Piles réseau économes en énergie dans les Réseaux de Capteurs sans Fil

David Hauweele

Université de Mons

UMONS Faculté des Sciences

25 mars 2016

Internet of Things

Internet today

- Not only interconnected computers
- Mobile Internet (smartphones, tablets)
- Interconnected objects (sensors, automation, monitoring)

Internet of Things

Internet today

- Not only interconnected computers
- Mobile Internet (smartphones, tablets)
- Interconnected objects (sensors, automation, monitoring)

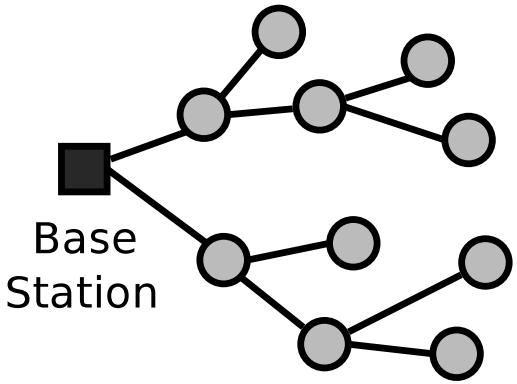
Definition

- The **Internet of Things** is a concept which seeks to integrate physical objects to Internet thanks to an adressing system which can uniquely identify them.
- These objects are generally equipped with *sensors* and *actuators* in order to interact with their environment and they have *limited* processing capabilities $\rightarrow$ *smart objects*.

Wireless Sensor Network

Features

- Wireless nodes
- Interconnected
- Common task



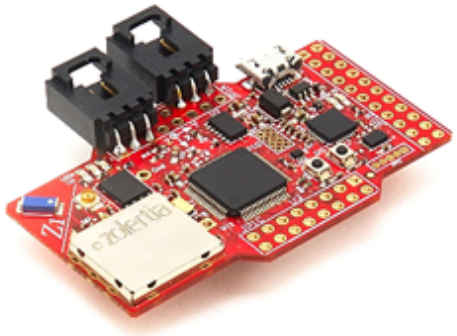
Application domains

- Environmental monitoring [11]
- Industrial applications [14]
- Health care [7]
- ...

$\Rightarrow$ **No single WSN design!**

Source: Automated irrigation [9]

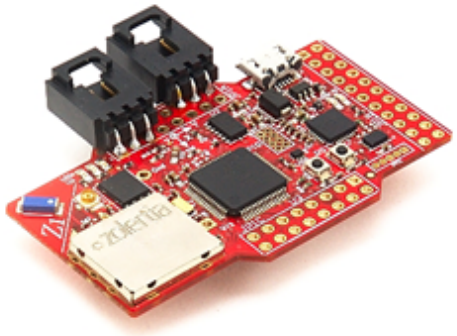
Typical wireless node



	Node	Smartphone
CPU	20 MHz	1000 MHz
RAM	8 kB	1 GB
Lifetime	years	days
Cost	10 €	200 €

Source: http://zolertia.sourceforge.net

Typical wireless node



	Node	Smartphone
CPU	20 MHz	1000 MHz
RAM	8 kB	1 GB
Lifetime	years	days
Cost	10 €	200 €

Problems

- Limited resources
- Energy constraints
- Single application

Need an optimized and modular network stack

Source: http://zolertia.sourceforge.net

Lifetime

How can we achieve years of longevity?

- Low-power micro-controllers and radio
- Energy harvesting techniques
- *Energy aware communication protocols*

Lifetime

How can we achieve years of longevity?

- Low-power micro-controllers and radio
- Energy harvesting techniques
- *Energy aware communication protocols*
  - **Link layer:** RDC, dedicated MAC [12, 3, 6]
  - **Network/transport layer:** 6LoWPAN [10], Rime [5]
  - **Routing:** RPL [15]
  - **Application layer:** CoAP [13]

Lifetime

How can we achieve years of longevity?

- Low-power micro-controllers and radio
- Energy harvesting techniques
- *Energy aware communication protocols*
  - **Link layer:** RDC, dedicated MAC [12, 3, 6]
  - **Network/transport layer:** 6LoWPAN [10], Rime [5]
  - **Routing:** RPL [15]
  - **Application layer:** CoAP [13]
- *Energy aware task scheduling*

