ARM mbed Support for Transient Computing in Energy Harvesting IoT Systems

Bogdan Lazarescu, Domenico Balsamo, Geoff Merrett

UNIVERSITY OF Southampton

EPSRC Engineering and Physical Sciences Research Council

Introduction

Energy harvesters offer the possibility for embedded IoT computing systems to operate without batteries. However, their output power is usually **unpredictable and highly variable**.

To mitigate the effect of this variability, systems incorporate large energy buffers, increasing their size, mass and cost.

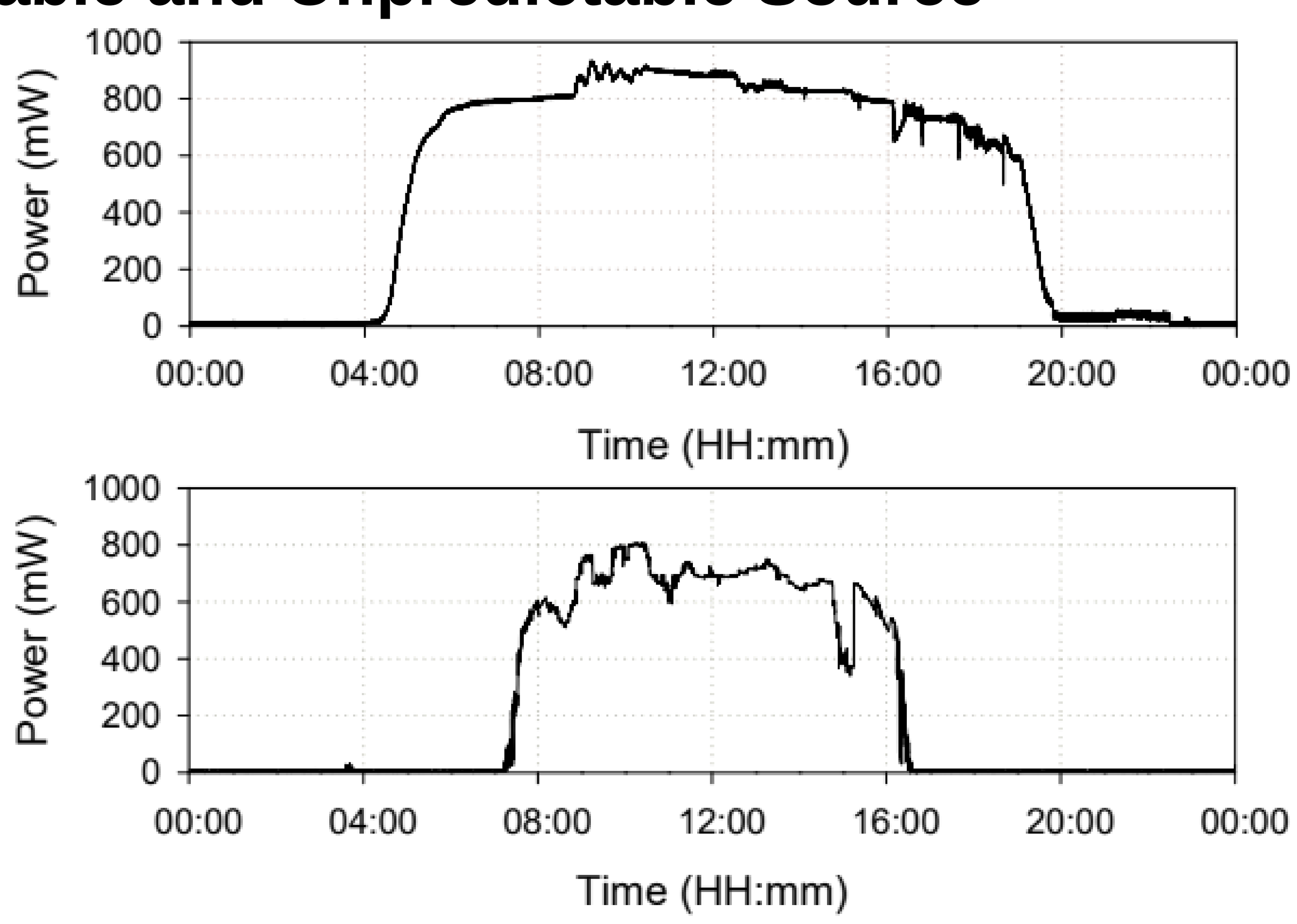
The emerging class of **transient computing** systems differs from this approach, operating directly from the energy harvesting source and minimizing or removing additional energy storage.

