UNIVERSITY OF Southampton

Energy-Driven Computing:

Rethinking the Design of Energy Harvesting Systems

10 February 2020

UNIVERSITY OF SOUTHAMPTON

University of Southampton

- ~25,000 students
- Top 100 universities worldwide (QS‘19)
- Founding member of UK’s Russell Group

School of Electronics and Computer Science

- ~2,000 students
- ~250 PhD research students
- ~100 academics/faculty
- Top 3 in UK for EEE
- 14 research groups/centres

POWERING THE IOT

- We've got batteries!
  - So what's the problem?

- More things = batteries/wires/people
  - Pervasive/IoT/ubiquitous
- Fit-and-forget/maintenance issues
  - Smart homes/grid/metering
- Weight vs volume vs lifetime
  - e.g. Wearables

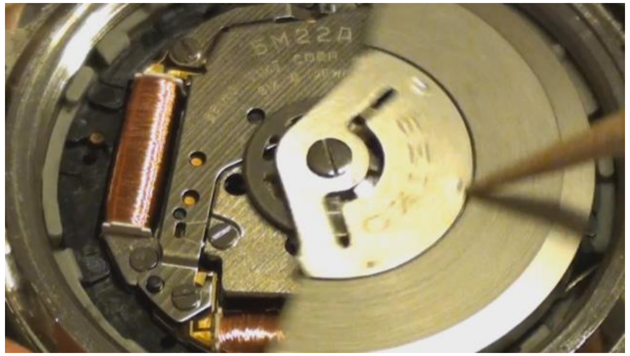


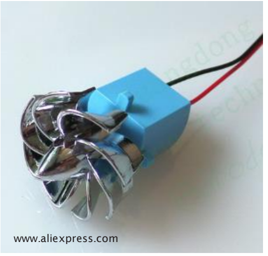
image: RS fabrications, http://www.rsfabrications.com/wp-content/uploads/2009/09/cimg2821.jpg
image: http://commons.wikimedia.org/wiki/File:Canary_Wharf_at_night,_from_Shadwell_cropped.jpg
image: Australian Transport Safety Bureau, http://www.atsb.gov.au/media/2489702/ro2010001.jpg

EH SOURCES

varies temporally

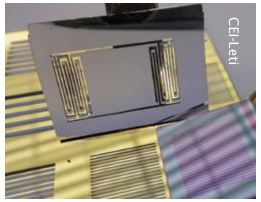
varies spatially

Flow



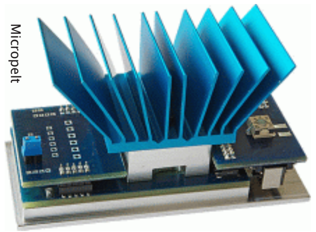
D. Balsamo et al. Hibernus++: a self-calibrating and adaptive system for transiently-powered embedded devices. IEEE TCAD, 1-13.

Vibration



Beeby, S.P. et al. A comparison of power output from linear and non-linear kinetic energy harvesters using real vibration data. Smart Materials and Structures, 22, (7), 075022.

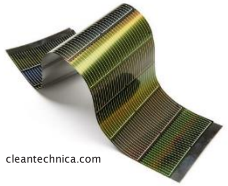
Heat

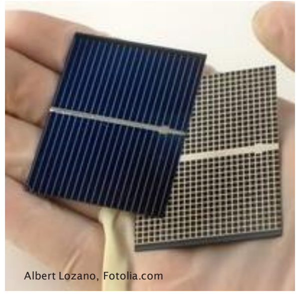


V. Leonov, "Thermoelectric Energy Harvesting of Human Body Heat for Wearable Sensors," IEEE Sensors Journal, vol.13, no.6, pp.2284-91, June13

Power / Energy

Light





http://solar.rainham-kent.co.uk

Force

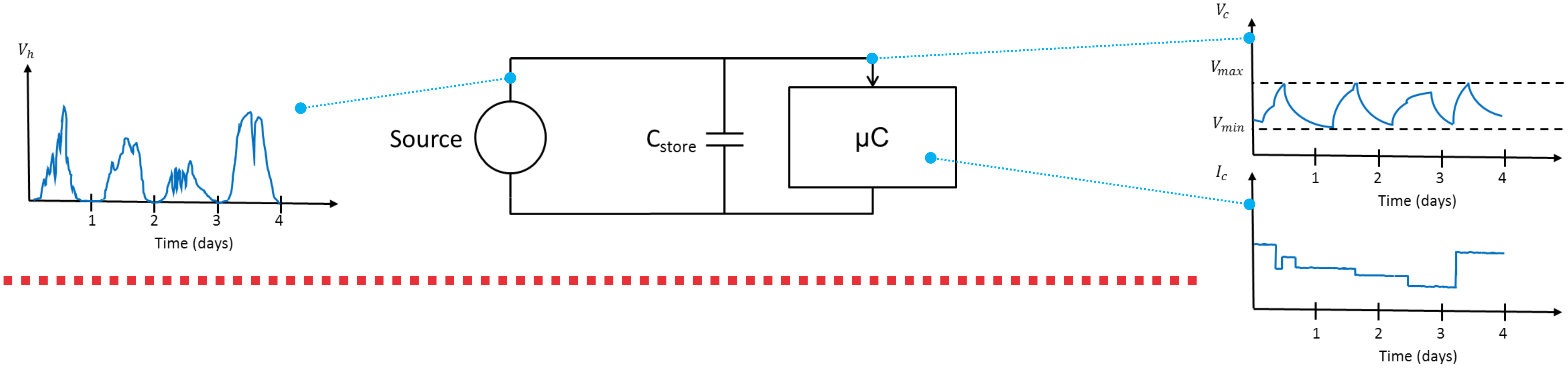
Zhao, J. et al. "A Shoe-Embedded Piezoelectric Energy Harvester for Wearable Sensors," Sensors 2014, 14, 12497-12510.

Highly variable supply + variable consumption!

UNIVERSITY OF Southampton

ENERGY-NEUTRAL COMPUTING

Schematic



Block Diagram

Power Supply → Power Conversion → Energy Storage → Power Conversion → Computation (Load)

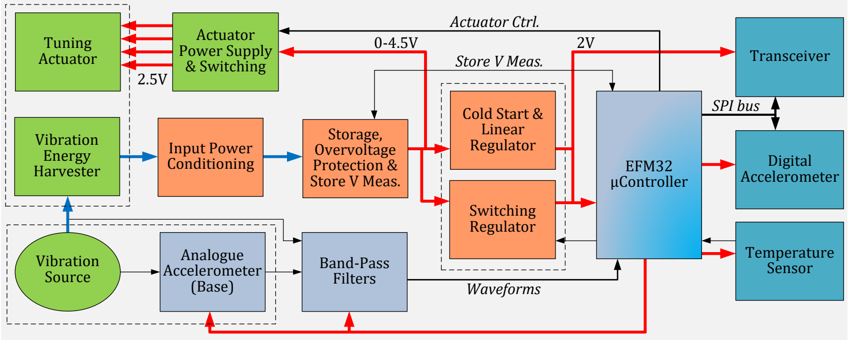
$$\int_{(n-1)\cdot T}^{n\cdot T} P_h(t)dt = \int_{(n-1)\cdot T}^{n\cdot T} P_c(t)dt$$

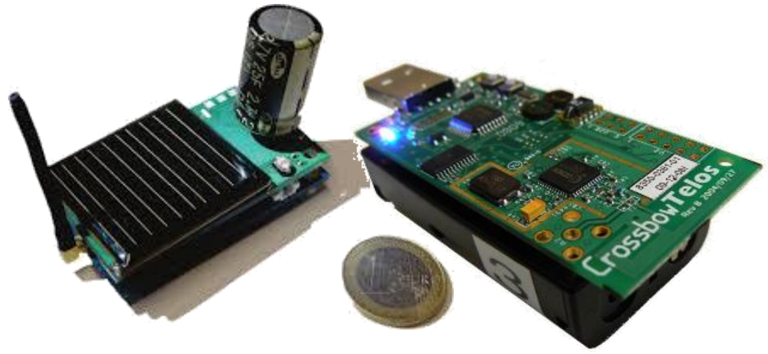
Stored Energy

University of Southampton

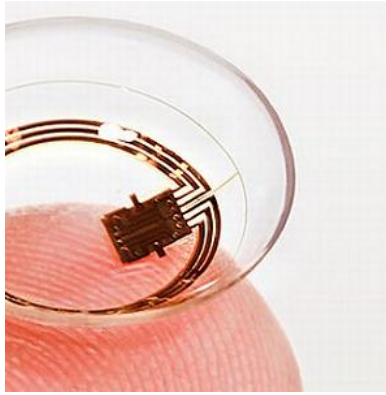
ENERGY-DRIVEN COMPUTING

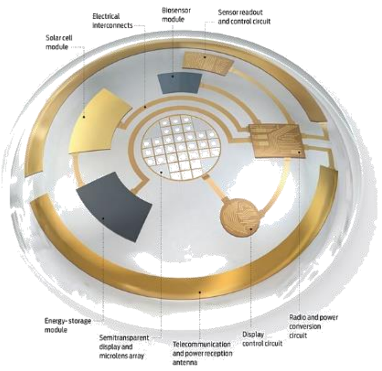
- What's wrong with energy storage and complexity?