Radio Duty Cycle

Radio dominates node power consumption

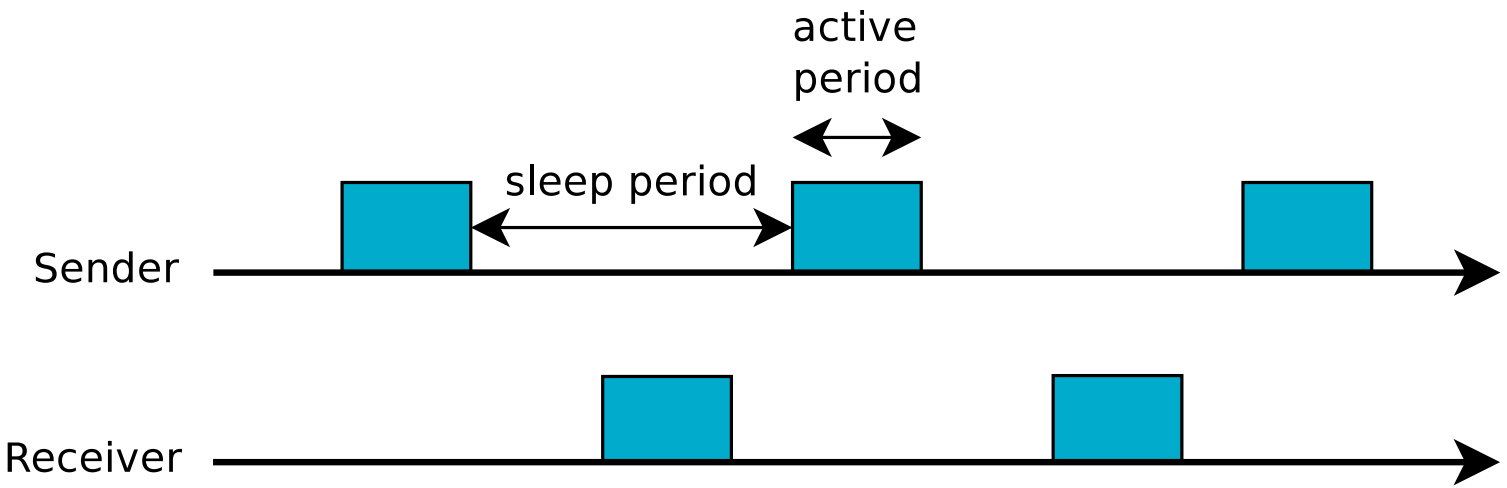
- Transmission and reception $\approx$ 25 mA
- Battery 2500 mAh $\rightarrow$ only 4 days lifetime
- Sleep mode $\approx$ 1 μA

Radio Duty Cycle

Radio dominates node power consumption

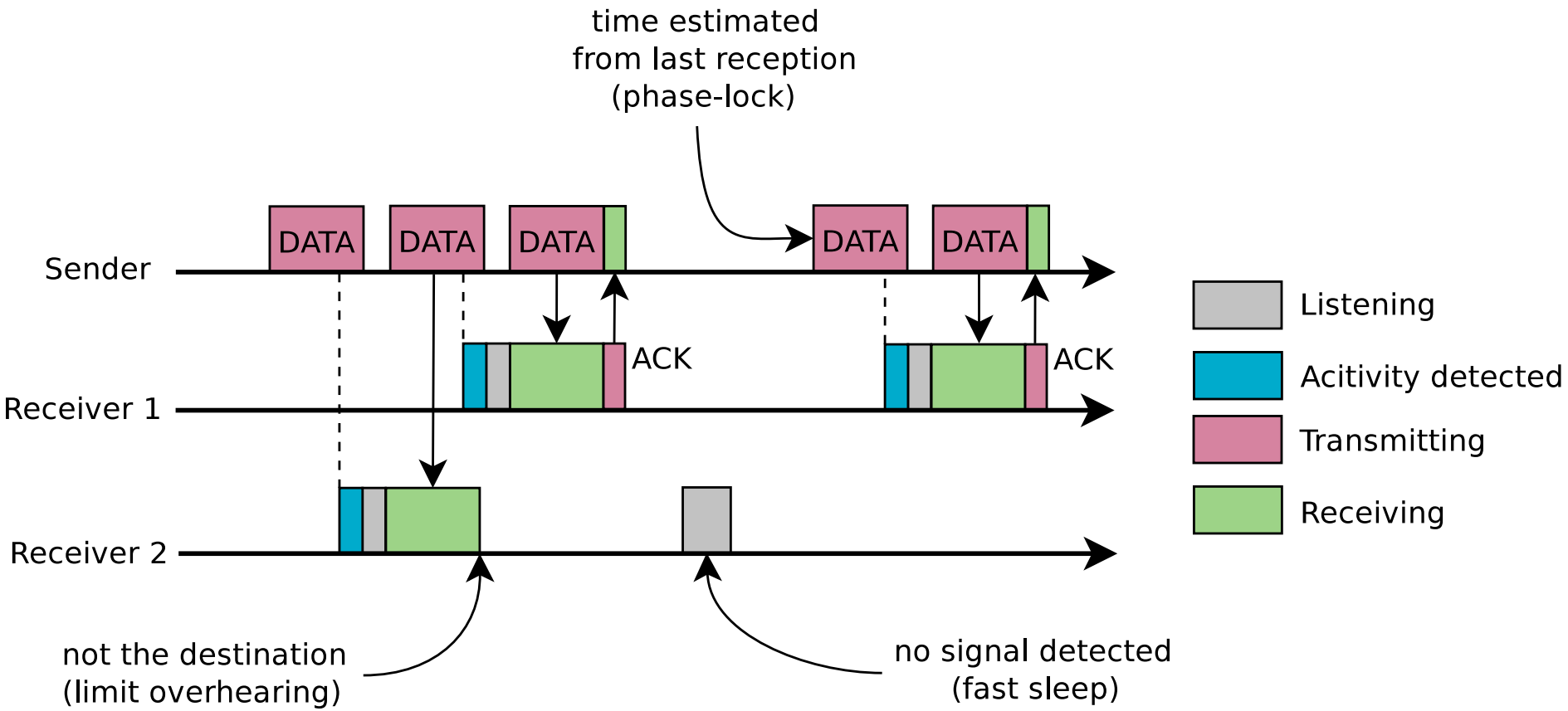
- Transmission and reception $\approx$ 25 mA
- Battery 2500 mAh $\rightarrow$ only 4 days lifetime
- Sleep mode $\approx$ 1 μA

