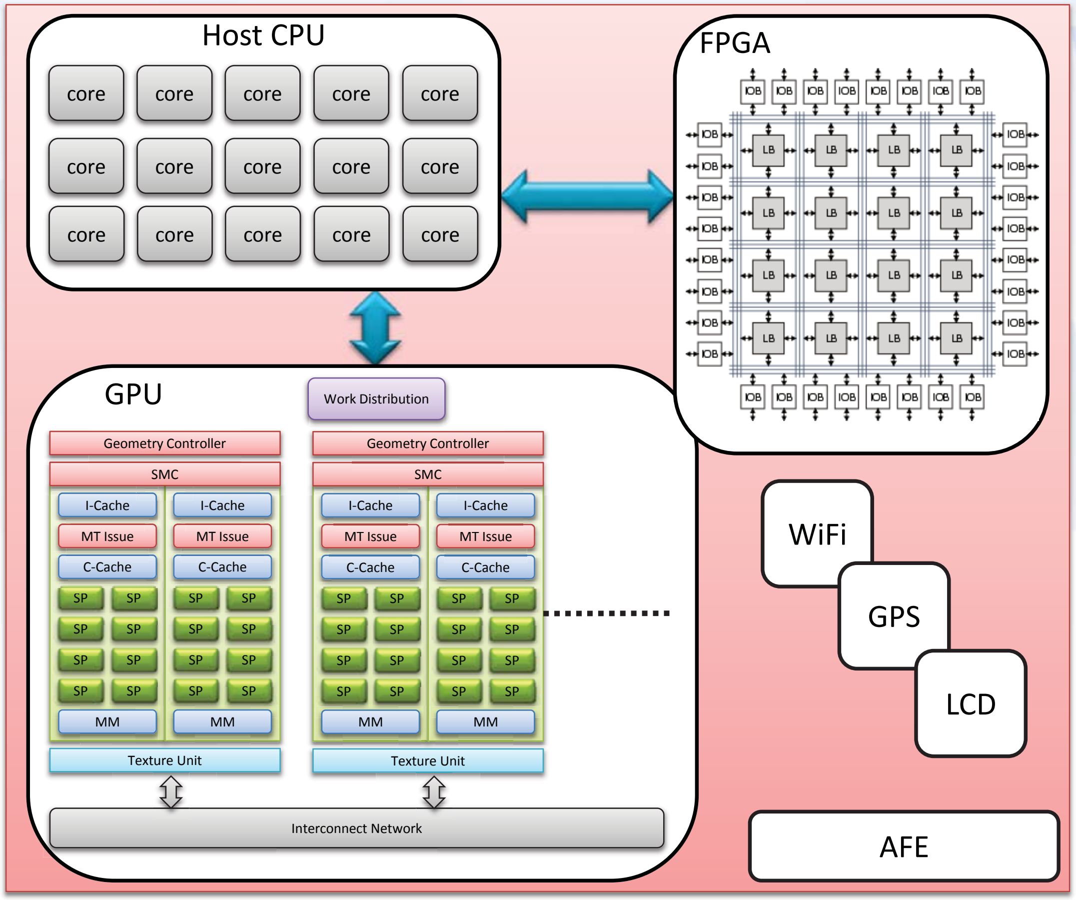
PRiME
www.prime-project.org

# The SpiNNaker project

*Can run-time fault recovery and load-balancing be compatible with delivering continuous real-time performance on a system such as SpiNNaker?*

# Future Many-core Architectures




- Some architectures can 'approximate' this (and may appear as we progress).
- For others, we can use simulation.
  - Gem5
  - VLAB

