



Green Gateway Project CSD Fall 2015

Markus Hidell, <mahidell@kth.se>
Robert Olsson, <roolss@kth.se>

Problem: Power-Use

Gateways for wireless networks are too power hungry for many applications. Although consuming only a few Watts this is magnitudes more than sensor nodes (motes).

Example of idle power use :

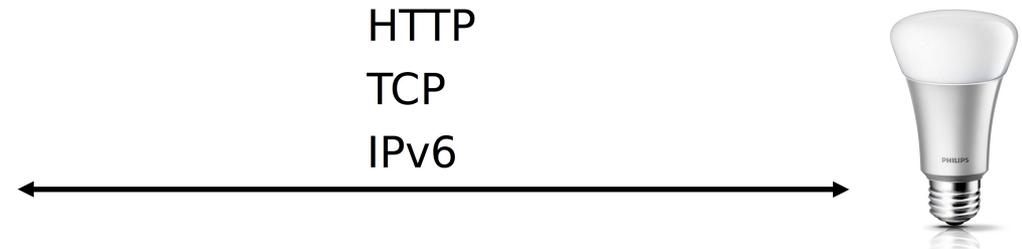
Mote: 40uW @ 3V

Gateway: 1W = 1000000uW @ 5V

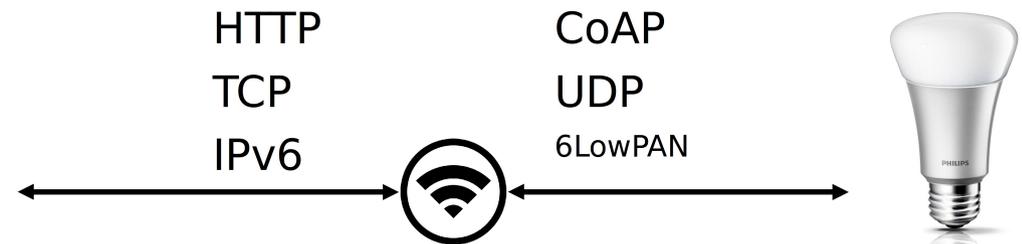
Initial goal: half the power consumption for GW

Integration into the Internet

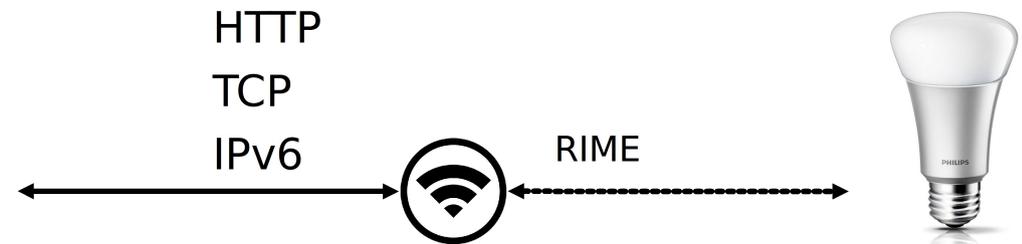
Internet protocols end-to-end



Gateway translates between two standardized internet protocol stacks.



Gateway translates between two (standardized) protocol stacks.



Motivation: Low-power GW platform

Sensors deployed in areas with limited power

Sensor gateway crucial point

Lots of hardware and software platforms available

Evaluation needed

Objectives: Low-power GW platform

Use with different applications

Robust, rugged

Low-power, renewable power sources

Open design, open source

Off-shelf components

Affordable price

Uplink support for 3/4G & wired Ethernet

Objectives: Low-power GW platform



Approach: Survey of hardware

Rpi, BB, Odroid etc small (Project focus)

Other:

Android, telephones/OTG

Arduino

Development boards, AVR, PIC, ARM, STM

Knowledge & skills needed

Operating system, Linux kernel build

Programming, C, bash

Communication, IP

Embedded system experinence

Electronics basic skills

Hands-on work is required.

Basic tasks

Litterature study
Procurement if needed

Installation, HW / SW gateway
Installation energy monitoring
Installation WSN

Development & Research

Component level
System level

Power & Performance & Robustness

Examples:

- Sleep-Modes HW.
- Power-Save Linux.