Existing transient approaches are **largely designed for specific applications and architectures**. Hence, they suffer from not being broadly applicable across multiple embedded IoT platforms.

To address this challenge, transient approaches need to be integrated within a general IoT programming framework such as **ARM's mbed IoT Device Platform**.

This support is offered through **libraries and application programming interfaces(APIs)** which enable transient computing to be implemented as a service on top of IoT application protocols.

Variable and Unpredictable Source



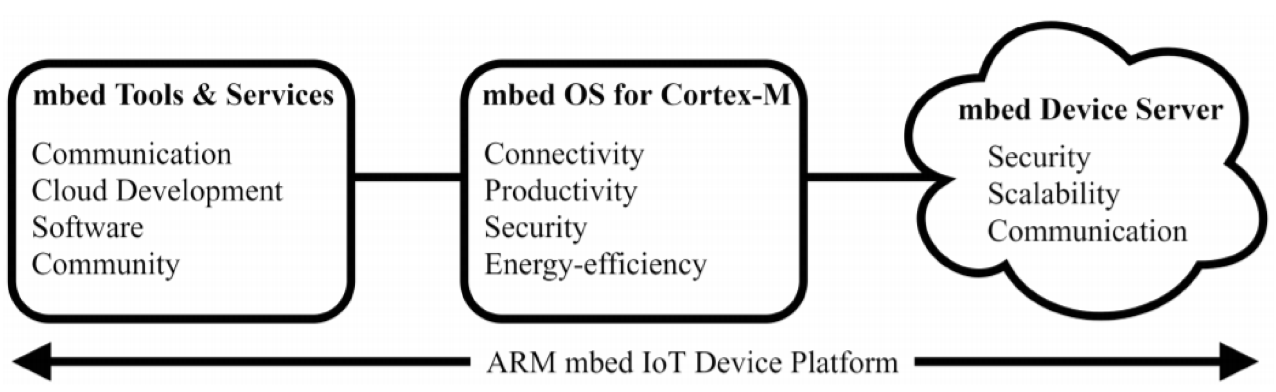
Power outputs from two PV cells located outdoor

To accommodate this variability systems incorporate large energy buffers such as rechargeable batteries or super capacitors.

Transient computing enables computation to be sustained despite power outages, by **retaining the system state** before a power failure. However, existing approaches are **designed for specific applications** or **platforms**.

ARM mbed ecosystem

To address this challenge, transient computing can be integrated as part of IoT programming frameworks such as **ARM's mbed IoT Device Platform.**