Solution : leverage the sleep mode $\rightarrow$ Radio Duty Cycle



Phase lock

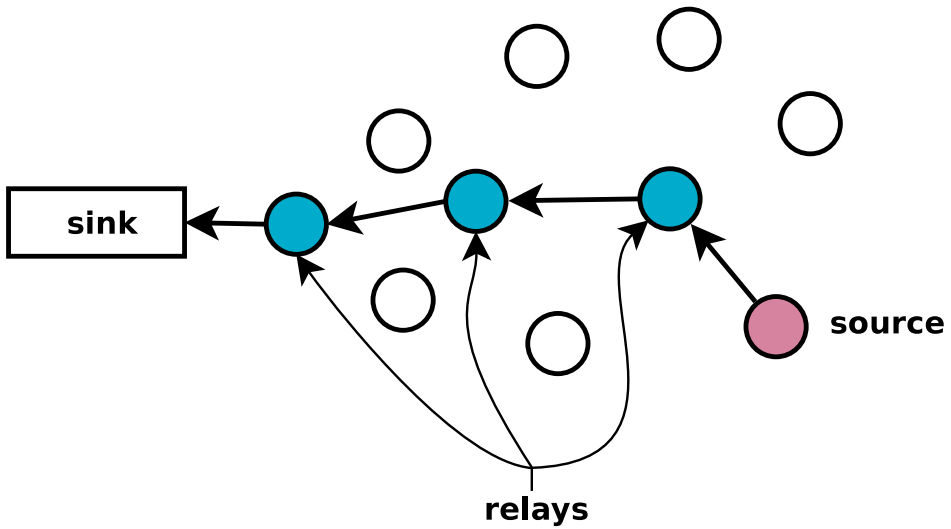
How to talk with a neighbor that sleeps 99% of time?



Multi-hop Routing

Transmit messages over longer ranges

- Low-power radio $\rightarrow$ limited range
- Nodes can act as *relay* for other nodes
- Routing $\rightarrow$ compute lowest cost paths



What is the cost of a path in WSN ?

Interoperability

How to connect these wireless nodes to Internet?

- Physical and link layer $\rightarrow$ IEEE 802.15.4
- Network layer $\rightarrow$ ZigBee or network IP (IPv6)
- Upper layers $\rightarrow$ dedicated framework, CoAP or TCP/UDP/ICMP

Source: http://compixels.com

Interoperability

How to connect these wireless nodes to Internet?

- Physical and link layer $\rightarrow$ IEEE 802.15.4
- Network layer $\rightarrow$ ZigBee or network IP (IPv6)
- Upper layers $\rightarrow$ dedicated framework, CoAP or TCP/UDP/ICMP

We use IPv6 on top of the IEEE 802.15.4 standard

Source: http://compixels.com

6LoWPAN

Network layer IPv6

- Very large number of adresses ($2^{128} \approx 3.4 \times 10^{38}$)
- Directly exposed to Internet

Problems

The IPv6 network is optimized for use with efficient links (Ethernet, Wi-Fi, ...). This poses harsh contraints on the nodes used within a wireless sensor network.

6LoWPAN

Network layer IPv6

- Very large number of adresses ($2^{128} \approx 3.4 \times 10^{38}$)
- Directly exposed to Internet

Problems

The IPv6 network is optimized for use with efficient links (Ethernet, Wi-Fi, ...). This poses harsh contraints on the nodes used within a wireless sensor network.

Solution $\rightarrow$ 6LoWPAN [10]

- Adaptation layer between IPv6 and IEEE 802.15.4
- Adjust the IPv6 network to the performances of smart objects