- Emerging applications demanding small dimensions, volumes, weight, cost, etc





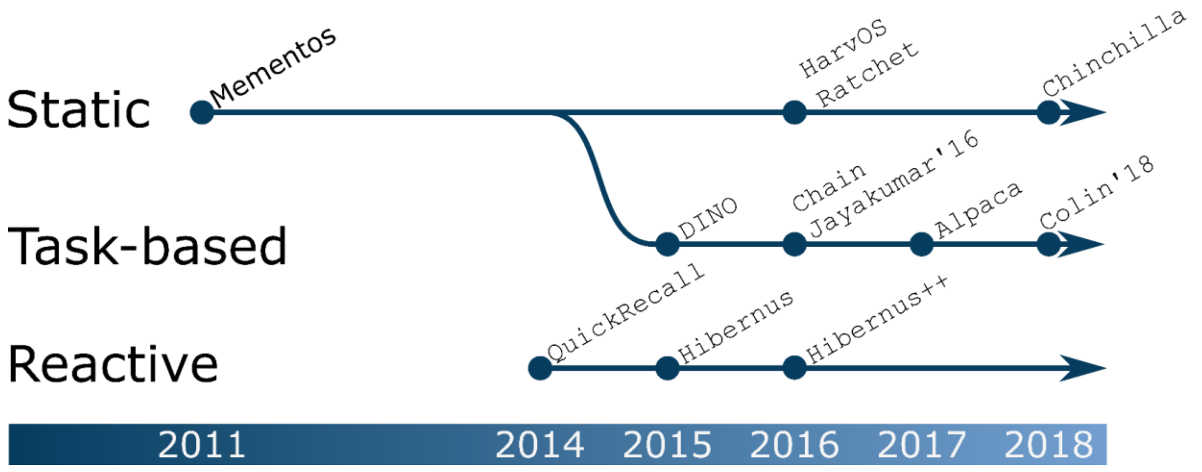
INTERMITTENT COMPUTING

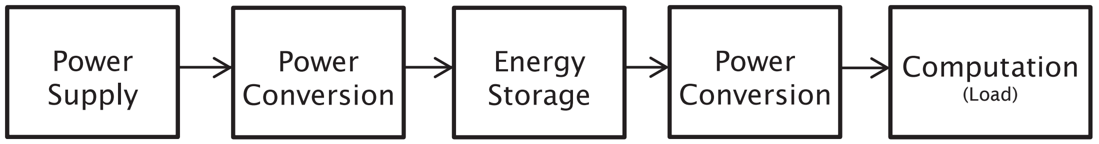
Compute operates across power outages

Static Approaches
- Application instrumented with checkpoints

Task-based Approaches
- Application divided into set of small tasks
- State saving during transitions

Reactive Approaches
- Save state on power failure
- No roll-back -> consistent memory
- Less overhead (runtime & mem)



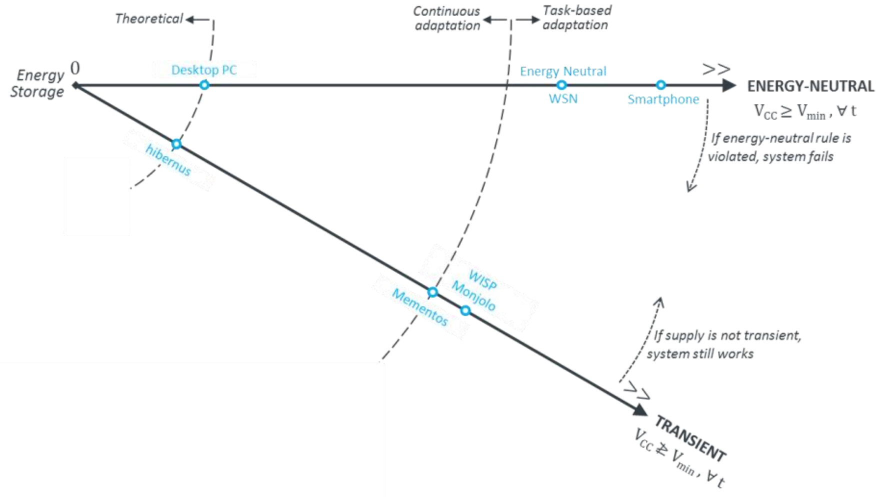


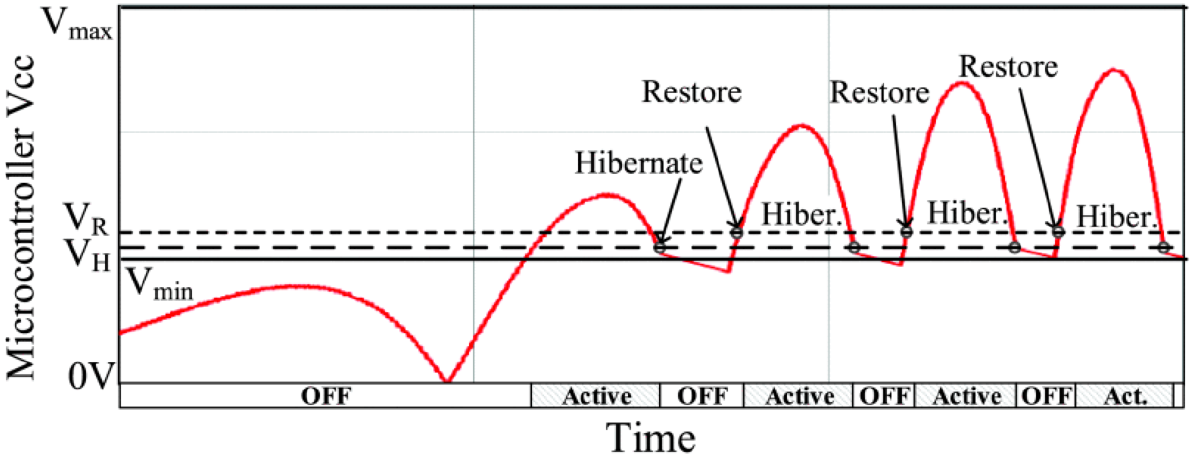
REACTIVE IC: *hibernus*

- Make only a single (but always a single) snapshot per supply ‘failure’
  - Removes wasted snapshots
  - Ensures always makes valid snapshot
- Make it as late as possible
  - Avoids re-executing code
  - Maximises execution time

$$\underbrace{T_{\text{Hibernus}}}_{\text{Total execution}} = \overbrace{T_a}^{\text{Algorithm}} + \underbrace{n_\iota}_{\text{No. interruptions}} (\overbrace{T_s}^{\text{Save snapshot}} + \underbrace{T_r}_{\text{Restore snapshot}} + \overbrace{\overline{T_\lambda}}^{\text{Sleep}})$$

$$\underbrace{T_{\text{mementos}}}_{\text{Total execution}} = \overbrace{T_a}^{\text{Algorithm}} + \underbrace{n_\iota}_{\text{No. interruptions}} (\overbrace{T_r}^{\text{Restore snapshot}} + \underbrace{\frac{T_a}{2n_m}}_{\text{Backtrack}}) + \overbrace{n_m(T_m + \rho_s T_s)}^{\text{Monitoring and save snapshot}}$$



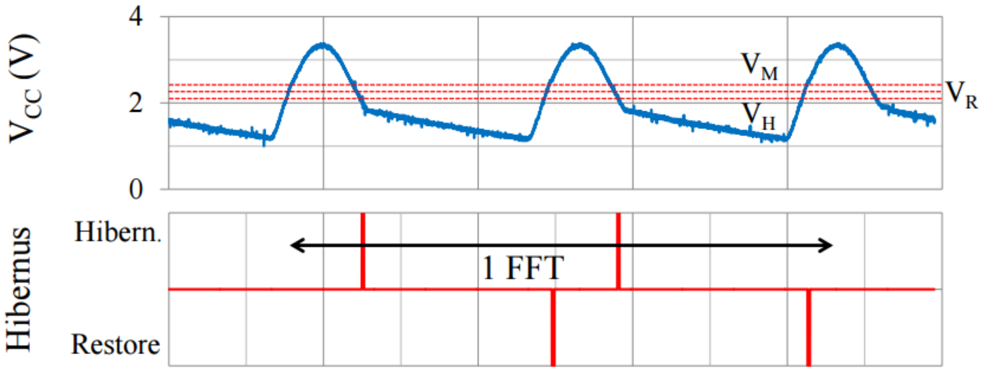


D. Balsamo, A.S. Weddell, G.V. Merrett, B.M. Al-Hashimi, D. Brunelli, and L. Benini, “Hibernus: Sustaining Computation during Intermittent Supply for Energy-Harvesting Systems,” IEEE Embedded Systems Letters, 2014