Duty cycling:

- Radio 3g/4g modem
- On-demand GW
- Suspend/Resume

How measure performance and robustness?

Final design

Test
Verification
Deployment
Demo
Report
Hand-Over. Repository

Project demo: deployment in Electrum!

Practical Example

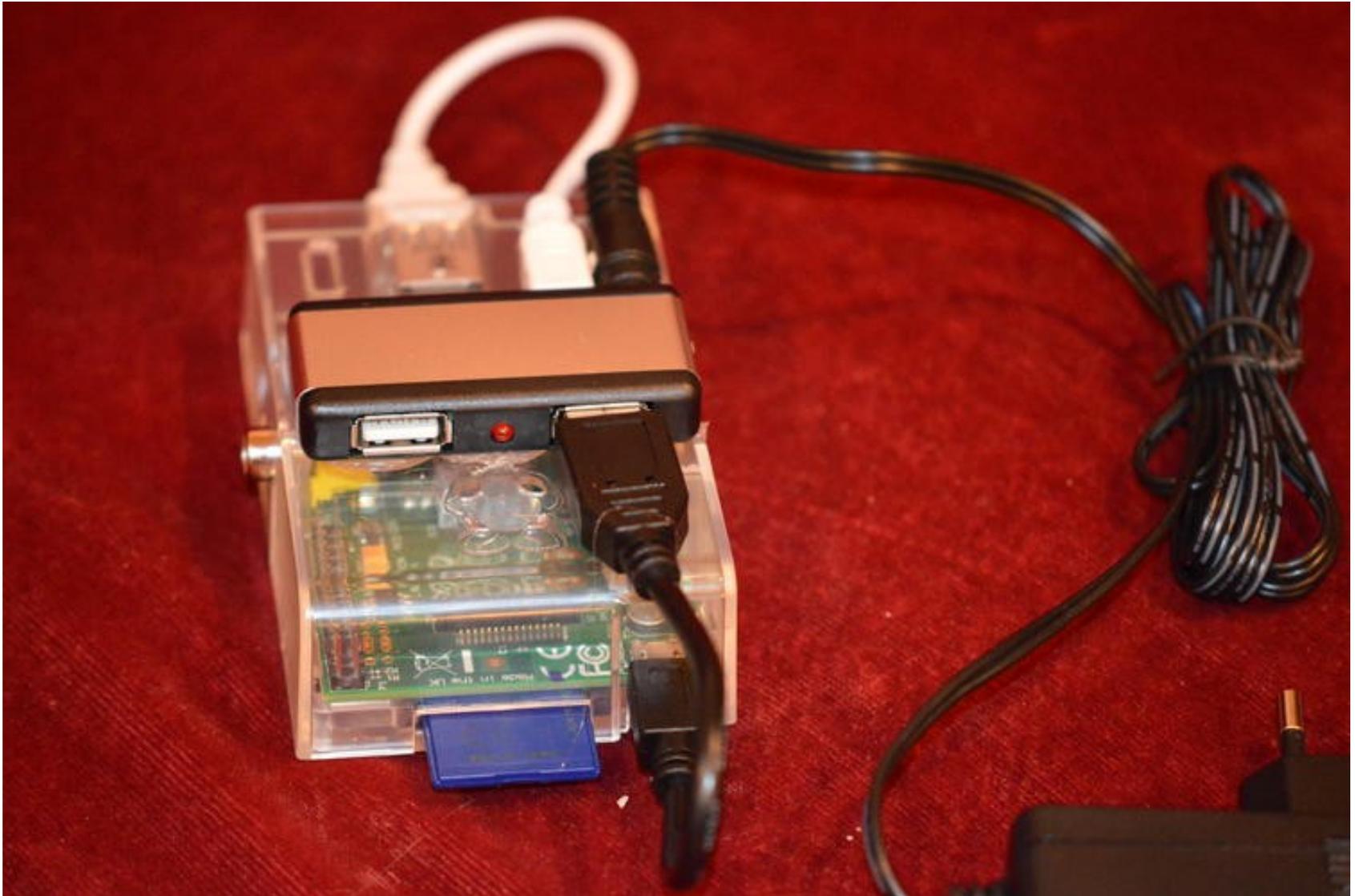
Automatic Weather Station (AWS) prototype.

