Measuring the energy consumption of embedded software

Steve Kerrison

Senior Research Associate in Energy Aware System Design
Department of Computer Science, University of Bristol

NiPS summer school 2016

In this demo

A thirty minute session of:

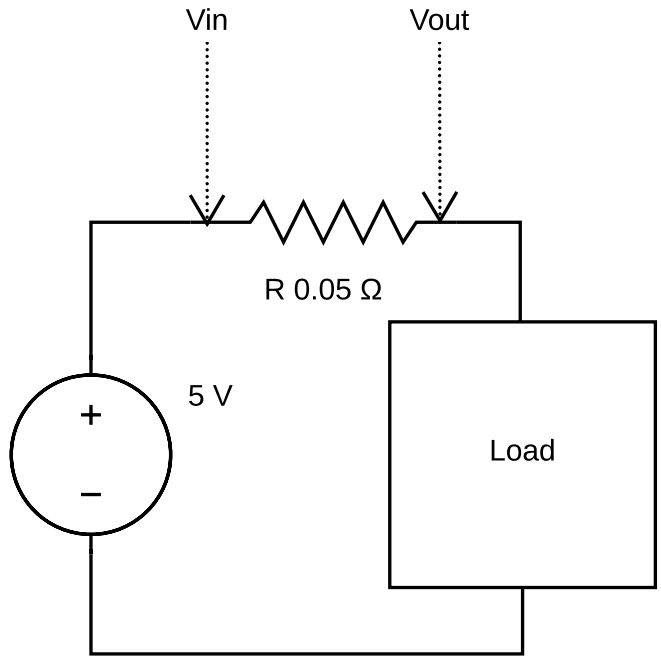
- Overview of energy measurement essentials (theory and practice).
- Example measurement platform: MAGEEC measurement board.
- Tools demo: MAGEEC + RaspberryPi.
- Interactive session experimenting with the Pi.

Funded by:

ICT-Energy - Co-ordinating Research Efforts of the ICT-Energy Community (Project number 611004).

Energy measurement: Current sense

- Sample the voltage drop across a resistor of known resistance.
- Use this to determine the current and instantaneous power.
- Repeatedly sample to determine average/peak/min power, total energy.
- Assume the current measured is for something *interesting*.



The maths:

$$P = VI, \qquad V = IR, \qquad I = \frac{V_{in} - V_{out}}{R}$$

$$\text{Eg:} \quad R = 0.05, \quad V_{in} = 5.0, \quad V_{out} = 4.975$$

$$\therefore \quad I = 0.05, \quad P = 0.25\,W$$

Energy measurement: Tooling

To make this work, you need:

- A suitable device to test (fairly low power - think safety!)
- A sense chip (ADC).
- Appropriate shunt resistor (wattage & range).
- Fast enough sample period (KHz, MHz, ?).
- Suitably design/modified target board.
- Data collection & sample triggering HW/SW.

Energy measurement: Tooling

To make this work, you need:

- A suitable device to test (fairly low power - think safety!)
- A sense chip (ADC).
- Appropriate shunt resistor (wattage & range).
- Fast enough sample period (KHz, MHz, ?).
- Suitably design/modified target board.
- Data collection & sample triggering HW/SW.