RTOS

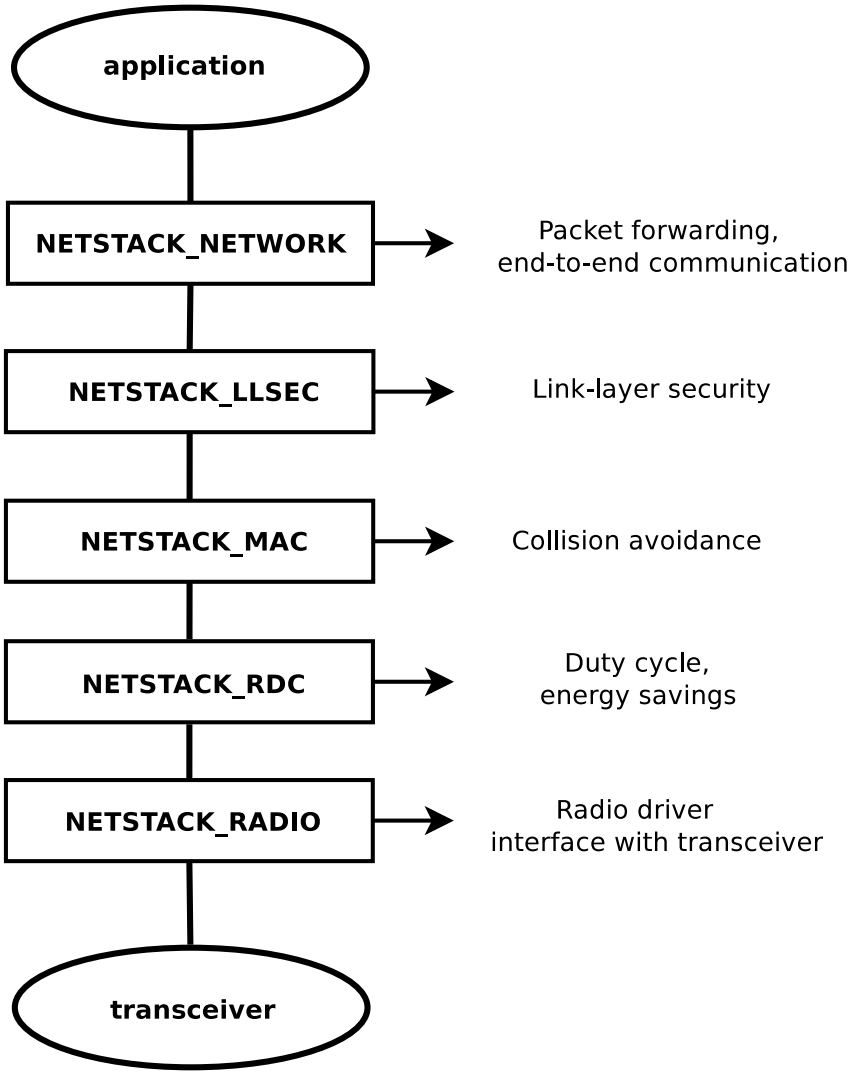
RTOS

The nodes of a wireless sensor network are programmed using specialized Real Time Operating Systems (RTOS) among which the most popular, Contiki [1] and TinyOS [2].

Example: Contiki

- Support IPv4, IPv6, 6LoWPAN, several MACs
- Lightweight network layer for IoT, RIME
- Event-driven cooperative scheduling
- Modularity
- Implemented in C

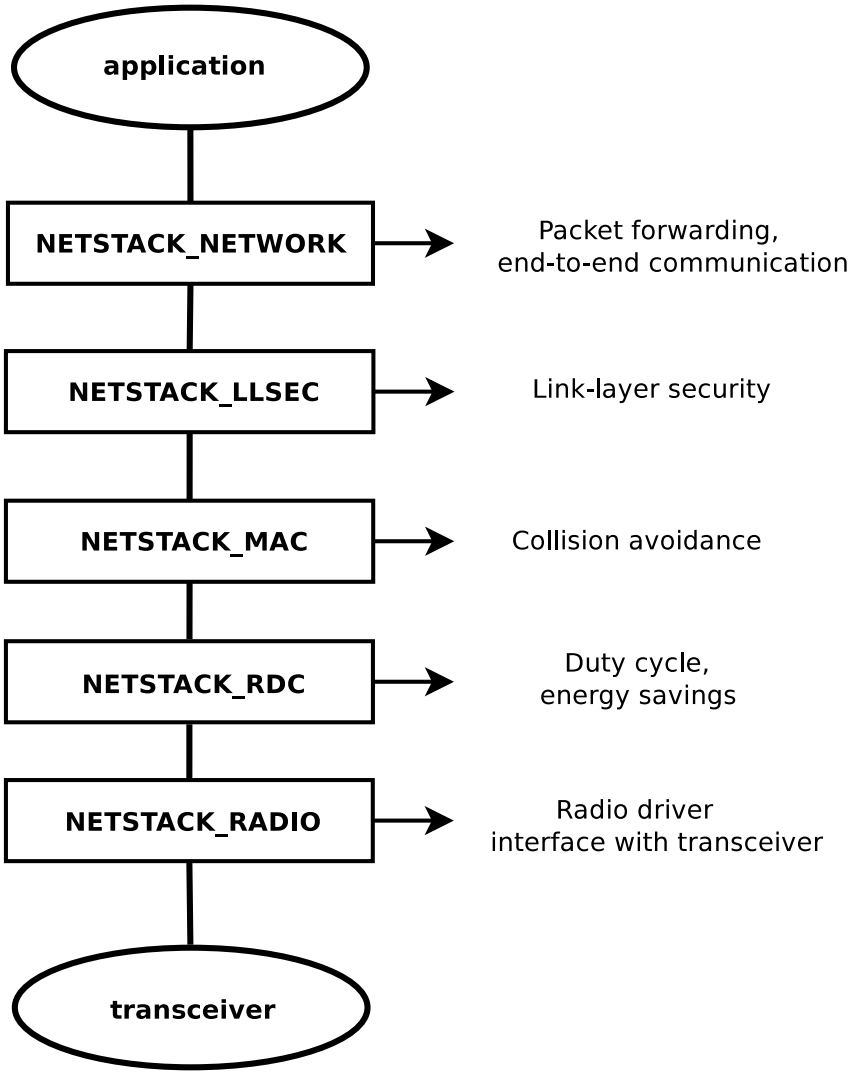
The Contiki network stack



Problems

- Limited Modularity
- Modules developed separately
- Complex code (low-level)

The Contiki network stack



Problems

- Limited Modularity
- Modules developed separately
- Complex code (low-level)

How can we optimize?