# Applications & Demonstrators

UNIVERSITY OF Southampton

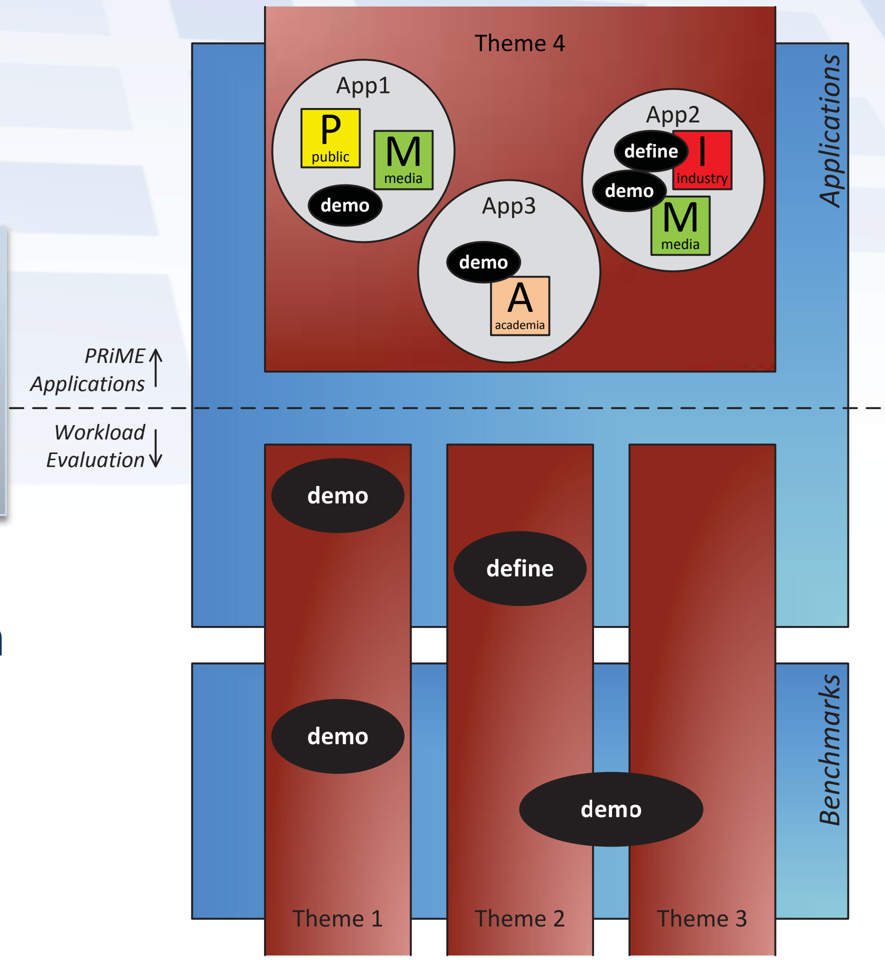
PRiME
www.prime-project.org

# Applications & Demonstrators

*“An unfortunate computer science tradition is we build research prototypes, then wonder why applications people don’t use them.”*

Asanovic et al., “A view of the parallel computing landscape,” *Commun. ACM*, vol. 52, iss. 10, pp. 56-67, 2009.

- Workloads needed to:
  - Enable individual research efforts to demonstrate benefits/advantages. (***evaluation workloads***)
  - Create engaging ***showcase applications***, focussing research on common drivers/goals, and showcasing cross-theme outcomes.
- Workloads used to showcase research (***demonstrate***), or inform research challenges (***define***).
- PRiME is not trying to invent many-core killer applications



# Applications in PRiME

- Applications and Demonstrators draft report produced, and workshop held to discuss
- Applications and Demonstrators wiki growing, aiding cross-theme/institution involvement.
- Experimentation with identified benchmarks on existing platforms (ODROID/Xeon Phi etc)
- Cross-theme demonstrator produced

**PRiME: Applications and Benchmarks**

Geoff V. Merrett (Theme 4, Uni. of Southampton)
*Version 0.1, 26 March 2014*

This report (which is in draft and a work-in-progress) provides a necessary understanding applications and benchmarks, or *workloads*, relevant to PRiME research. Workloads will be used both *drive* and *demonstrate* research, and it is hoped that this report will identify the definitive cros theme/institution *PRiME Applications* whilst also providing a *Workload Evaluation* to inspire an inform individual researchers. PRiME has particular requirements on workloads, including scalabilit power, reliability, open source, and relevance. Recent trends have emerged in predicting an categorising future workloads, including Intel's RMS, Berkeley's dwarfs/motifs (and other views), an approximate computing. The report presents an initial investigation into a scoping exercise available benchmarks that might be suitable for PRiME. Additionally, the early outputs of application scoping exercise are presented, considering a range of different application domains an algorithms. While findings are premature, a number of suitable workloads are evident for PRiME.