An AWS is based on 4 WSN-nodes. 10m, 2m plus ground and sink nodes. Sink node is connected to a GW Rpi via USB.

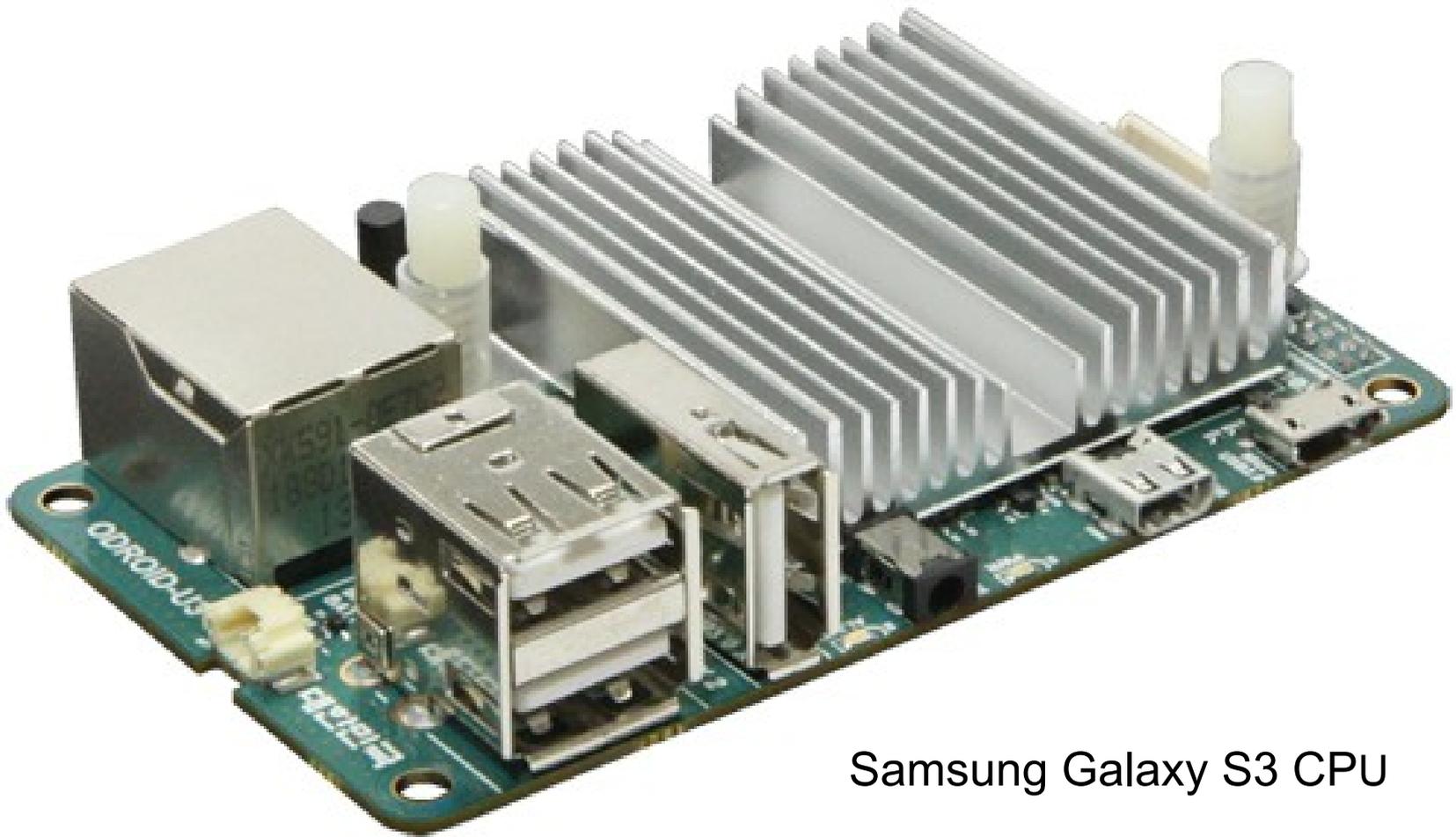
Data packets (reports) are sent every minute from each node. 10m node reports 3 pkts/min. An IEEE 802.15.4 is max 127 bytes.