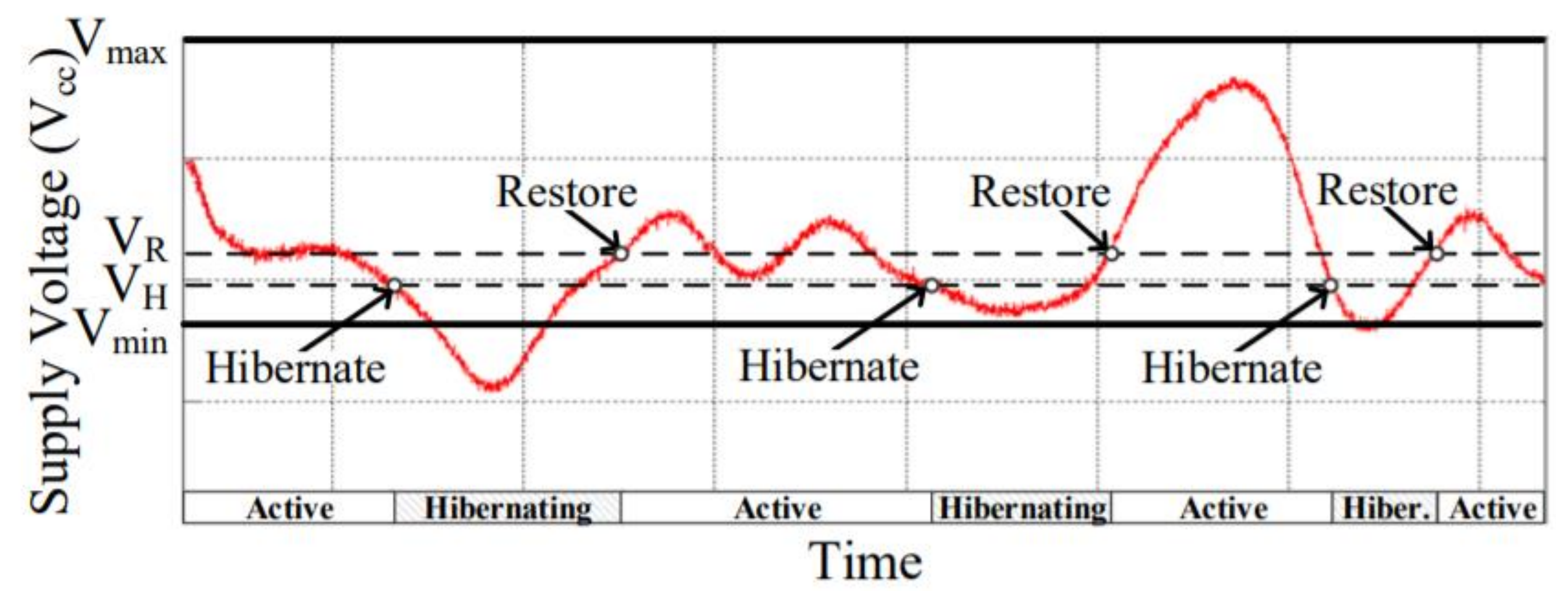


mbed Requirements for Transient

mbed requirements	Transient Approaches: Mementos	QuickRecall	Hibernus	Hibernus++
No compiler-level modification required		✓	✓	✓
No specific architectures and extra HW	✓		✓	
Applicable to Flash memory	✓		✓	✓