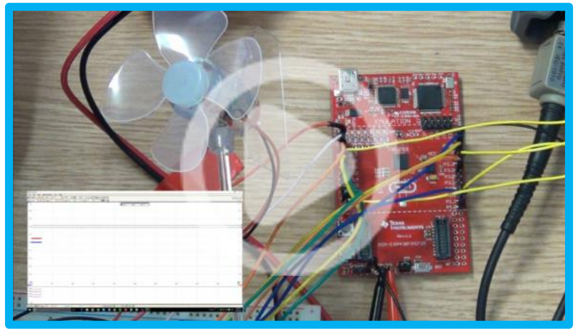
REACTIVE IC: *hibernus*

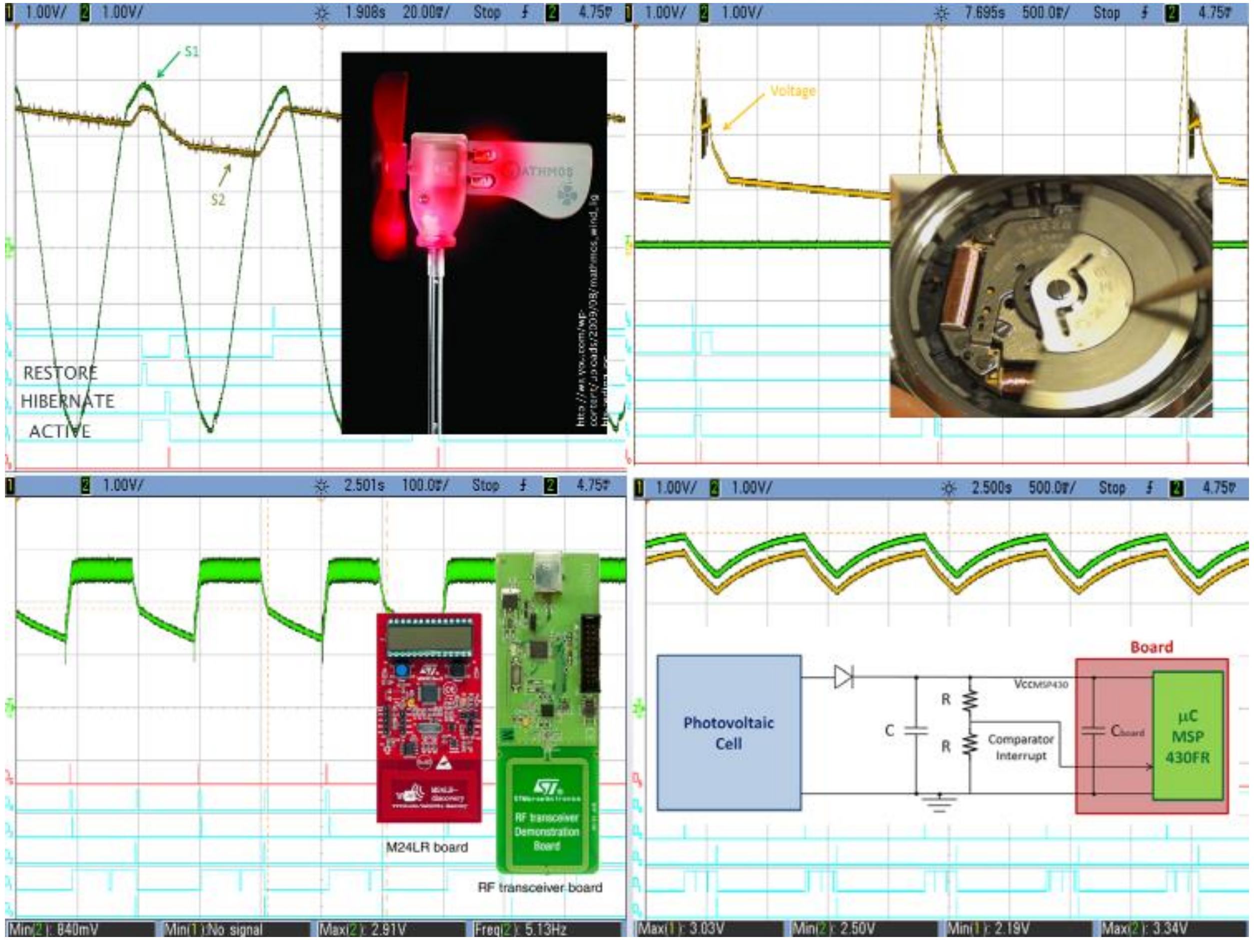
Time overheads reduced by 75-100%
Energy overheads reduced by 50-80%

- Controlled source (signal generator)



- Real energy harvesting sources





D. Balsamo, A.S. Weddell, A. Das, A. Rodriguez Arreola, D. Brunelli, B.M. Al-Hashimi, G.V. Merrett, L. Benini, (2016) Hibernus++: a self-calibrating and adaptive system for transiently-powered embedded devices. *IEEE TCAD*, 1-13.

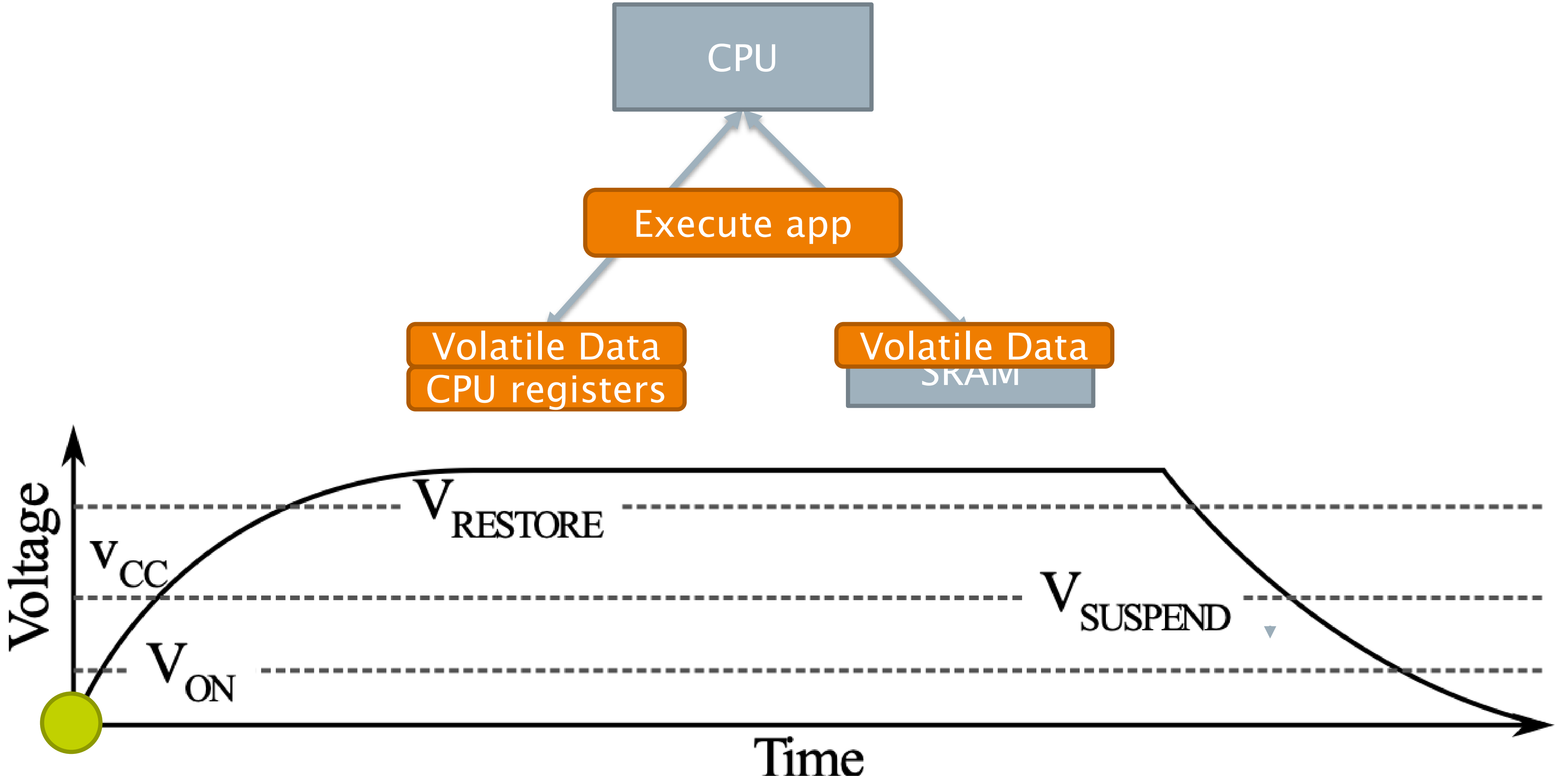
REACTIVE IC: *hibernus*

Developed a range of optimisations/extensions

- Self calibration for runtime hibernate threshold optimisation (*hibernus*++)
- Adaptive restore based on EH properties (*hibernus*++)
- Hibernation and restore of external peripheral state (*RESTOP*)
- Support for Arm Mbed
- Support for communication and mesh networking
- Applications (cycle computer, fitness monitor, etc)
- Fine-grained power adaption (*PowerNeutrality*)
- Efficient state retention (Selective Policies, *ManagedState*)
- Tooling for system design (*ENSPECT*, *FUSED*, Device Sizing)

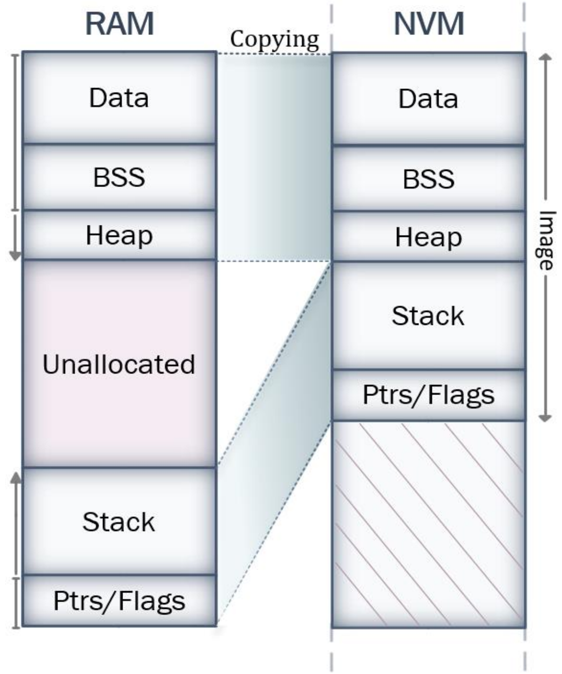
www.transient.ecs.soton.ac.uk

EFFICIENT STATE RETENTION

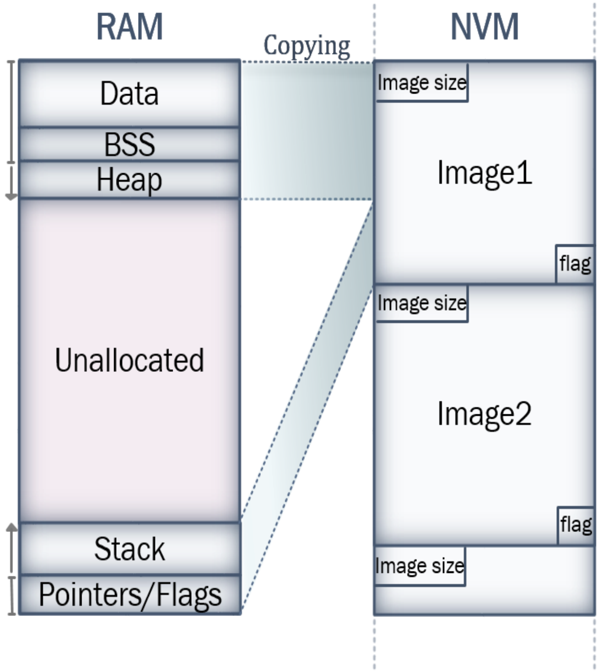


STATE RETENTION POLICIES

Allocated State (symmetric/asymmetric NVMs)