GW uses AC-power today.

RPI & USB hub unit

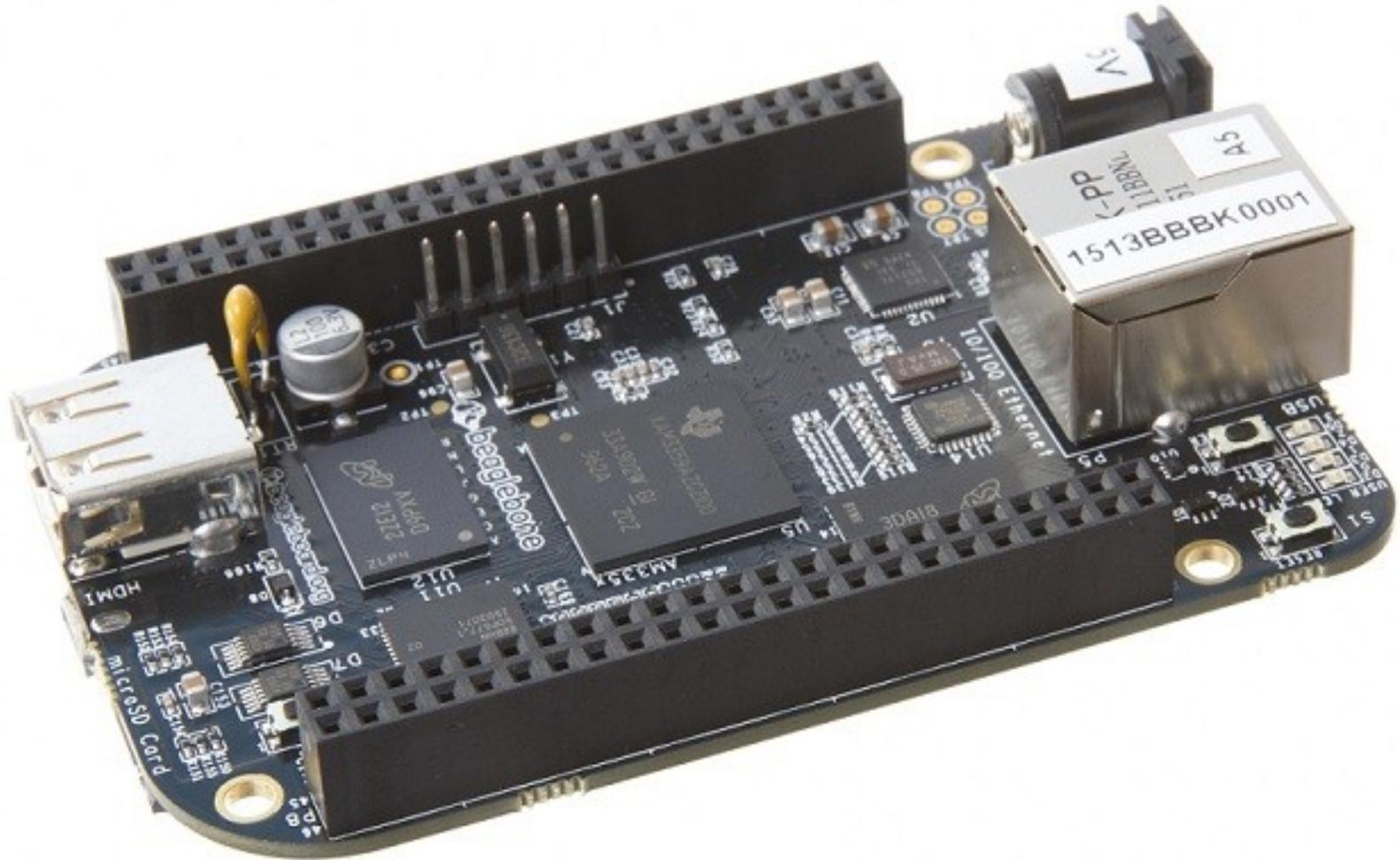


Odroid 1.7 GHz 4 cores

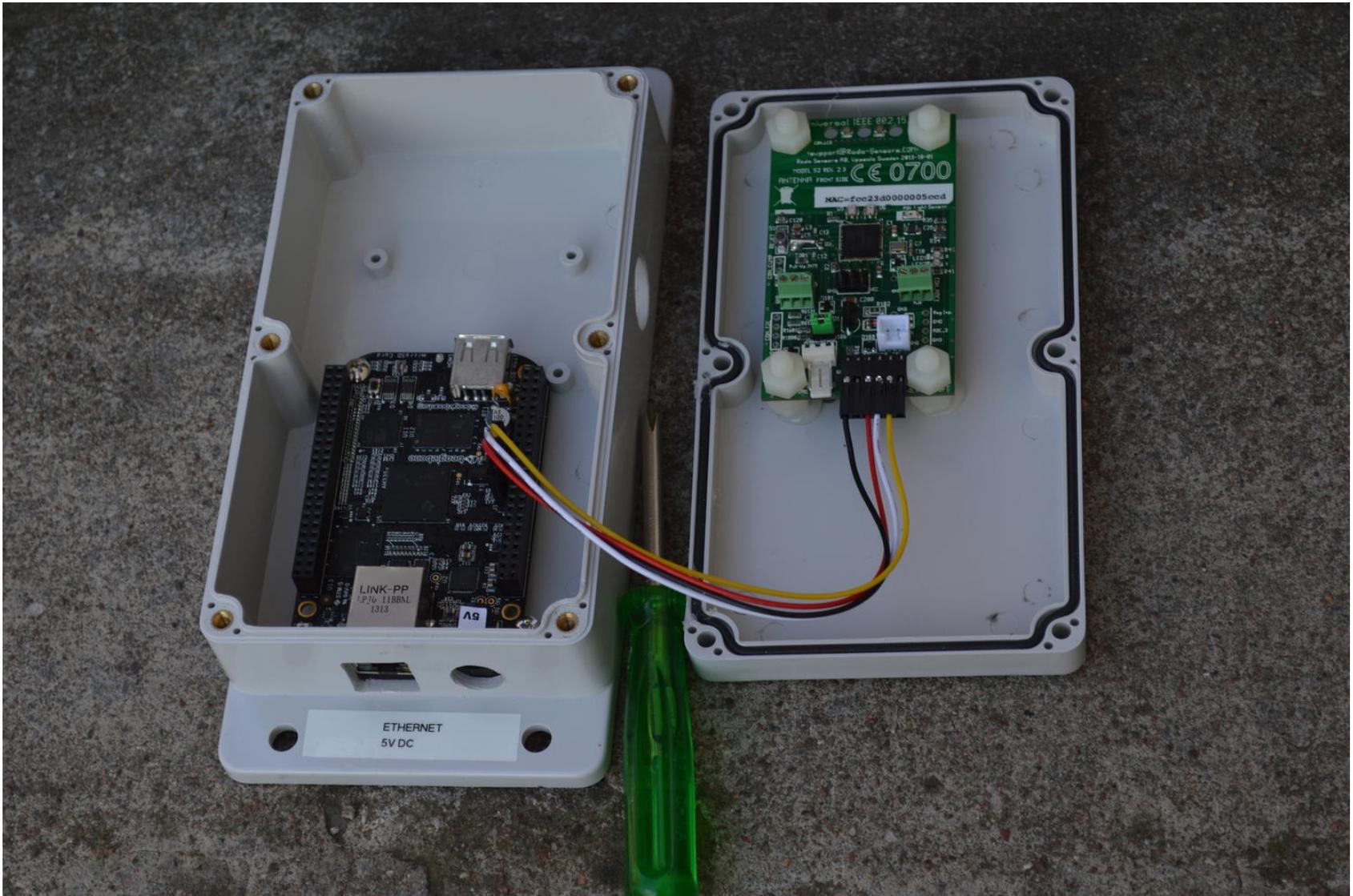


Samsung Galaxy S3 CPU

Beaglebone Black, TI SoC



GW assembly prototype



Intel Quark @ 2.2W

Quark™ SoC X1021

(16K Cache, 400 MHz)