Hibernus

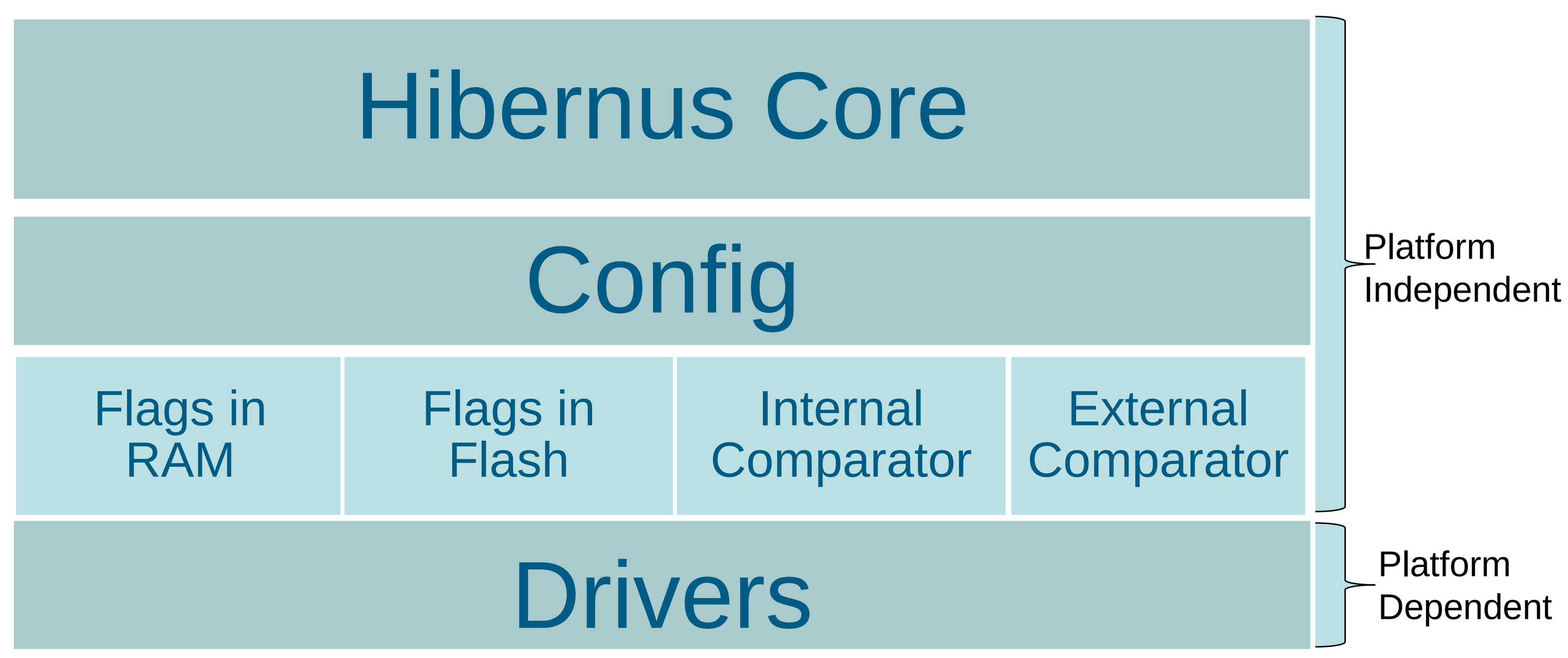
Here, it is illustrated the behaviour of Hibernus in response to a variable supply voltage. When the voltage falls below V_H, the system saves a snapshot and hibernates. When the voltage rises above V_R, the system restores a snapshot and continues operation.



Addressed Challenges

- Memory Layout
- Memory configurations and sizes
- Flash IAP for different boards and manufacturers
- Presence of an internal analog comparator
- Use of an external analog comparator
- Saving the flags in RAM memory when the Flash size is limited
- Peripheral registers restoration
- Avoid fixed memory addresses conflicts
- Create a general library that can be easily used and modified in order to work with all boards that offer support for mbed or mbed OS

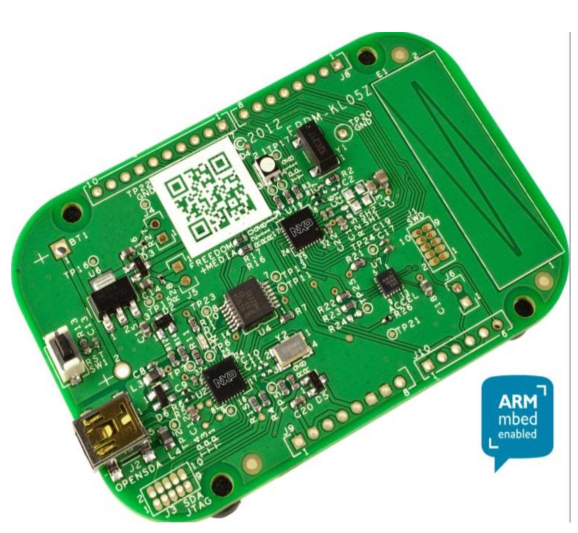
Hibernus Software structure with mbed



Supported Platforms

Supported Platforms

- **Freescale KL05Z**
  - Cortex M0+
  - Internal Comparator
  - Flags stored in a separate Flash area
  - Small RAM capabilities(4MB)



- **LPC11U24**
  - Cortex M0
  - External Comparator
  - Flags stored in Ram memory
  - Higher RAM capabilities(32MB)

www.transient.ecs.soton.ac.uk

ARM Research Summit 11-13 September 2017

Contact: Dr. Domenico Balsamo

Email: D.Balsamo@soton.ac.uk

ARM Research Summit 11-13 September 2017