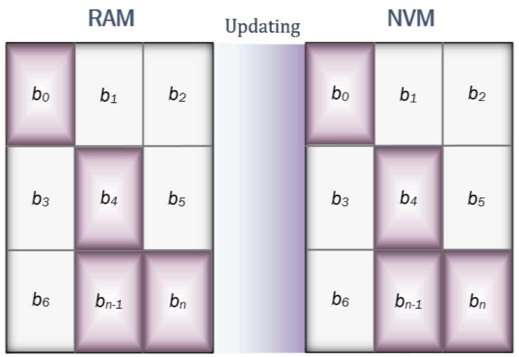


NVMs with no erase cost

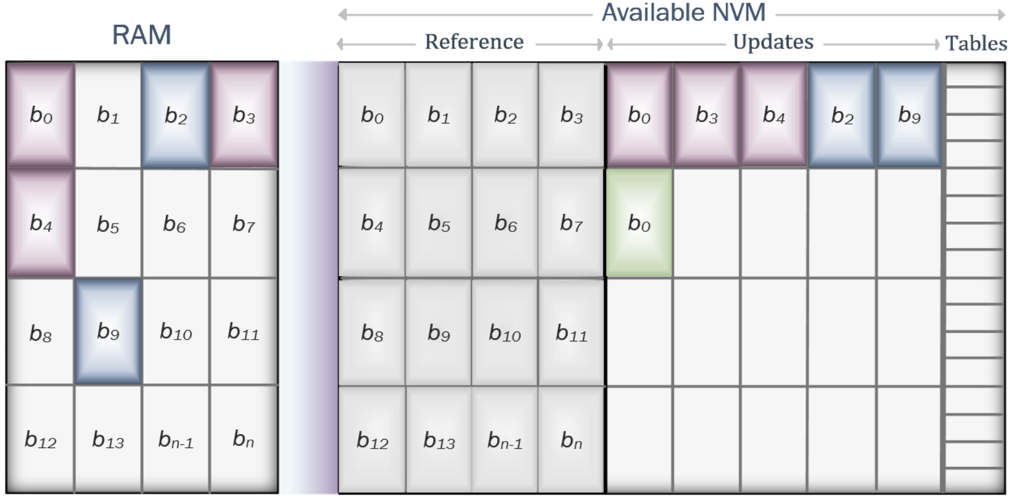


NVMs with erase cost

Multiple Blocks (asymmetric NVMs)



NVMs with no erase cost

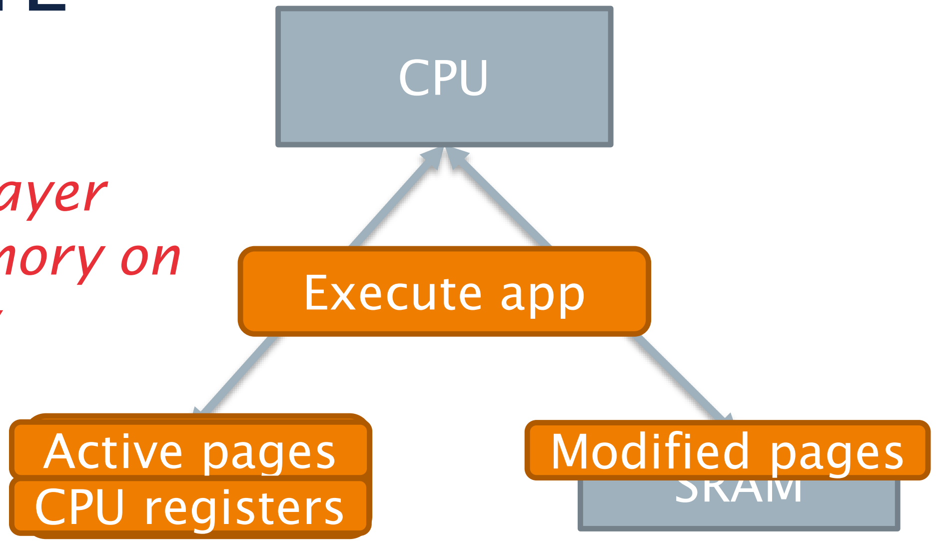


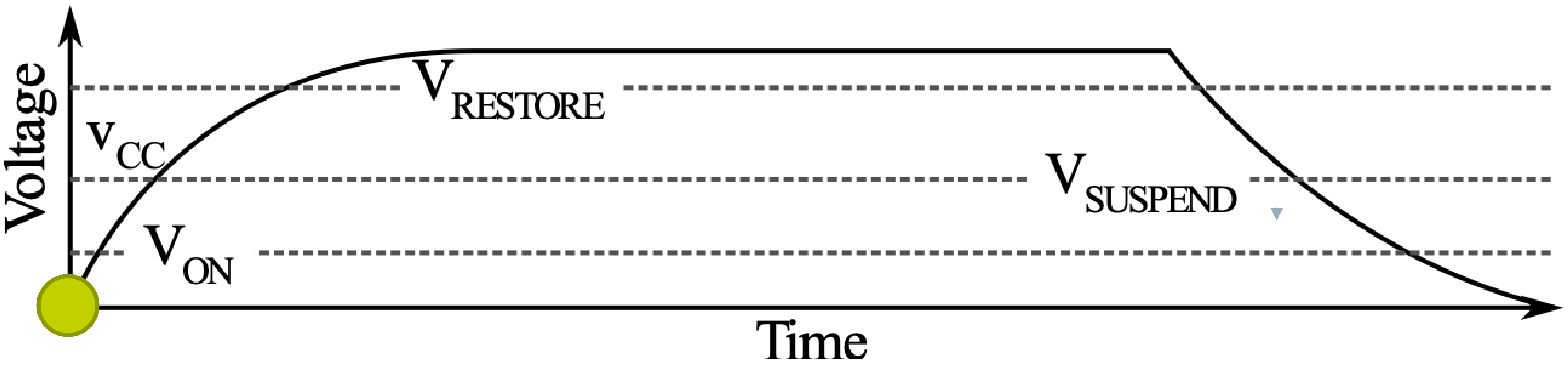
NVMs with no erase cost

Verykios, Theodoros D., Balsamo, Domenico and Merrett, Geoff V. (2018) **Selective policies for efficient state retention in transiently-powered embedded systems: exploiting properties of NVM technologies.** *Sustainable Computing: Informatics and Systems.*

“MANAGED STATE”

A lightweight memory management software-layer that loads pages of memory on demand, and saves only modified pages.

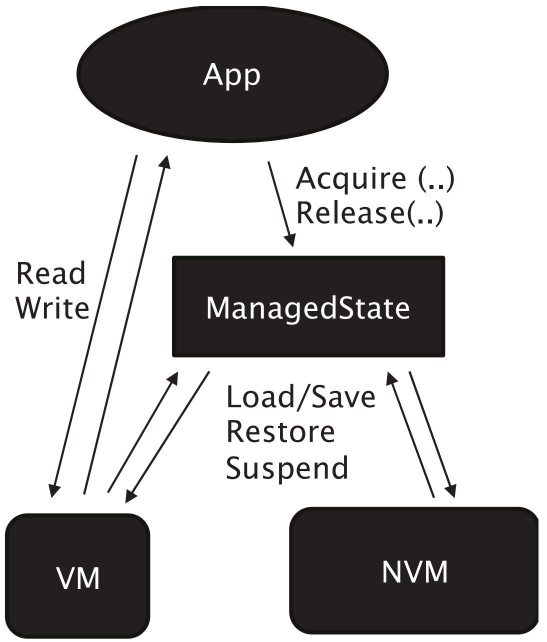


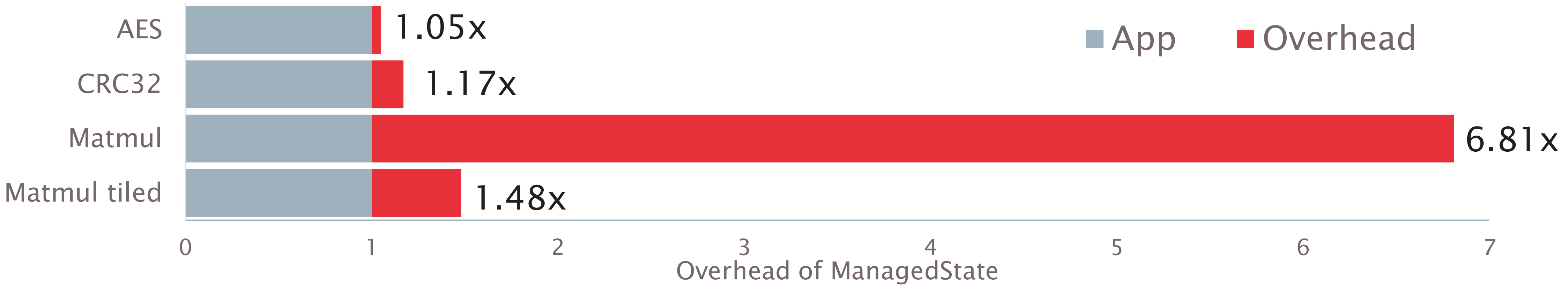


Sliper, Sivert T., Balsamo, Domenico, Nikoleris, Nikos, Wang, William, Weddell, Alexander and Merrett, Geoff (2019) **Efficient state retention through paged memory management for reactive transient computing**. *Design Automation Conference*, Las Vegas, United States. 02 - 06 Jun 2019. 6 pp .