Note: this document will evolve alongside PRiME's *Applications and Benchmarks Wiki*, available https://github.com/PRiME-project/main/wiki/Benchmarks-and-Applications

**Workload Parameters and Uses in PRiME**

This report is a draft working document, which will eventually provide (PRiME deliverable 4.3a January 2015) a necessary understanding of applications and benchmarks, or *workloads*, fo demonstrating, exploiting and driving PRiME research[1]. Applications are critical to the success PRiME, in order to both demonstrate and focus/drive research, hence closing the gap betwee emergent application opportunities and the design community [1]. It is envisaged the final report which will subsequently be an evolving document, routinely updated and developed – will be used to:

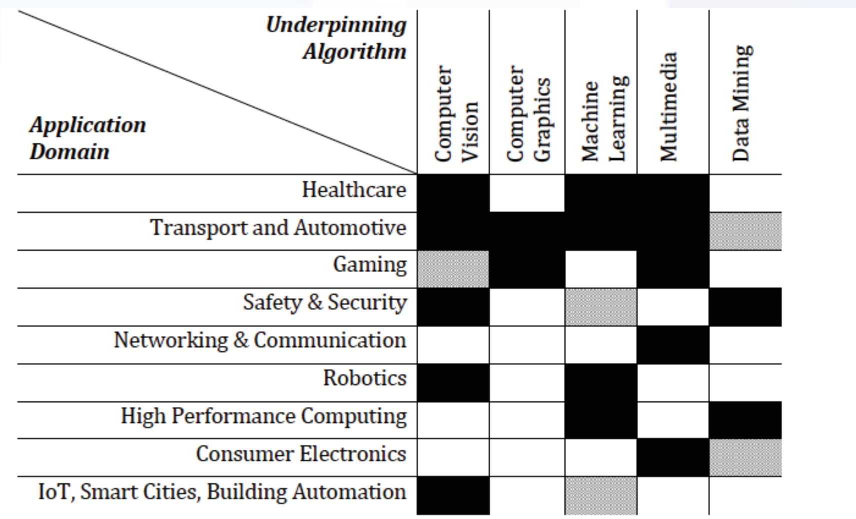
1. **PRiME Applications**: identify a (limited; 2-4) set of applications that will pull together PRiM research to showcase outcomes to different stakeholders – for example the public, media, industr and academia. As such, they focus research across the consortium to a common set of drivers/goals
2. **Workload Evaluation**: provide a clear evaluation of other workloads so that PRiME researche can identify those that would be useful to demonstrate/evaluate specific research within the thematic areas.

In both cases, it is envisaged that workloads in PRiME will likely both *demonstrate* (where th application is selected to showcase a piece of academic research), and *define* (where the applicatio defines research challenges to be overcome) research. This is illustrated in Figure 1.

*“An unfortunate computer science tradition is we build research prototypes, then wonder why applications people don't use them.”*
*Source: [*

# Applications and Workloads

- Underpinning Algorithms:
  - Computer vision
  - Computer graphics
  - Machine learning
  - Multimedia
  - Data mining




- Application Domains:
  - Healthcare
  - Transport and Automotive
  - Gaming
  - Safety and Security
  - Networking and Comms
  - Robotics
  - High Performance Computing
  - Consumer Electronics
  - IoT, Smart Cities, Smart Buildings

# Requirements of an Application

- Scalability: **priMe**
  - Should scale to make use of available computing resources.
  - Highly parallel (existing HPC?) or multiple concurrent applications.
- Power: **Prime**
  - Aligned with a need and capability of reducing power/temperature.
- Reliability: **pRime**
  - Tolerating errors/imprecision (approximate computing)
  - Meeting reliability requirements (critical systems)
- Getting 'Bang for Buck'
  - Available/open source code, already parallelised
  - Suitable for execution on heterogeneous platforms
- Relevant to stakeholders