Complexity: the RIME layer

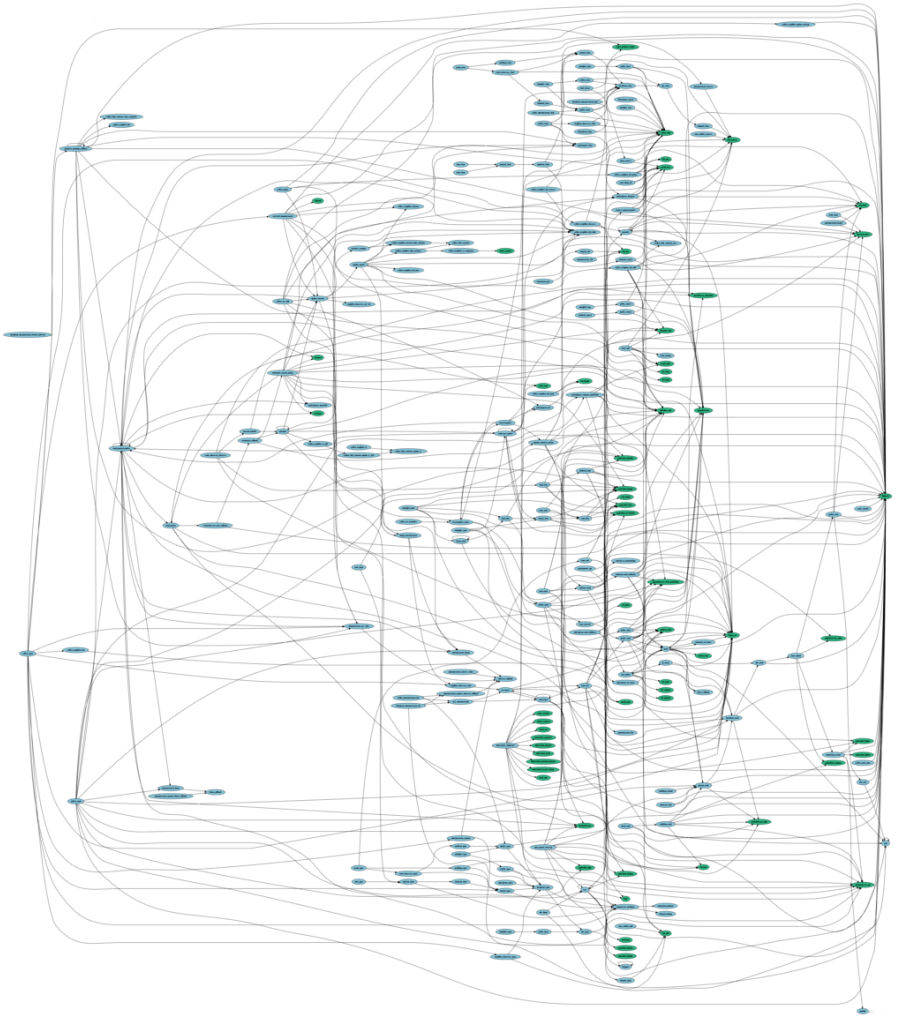


Figure: The RIME callgraph

Higher-level approach

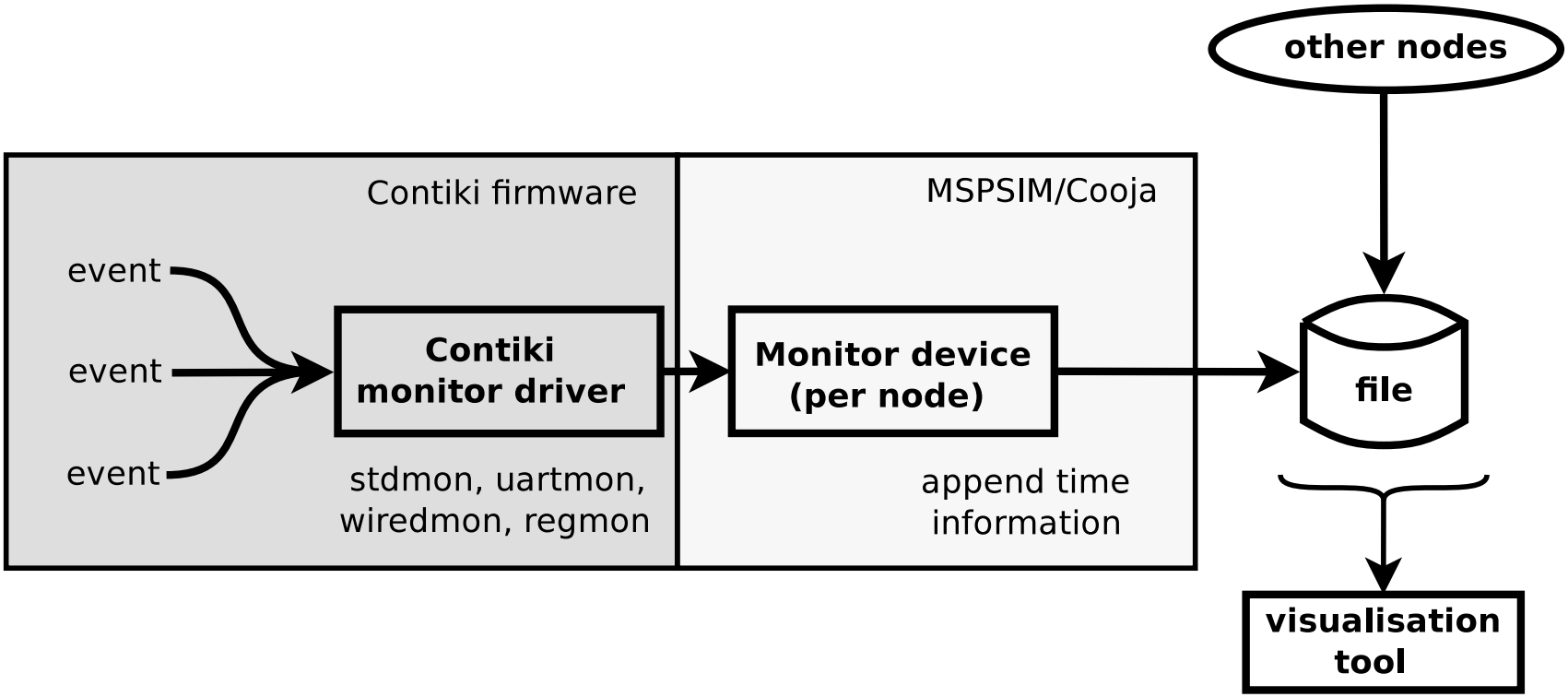
Need an inside view of the RTOS

- Monitor events in the RTOS
- Record the time of each event
- Correlation between events in different modules
- Better understanding of the network stack

Two approaches:

- *Hardware*: monitor devices changes in MSPSIM [4, 8]
- *Software*: monitor the firmware (and the netstack)

Monitor device



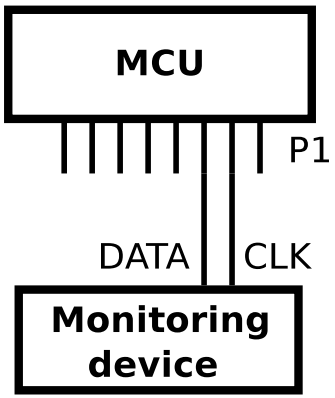