512MB DDR3 ECC

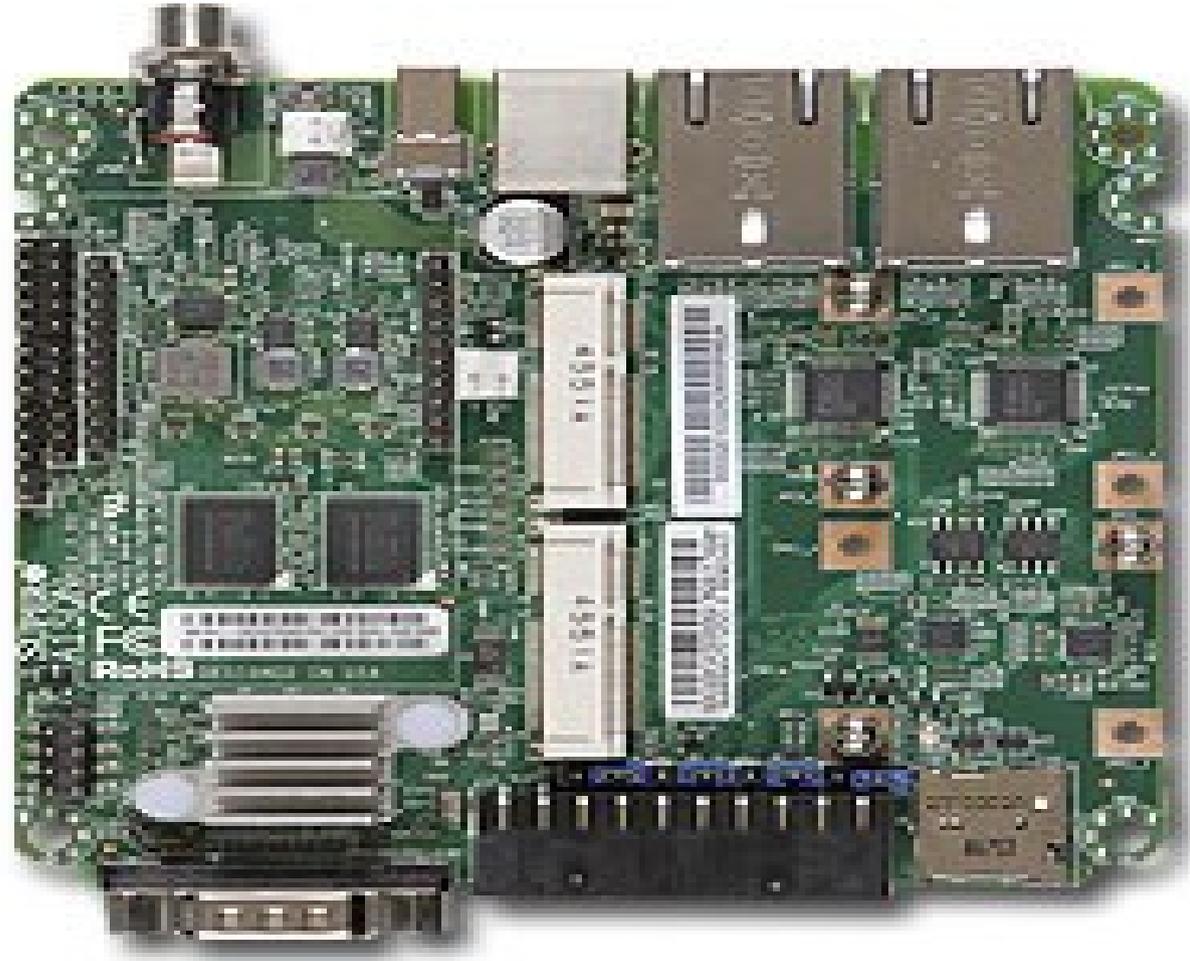
2x Mini-PCI-E slots;