# Showcase Demonstrator?

## Current thinking

*Application domain for stakeholder 1*

Automotive

*Application domain for stakeholder 2*

Consumer Electronics

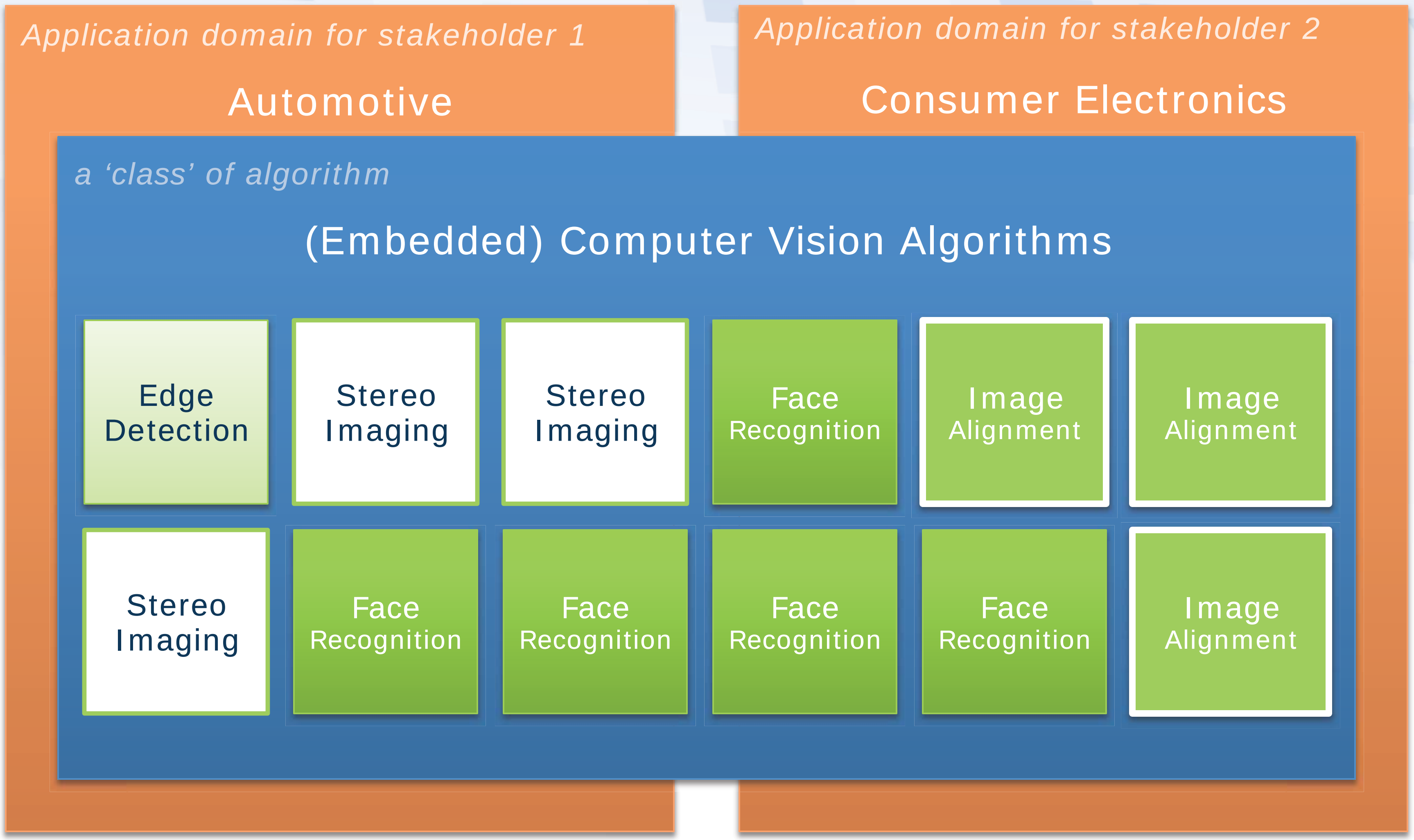
*a 'class' of algorithm*

(Embedded) Computer Vision Algorithms

- Edge Detection
- Face Recognition
- Stereo Imaging
- Image Alignment

# Showcase Demonstrator?

## Current thinking

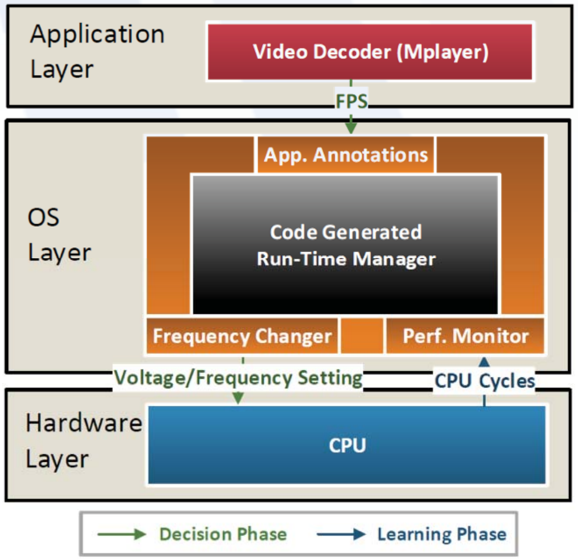


# Today's Demonstrators

PRiME
www.prime-project.org

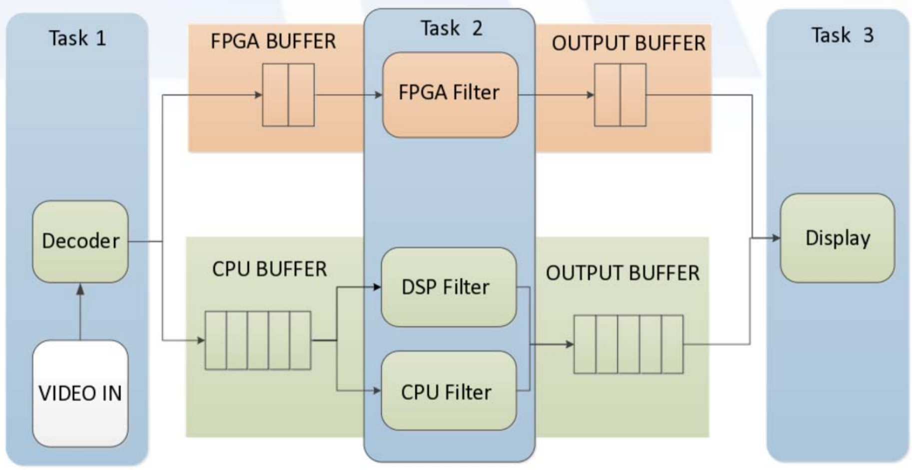
# 1. Formal methods for design & verification of RTM software & automatic RTM code generation

- Formal methods are mathematical techniques for the specification and development of complex systems which can be used to thoroughly verify the properties of a system, increasing the reliability and robustness of the design. Event-B is a formal method for system-level modelling and analysis which is used to create a formal model