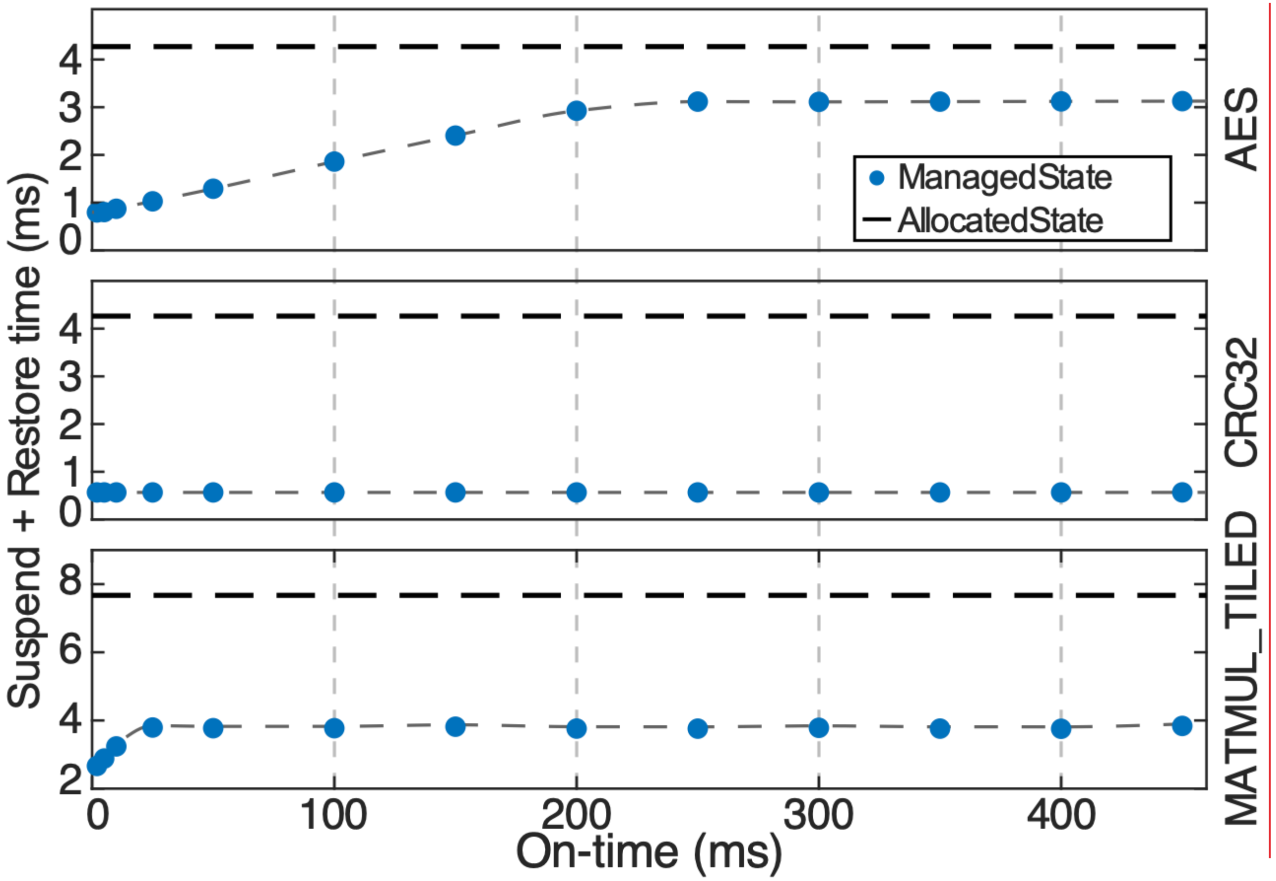
MANAGED STATE: USAGE AND OVERHEADS

```
#include <managed_state.h>
#define MMDATA __attribute__((section(".mmdata"))

uint8_t data [2048] MMDATA;
uint8_t key [16] = …;

...

for (uint8_t *ptr = data; ptr < end; ptr += 16) {
        ms_acquire(ptr, 16, MM_READWRITE);
        aes_encrypt(ptr, key);
        ms_release(ptr, 16);
}
```

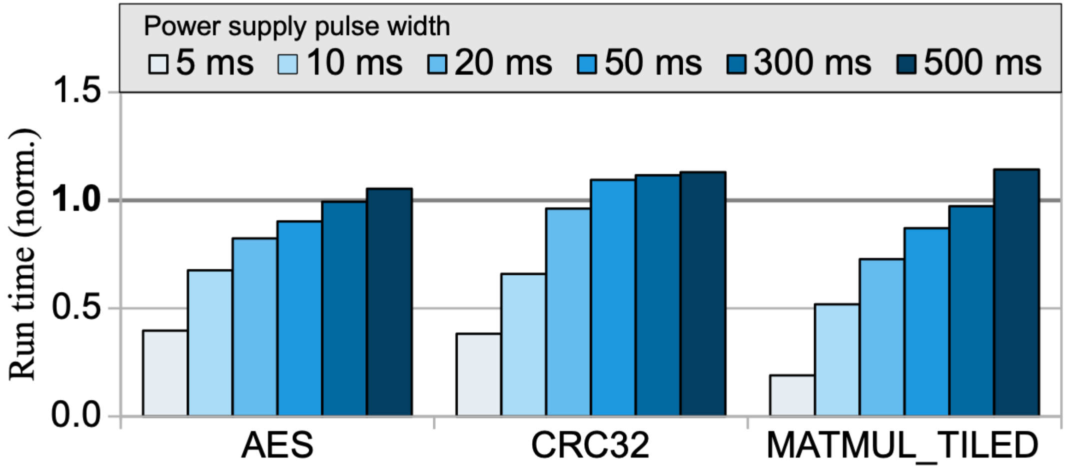




MANAGED STATE: RESULTS



Suspend + Restore time in relation to on-time



Completion time relative to AllocatedState.
Powered by voltage pulses of variable width

Sliper, Sivert T., Balsamo, Domenico, Nikoleris, Nikos, Wang, William, Weddell, Alexander and Merrett, Geoff (2019) **Efficient state retention through paged memory management for reactive transient computing**. *Design Automation Conference*, Las Vegas, United States. 02 - 06 Jun 2019. 6 pp .

University of Southampton

CLOSED-LOOP PERFORMANCE & ENERGY SIMULATION

Developing energy-driven computing systems is difficult

- Operation driven directly by availability of energy
- Introspection/debug inevitably affects stored/harvested energy (and therefore operation)
- Typical embedded systems development can't cope with a DUT that frequently powers off
- Repeatability of EH is problematic

FUSED (Full-system Simulation of Energy-Driven Computers)

- Open source full-system simulator for energy-driven computers
- SystemC for digital and mixed-signal simulation, modelling microcontroller and mixed-signal circuitry
- Models power supply/consumption and execution in a closed loop, modelling the interaction between.
- Accurate power model obtained correlates microarchitectural events with real power measurements
- Enables hardware-software codesign and design space exploration.

Download: www.arm.ecs.soton.ac.uk/technologies/fused

Sivert Sliper, William Wang, Nikos Nikoleris, Alex S. Weddell, Geoff V. Merrett, "**Fused: Closed-loop Performance and Energy Simulation of Embedded Systems**," *ISPASS 2020* (in press)

FUSED

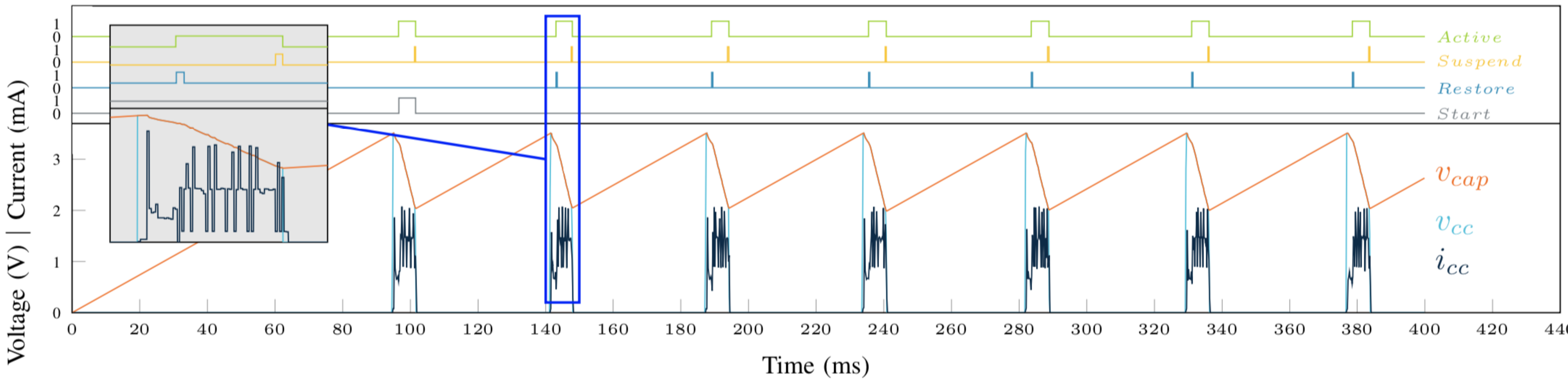


Fig. 9. Simulation trace of a reactive intermittent computing system powered by a 200 μA current-limited power supply. The top traces show the logic levels of GPIO pins indicating the operation of the device, and the lower traces show the microcontroller supply voltage (v_{cc}), the storage capacitor voltage (v_{cap}) and the current draw (i_{cc}).

Download: www.arm.ecs.soton.ac.uk/technologies/fused

Sivert Sliper, William Wang, Nikos Nikoleris, Alex S. Weddell, Geoff V. Merrett, "**Fused: Closed-loop Performance and Energy Simulation of Embedded Systems**," *ISPASS 2020* (in press)

FUSED

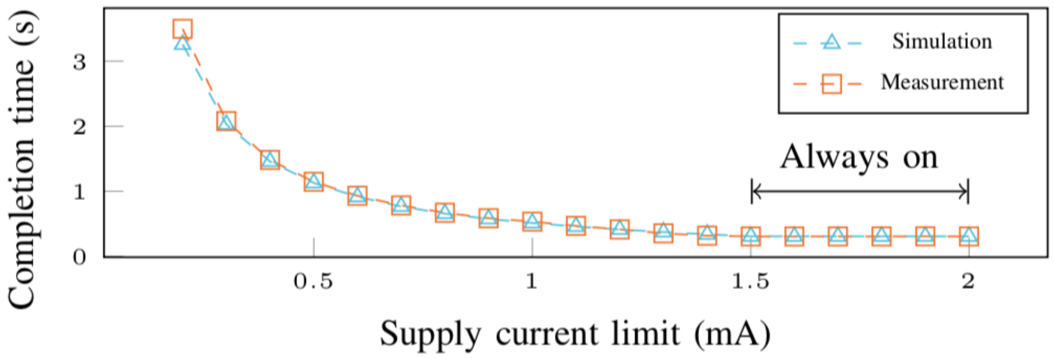


Fig. 10. Completion time of AES encryption when running intermittently, powered by a current-limited power source.