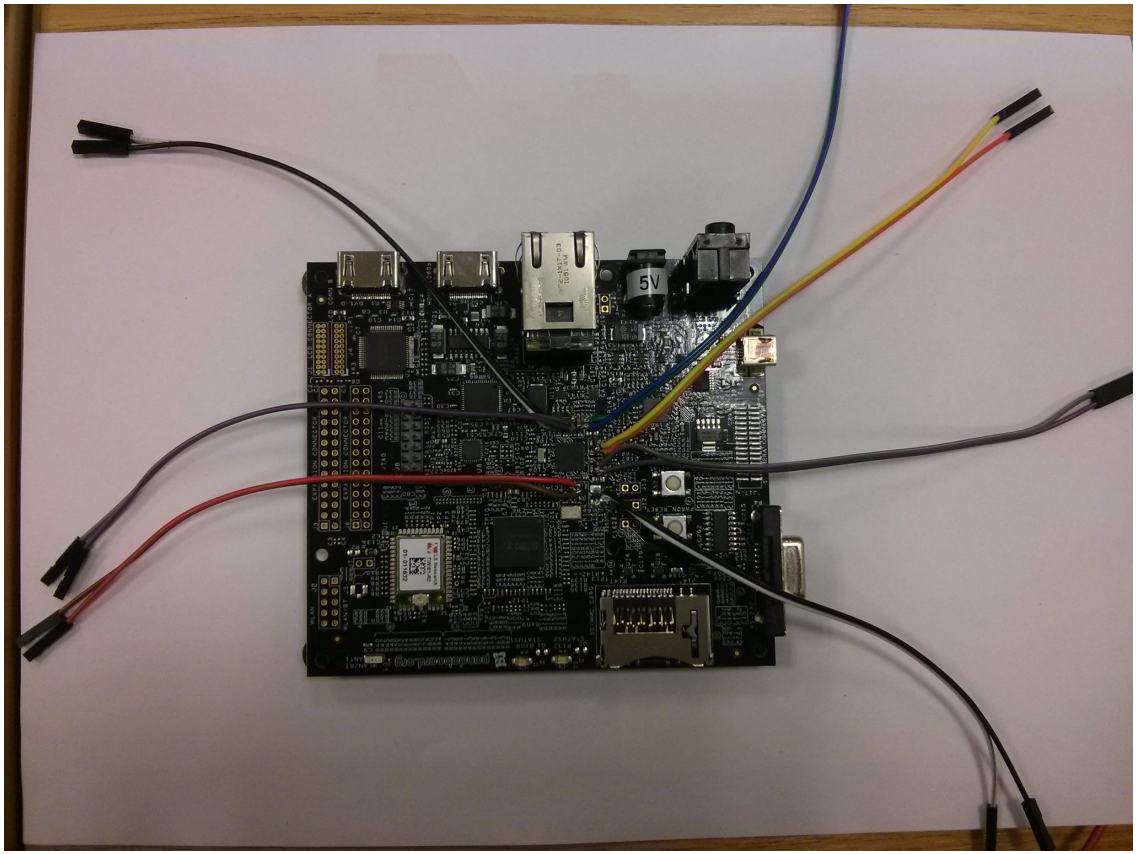
Should you do it all yourself?

Energy measurement: Tooling

To make this work, you need:

- A suitable device to test (fairly low power - think safety!)
- A sense chip (ADC).
- Appropriate shunt resistor (wattage & range).
- Fast enough sample period (KHz, MHz, ?).
- Suitably design/modified target board.
- Data collection & sample triggering HW/SW.

Should you do it all yourself?



MAGEEC measurement board

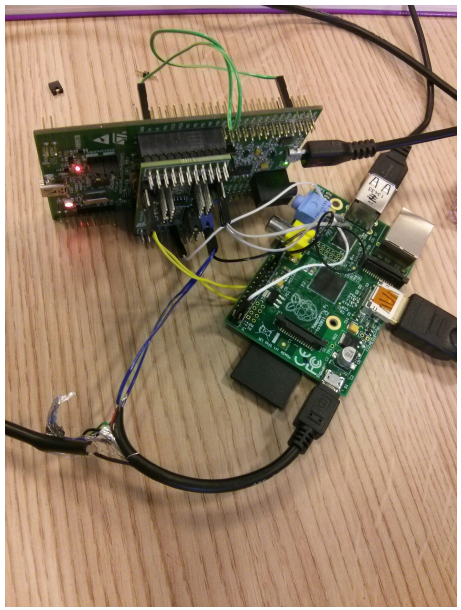
- MAchine Guided Energy Efficient Compilation project, needed measurement equipment.
- Developed a board that plugs into the readily available *STM32F4DISCOVERY* system.
- Captures to PC using USB and the `pyenergy` software library.
- Samples up to three power sources concurrently.
- Most of the processing done on the Discovery board.
- Configurable resistor sizes, inclusion of an inductor.