of the Run Time Management (RTM) software. This RTM formal model can be automatically translated into executable code to be run on the hardware, reducing the effort of hand-coding the implementation, and is portable across different architectures and Operating Systems.



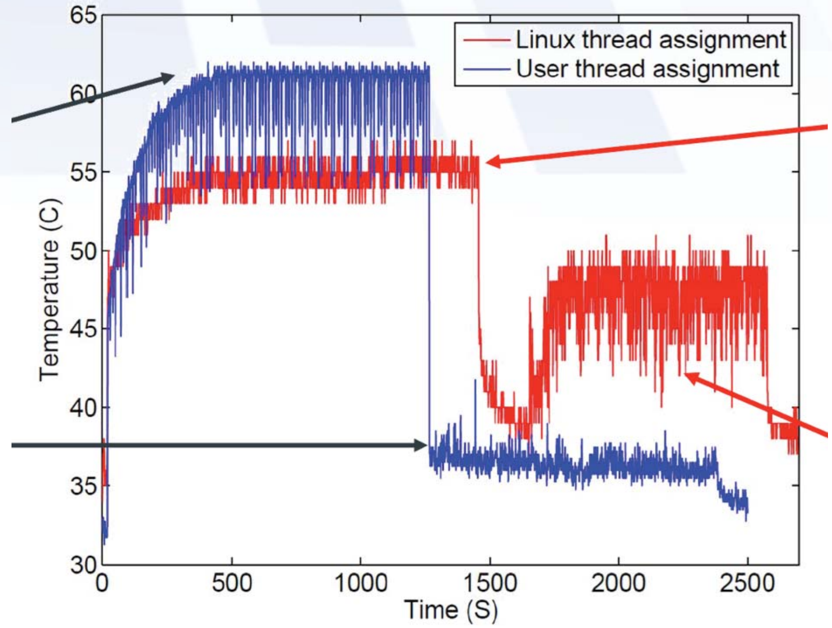
# 2. Adaptive energy optimisation techniques in heterogeneous embedded systems (CPU, DSP etc)

- PRiME has developed a novel adaptive energy minimization approach for embedded heterogeneous systems. This is based on a runtime model - generated through regression-based learning of energy and performance trade-offs between the different computing resources in the system. Using this model, an application task is mapped to the most suitable computing resource during runtime, to minimize energy consumption for the application's performance requirement. Also used is DVFS control which adapts to the performance and workload variations.