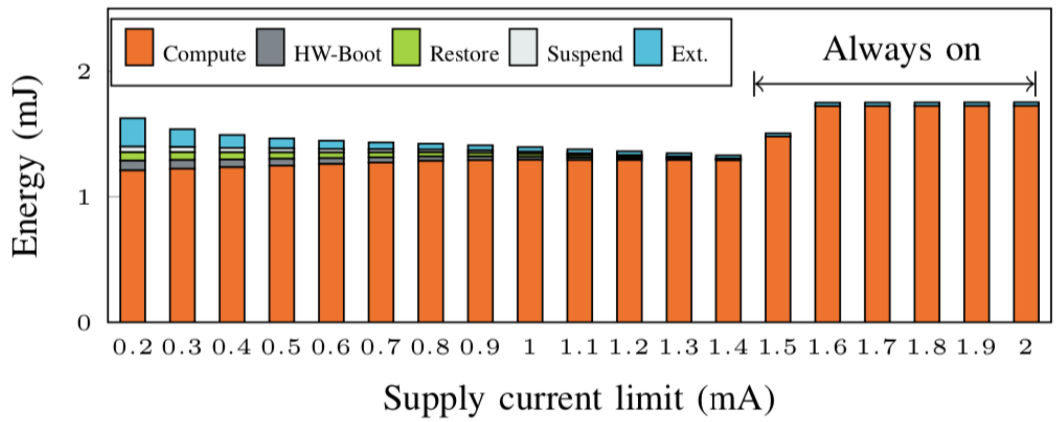


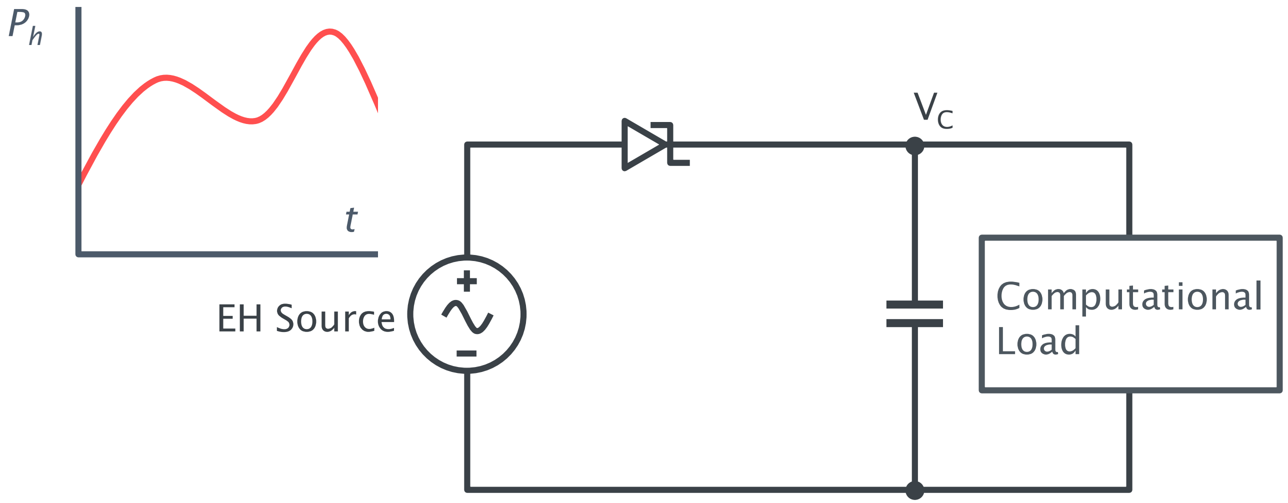
Fig. 11. Full-system energy consumption when running AES encryption intermittently, powered by current-limited power source. The energy consumption is divided into stacked bars for the external circuitry (*ext.*), hardware boot (*HW-Boot*), and the operational phases *restore*, *compute* and *suspend*.

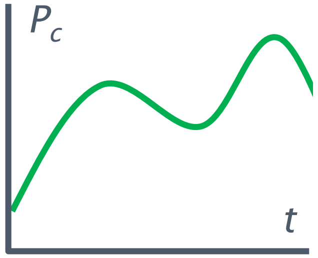
Download: www.arm.ecs.soton.ac.uk/technologies/fused

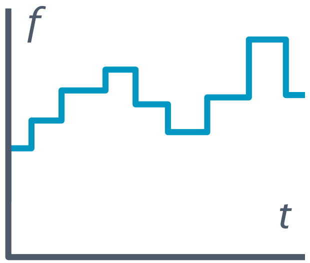
Sivert Sliper, William Wang, Nikos Nikoleris, Alex S. Weddell, Geoff V. Merrett, "**Fused: Closed-loop Performance and Energy Simulation of Embedded Systems**," *ISPASS 2020* (in press)

POWER NEUTRAL COMPUTING

- In Energy-Neutral computing, $\int_{(n-1)\cdot T}^{n\cdot T} P_h(t)dt = \int_{(n-1)\cdot T}^{n\cdot T} P_c(t)dt$ over a 'large' T
- In Power-Neutral computing, $P_c(t) \cong P_h(t)$



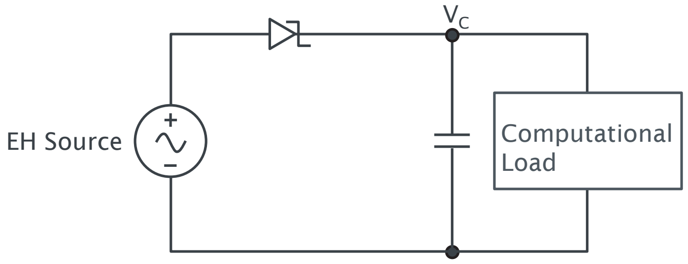


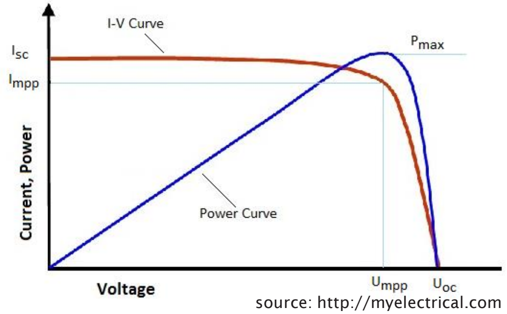


- Power consumption is modulated, eg through:
  - Core frequency and/or voltage
  - Power gating processor elements
- We can approximate power-neutral behaviour if $V_C(t) \approx k \ , \forall t$

POWER NEUTRAL COMPUTING

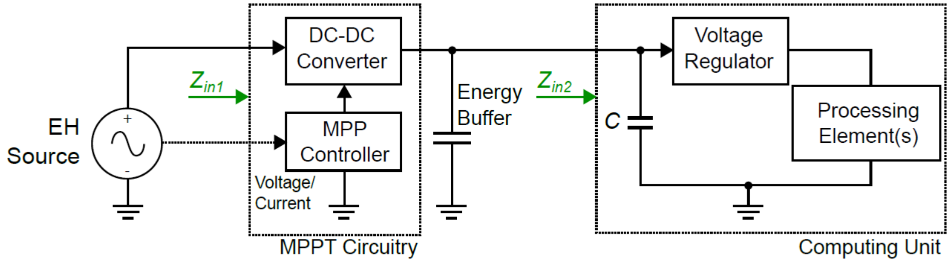
- What happens if V_C remains constant ($V_C(t) \approx k \ , \ \forall t$)?





source: http://myelectrical.com

- MPPT approaches are needed as $V_C(t) \neq V_{H\_MPP}(t) \ , \ \forall t$
- Traditional EH systems decouple source and load using MPPT