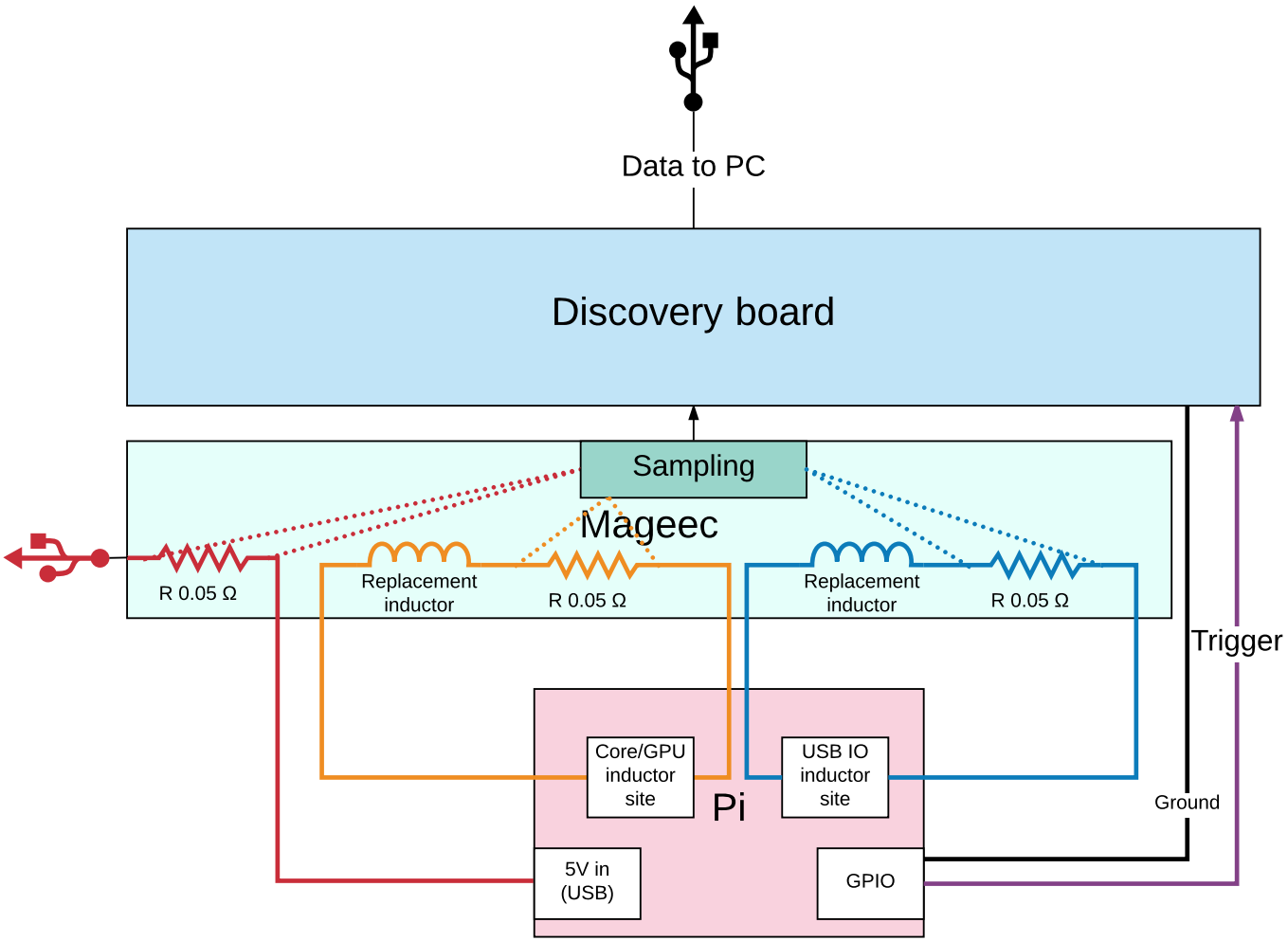
Resources

Discovery board: `http://tinyurl.com/STM32F4DISCOVERY`
MAGEEC board details: `http://mageec.org/wiki/Power_Measurement_Board`
MAGEEC board designs: `https://github.com/mageec/powersense-shield`
PyEnergy: `https://github.com/jpallister/stm32f4-energy-monitor`

Measuring a Raspberry Pi

Contents: 1x Pi, 1x Discovery, 1x MAGEEC board, 1x PC, associated software, cables, and some hardware modifications. . .





Tips

- Get V_{in} and V_{out} the correct way around, or results may be strange.
- Make sure measurement equipment can handle input voltage - depends on equipment!
- Choose a good shunt resistor value. Too small: High measurement noise. Too large: Voltage droop crashes device under test.

Tips

- Get V_{in} and V_{out} the correct way around, or results may be strange.
- Make sure measurement equipment can handle input voltage - depends on equipment!
- Choose a good shunt resistor value. Too small: High measurement noise. Too large: Voltage droop crashes device under test.
- Trigger measurements precisely to avoid noise from uninteresting activity.
- Ensure trigger period is long enough to connect sufficient samples.
- Run multiple measurement runs - check variance, choose a good strategy for managing quality of results.
- Try to combine "global" and "local" measurements to figure out exactly *where* energy is consumed.
- **Be safe!** Avoid batteries and higher voltage/current devices - other measurement methods exist for bigger systems.

Demo time!

Demo steps

- GUI and command line demonstration.
- Exploring the different sampling points.
- Looking at triggering.
  And then...
- Q&A, ad-hoc experiments.

End

With thanks to:

- Simon Hollis, hardware design.
- James Pallister, software tools.
- Embecosm, MAGEEC project partner.

Questions, comments, ideas & proposals:

Contact Steve Kerrison or Kerstin Eder,

`firstname.lastname@bristol.ac.uk`