1x ZigBee module socket

2x 10/100Mbps LAN

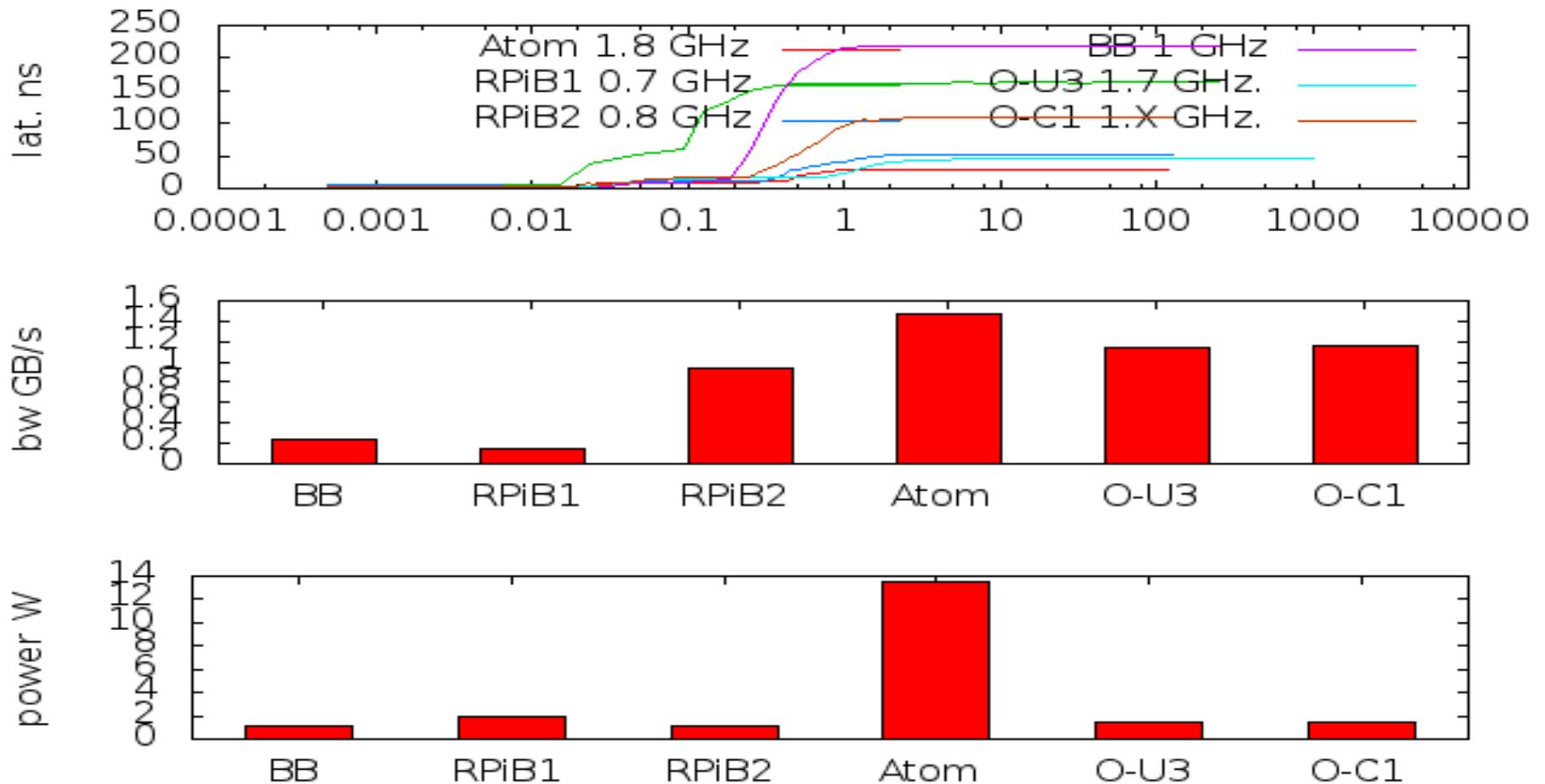
32 bit

1 Core



performance vs power

Mem. latency, mem. bandwidth & idle power. Plot rev 1.7



Public respostory for collaboration

Repository essential

Git, CVS, SVN

Public github.com

Suggestion github

References

The One Watt Initiative IEA in 1999

WIMEA RC3 project site.