- Reference clock
- Minimize in-firmware time
- Find exact moment when an event occured

WiredMon and RegMon

Two implementations on MSP430 emulator MSPSIM

WiredMon

- Implemented like a real device
- 2-wires interface to monitoring device
- Quite slow

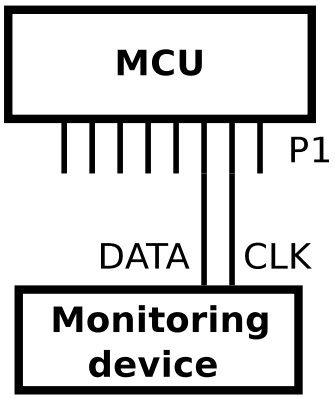


WiredMon and RegMon

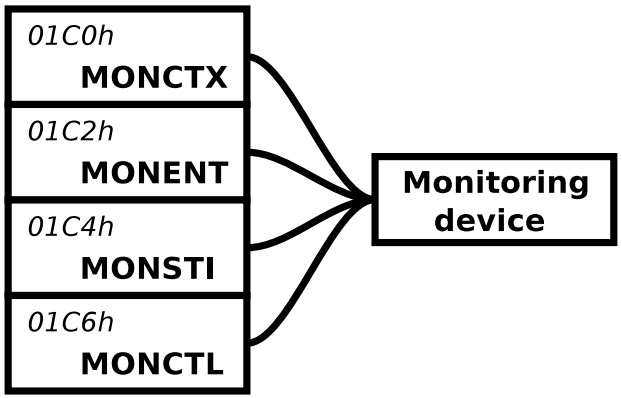
Two implementations on MSP430 emulator MSPSIM

WiredMon

- Implemented like a real device
- 2-wires interface to monitoring device
- Quite slow