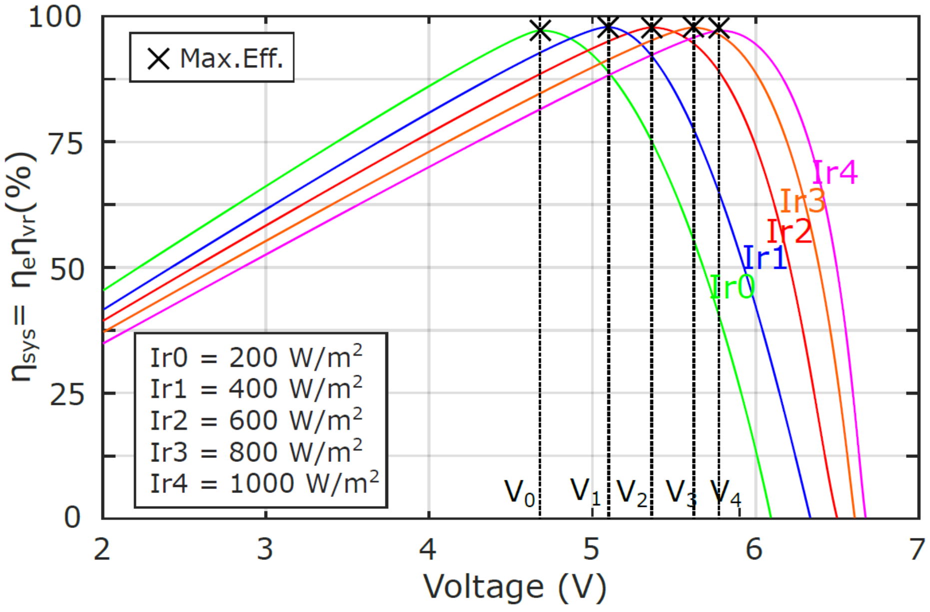


- 'Software-only' MPPT can modulate k, i.e. $V_C(t) = V_{H\_MPP}(t)$

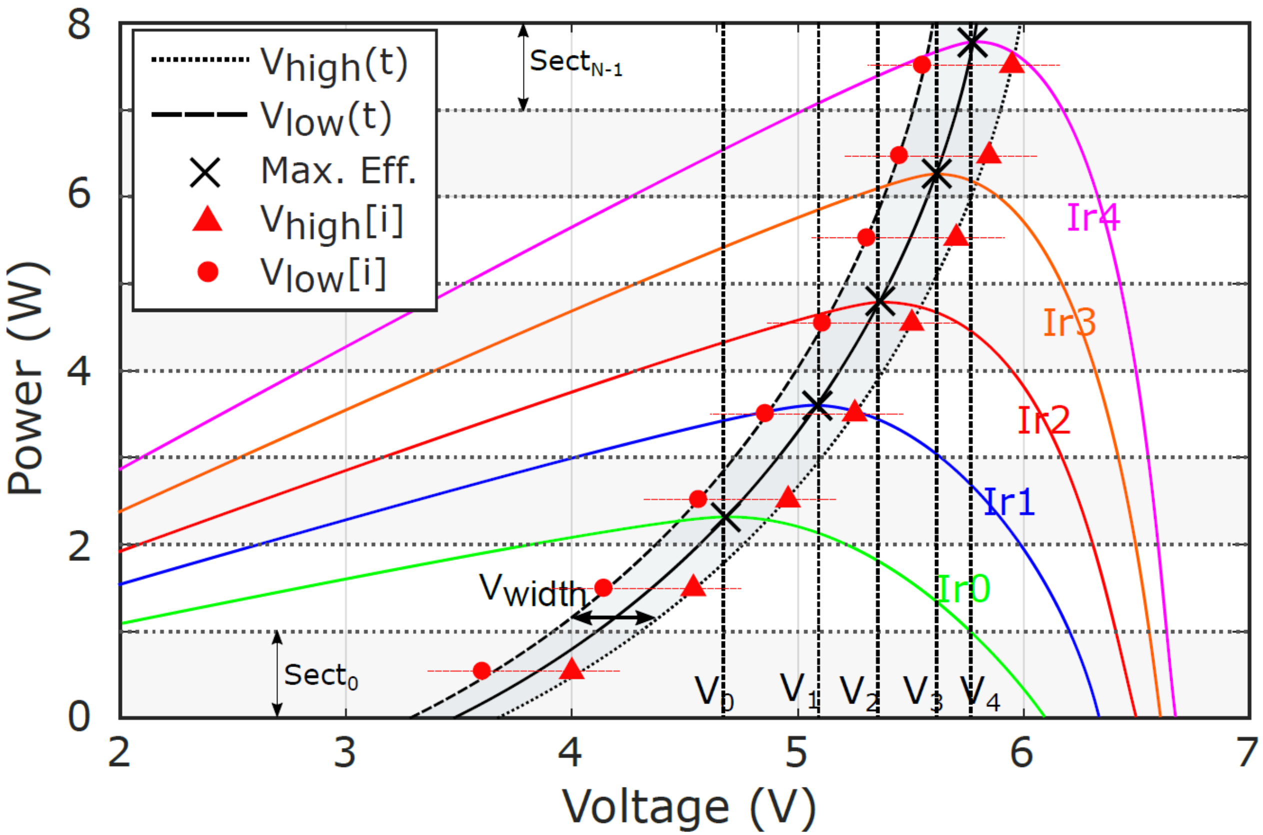
MAXIMUM SYSTEM EFFICIENCY

MPPT maximises η_{eh}; we maximise η_{sys}

$$\eta_{sys} = \frac{P_{out}}{P_{max}} = \left(\frac{P_{out}}{P_{in}}\right) \cdot \left(\frac{P_{in}}{P_{max}}\right) = \eta_{eh}\eta_{vr}$$



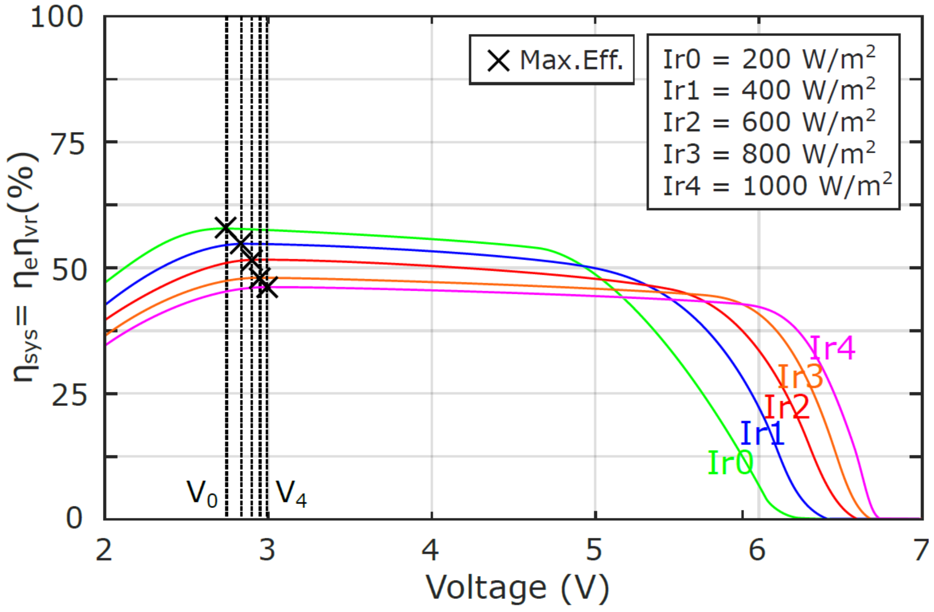
Estimating V_{cc} and P_c, we can identify MPP



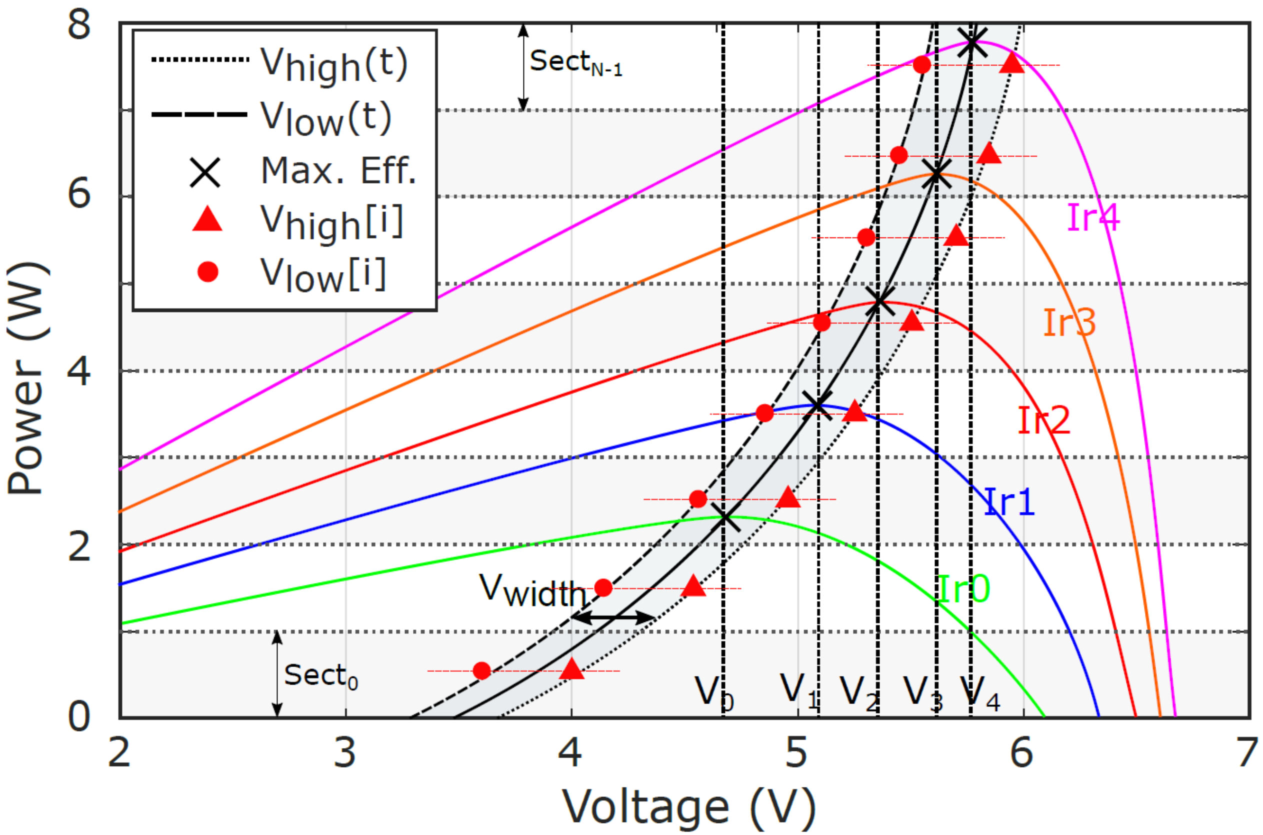
MAXIMUM SYSTEM EFFICIENCY

MPPT maximises η_{eh}; we maximise η_{sys}

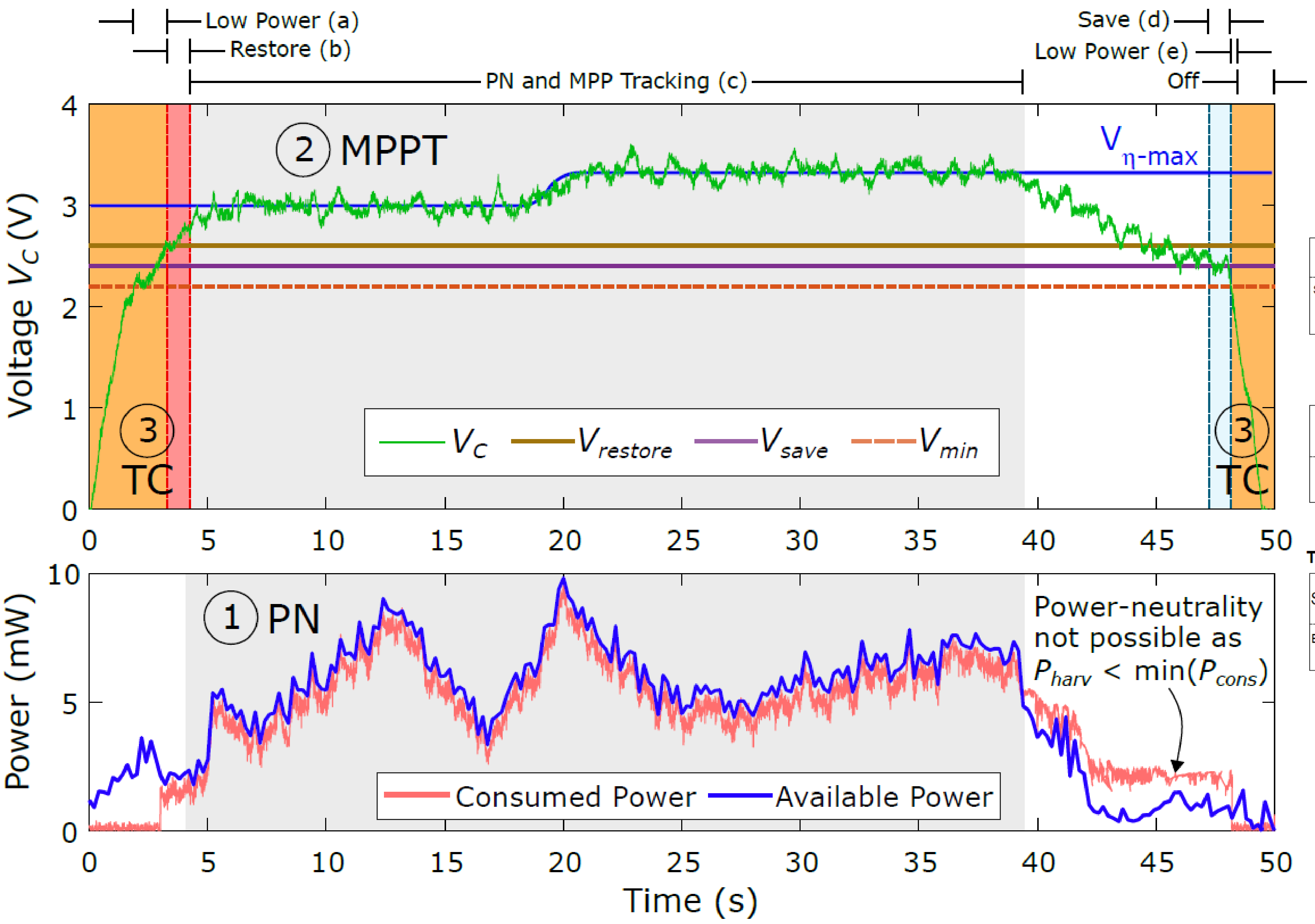
$$\eta_{sys} = \frac{P_{out}}{P_{max}} = \left(\frac{P_{out}}{P_{in}}\right) \cdot \left(\frac{P_{in}}{P_{max}}\right) = \eta_{eh}\eta_{vr}$$