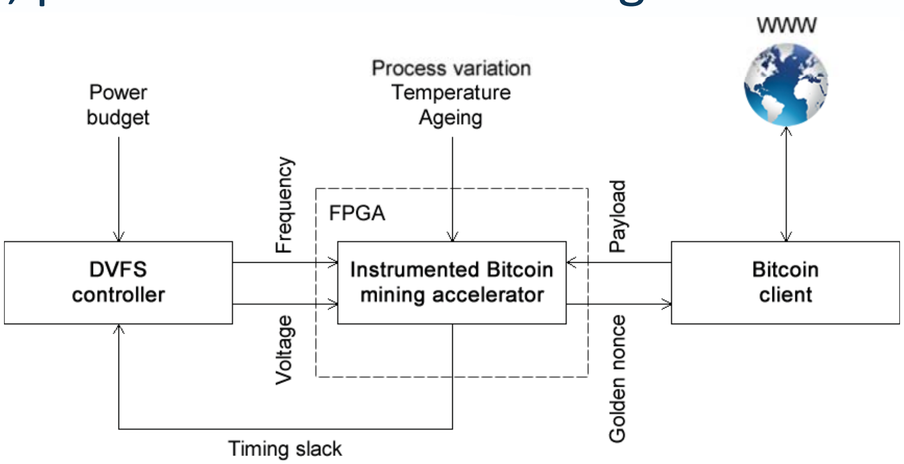
# 3. Energy-efficient thermal management of embedded systems

- This demonstrates thermal management techniques to improve the lifetime reliability of embedded systems. PRiME has investigated three main thermal parameters – peak temperature, average temperature and thermal cycling. Two control levers are used – CPU voltage-frequency and thread allocation – and the underlying control algorithm for managing energy and temperature is based on reinforcement learning principles, specifically Q-learning. Significant thermal improvements and energy reduction are demonstrated.



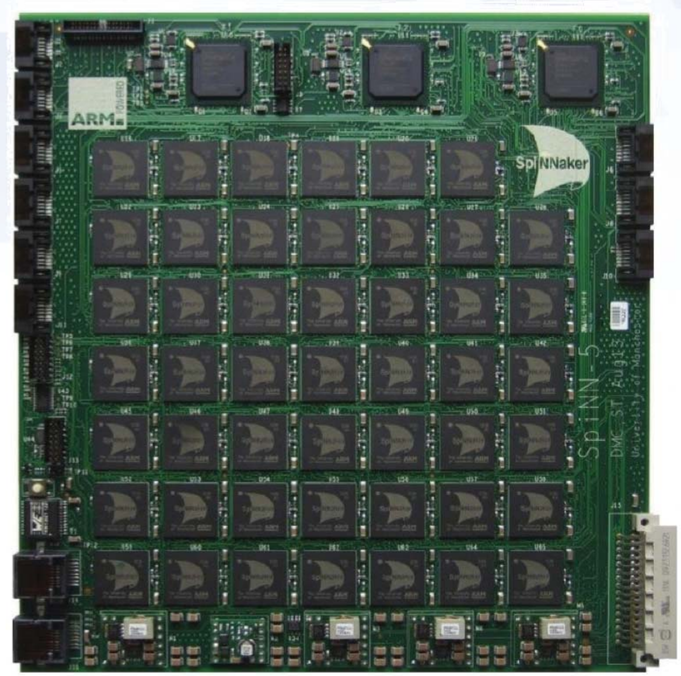
# 4. DVFS of a custom hardware accelerator using timing slack measurement

- This shows an example of dynamic voltage and frequency scaling (DVFS) of a custom hardware accelerator using timing slack measurement. Slack measurement cells and control logic are added to monitor the most critical timing paths in the design. The combination of timing slack measurement and DVFS allows for the safe erosion of the bulk of timing guardbands introduced by hardware vendors to protect against effects including supply voltage fluctuation, temperature change, process variation and degradation. By obtaining information direct from the application circuit, guardbands are able to be reduced from tens/hundreds of ns typically seen down to just a few hundred picoseconds.



# 5. SpiNNaker real-time power measurement and optimisation

- SpiNNaker is a novel, massively parallel computer architecture inspired by the working of the human brain. Each SpiNNaker chip contains 18 ARM968 processors, with 48 chips on each SpiNNaker board. A microcortical neural simulation (small building block of the brain) will be shown running on 15 out of 48 SpiNNaker chips, while measuring the power consumption in real-time of the various parts of the system (Spinnaker, SDRAM, Peripherals, FPGA, Total power). This will demonstrate the effects of gradually reducing the frequency of the unused chips on the Spinnaker board while the simulation is running.



# Thank you!

Any Questions?

**Dr Geoff Merrett**
Associate Professor

**PRiME Applications and Demonstrators Lead**
University of Southampton
Highfield Campus, Southampton, SO17 1BJ UK
Tel: +44 (0)23 8059 2775
Email: gvm@ecs.soton.ac.uk
Web: www.geoffmerrett.co.uk