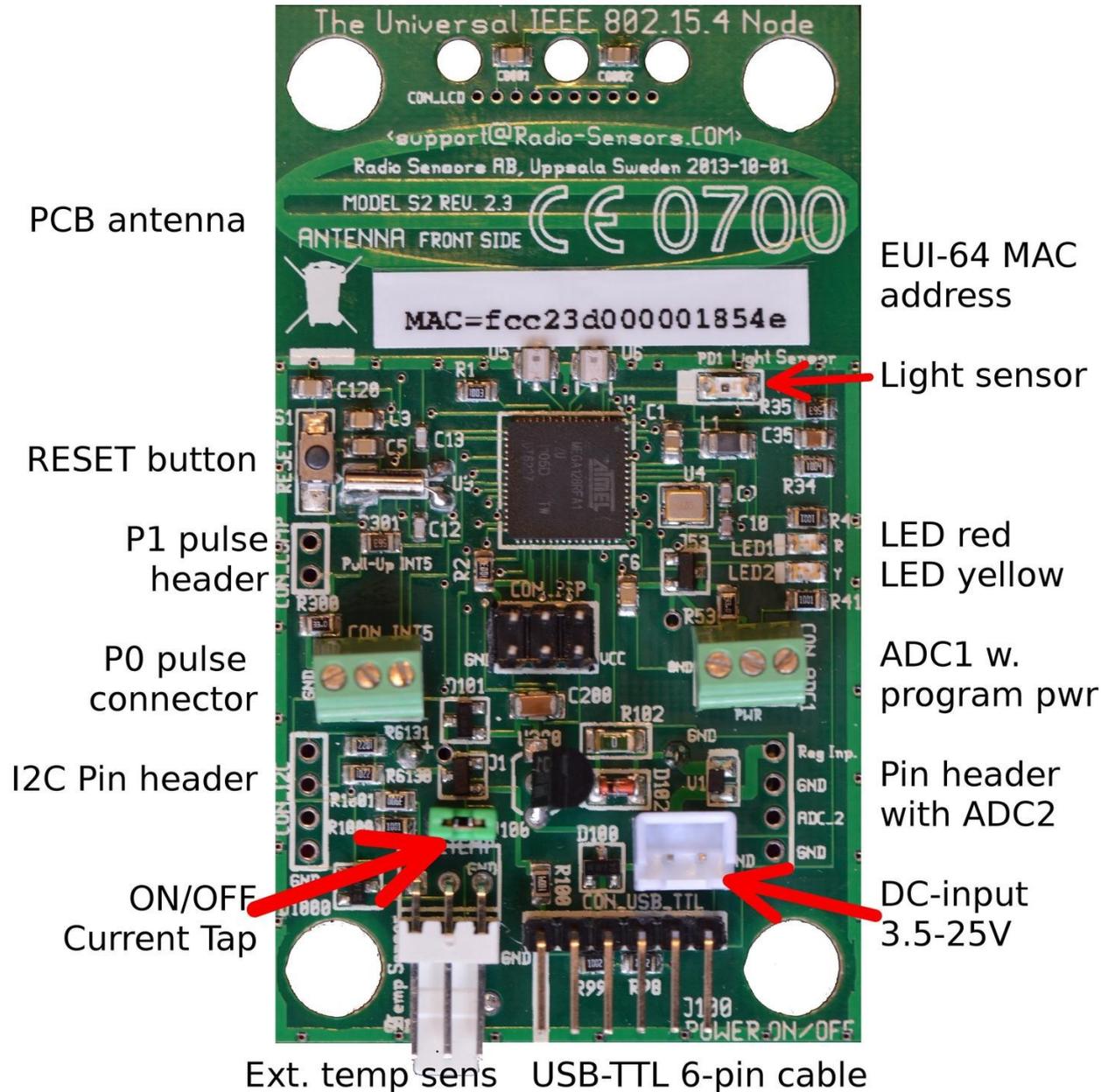
<http://www.se.ampr.org:8080/>

The Sleepy Pi Project/Shield

END

Questions?

WSN IEEE 802.15.4 Sensor Node



I2C add-ons and break-out boards

Almost everything... Just some I2C examples

Accelerometer & gyro (MPU-6050)

Magnetometer (HMC5883)

Temp & RH (SHT21, SHT25)

Pressure (BMP180, MS5611)

Hi-RES AD converter (MCP3424)

Lightning (AS3935)

Gamma, beta radiation (RD3024) TTL/Pulse

break-out boards do it yourself?

Relatively easy... SHT25, MS5611, MCP3424 for WIMEA meteorological project.