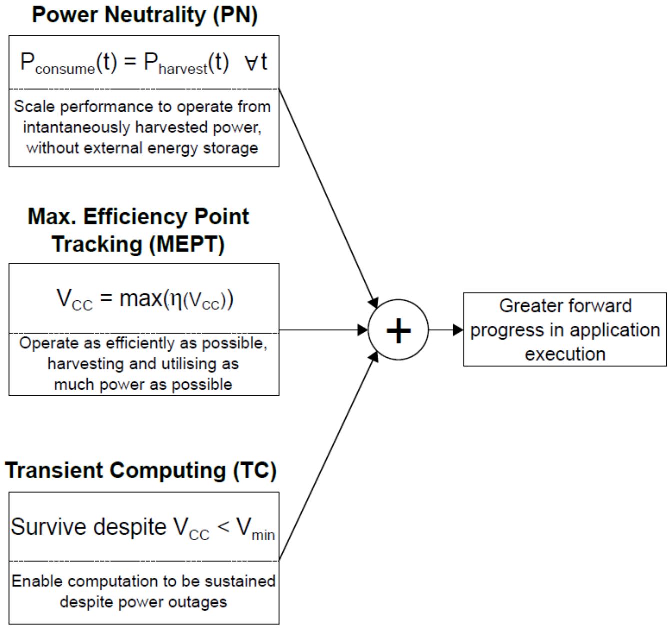


Estimating V_{cc} and P_c, we can identify MPP



University of Southampton

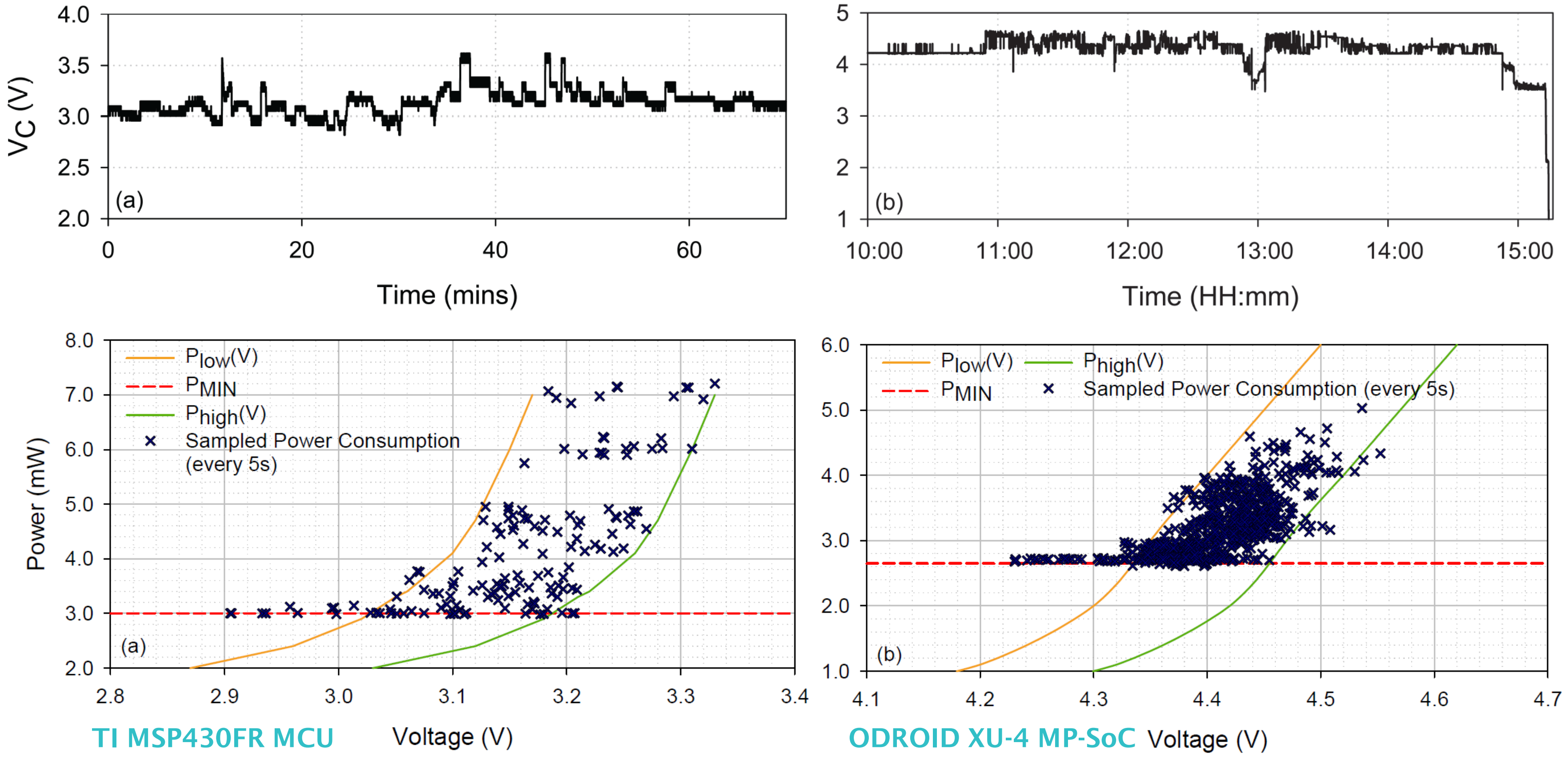




EXPERIMENTAL RESULTS

Operating from a PV Energy Harvester

- Software-Based MPPT



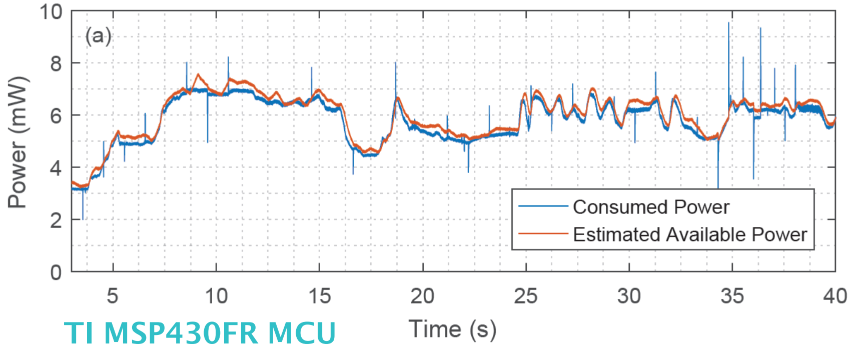
TI MSP430FR MCU

ODROID XU-4 MP-SoC

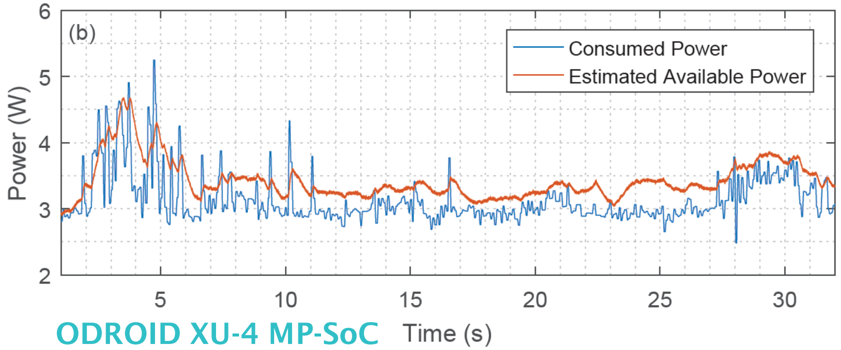
EXPERIMENTAL RESULTS

Operating from a PV Energy Harvester

- Power-Neutral Behaviour



TI MSP430FR MCU



ODROID XU-4 MP-SoC

- Application Forward Progress

Performance Scaling Technique	Number of FFTs per second
Static Approach [16]	1.07
MCU Power-Neutral Approach [17]	1.65
Momentum	1.83

TI MSP430FR MCU

Performance Scaling Technique	Billions of Instructions Per Second
Linux Powersave	0.69
MP-SoC Power-neutral Approach [2]	1.17
Momentum	1.27

ODROID XU-4 MP-SoC

CONCLUSIONS

- We need to rethink the way that we design self-powered systems
- Lots of existing approaches to help with this…
- …but lots of challenges

YOUR QUESTIONS

Professor Geoff Merrett | gvm@ecs.soton.ac.uk
Head of Centre for IoT and Pervasive Systems

UNIVERSITY OF
Southampton