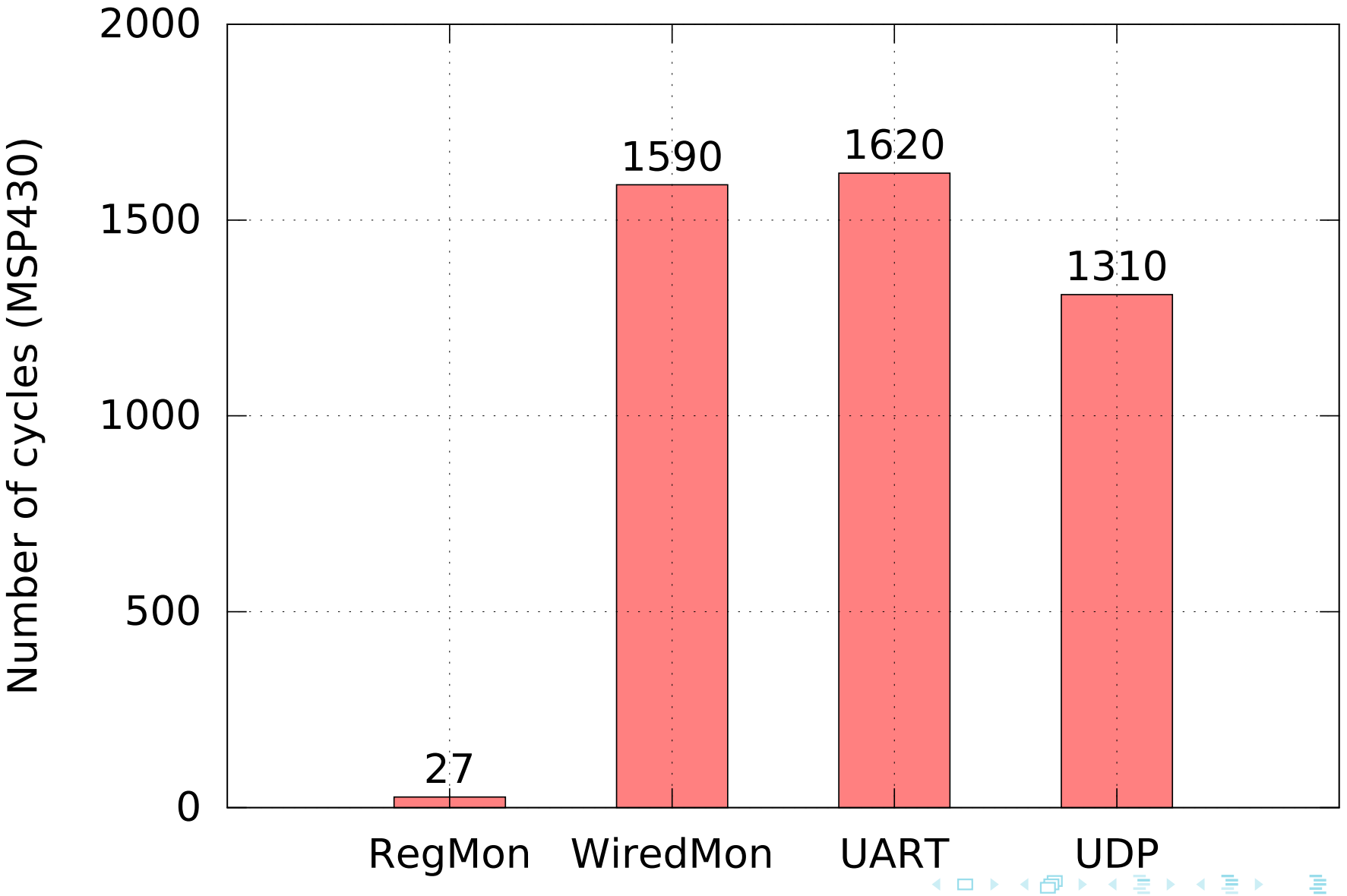


RegMon



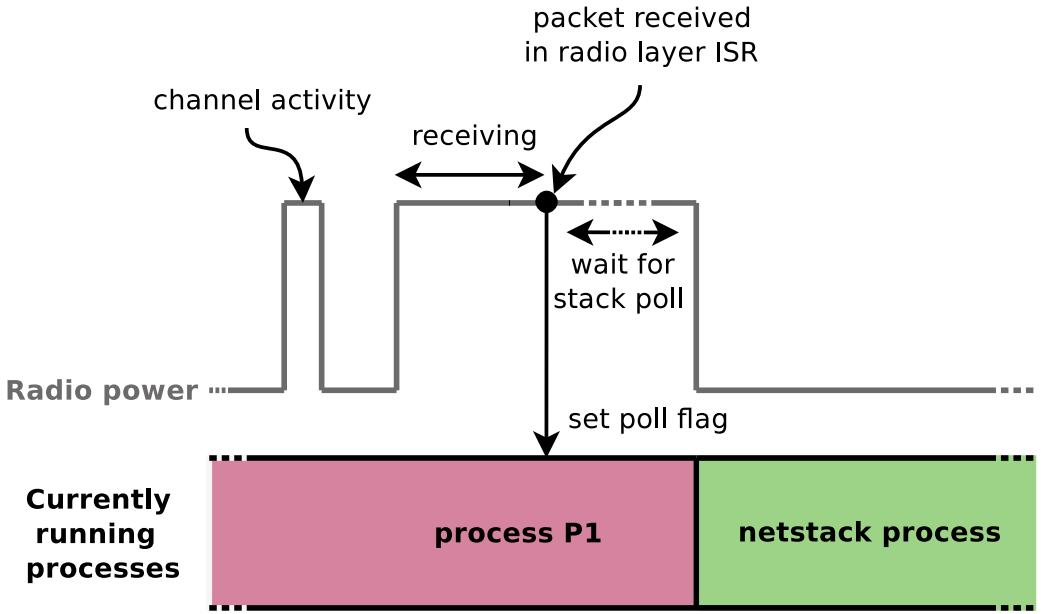
- Hack of the MSP430f1611
- 4 extra memory registers
- Really fast

Speed comparison



Example: delayed sleep

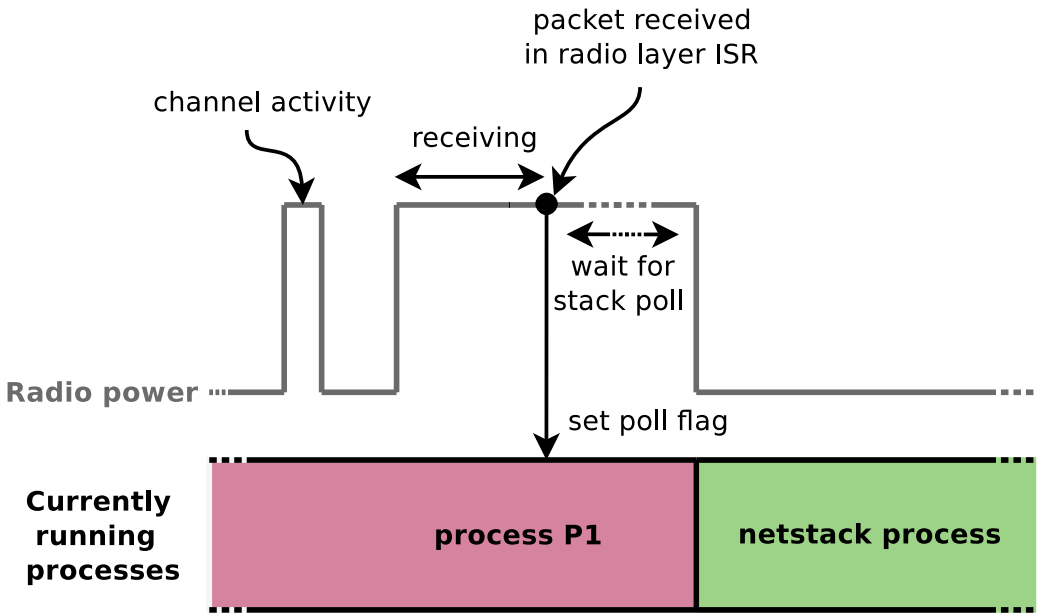
Receiving a packet in ContikiMAC



- Radio layer set poll flag on the stack process
- Poll flag checked after currently running process relinquishes CPU
- Radio stays on while currently running process still running

Example: delayed sleep

Receiving a packet in ContikiMAC



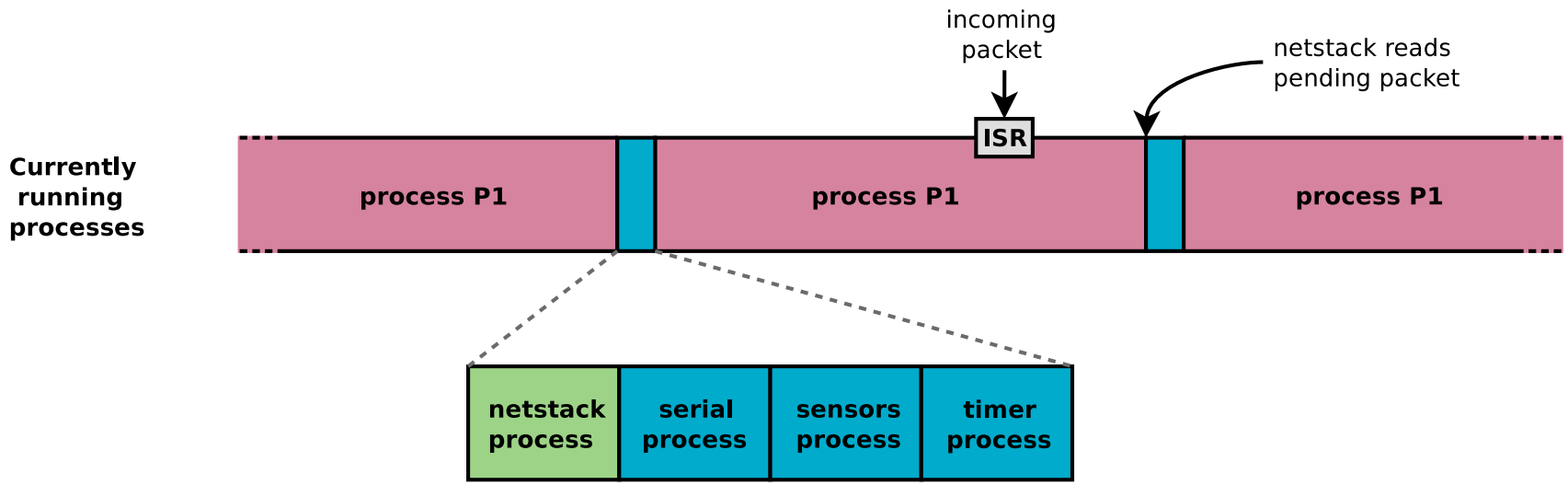
- Radio layer set poll flag on the stack process
- Poll flag checked after currently running process relinquishes CPU
- Radio stays on while currently running process still running

Solution $\Rightarrow$ cross layer optimization (RADIO/RDC layers)

Experiment

Delayed sleep experiment

- Two nodes: transmitter and receiver
- Transmitter sends one packet per second
- Receiver does some processing with process *P1*



Sleep delayed in average by half the working time of P1

Other optimizations

Pack short packets

- ContikiMAC imposes a lower bound on the frame size
- Short packet $\rightarrow$ padding
- Queue short packets and pack them together
- Reduce the number of transmissions

Optimistic timer

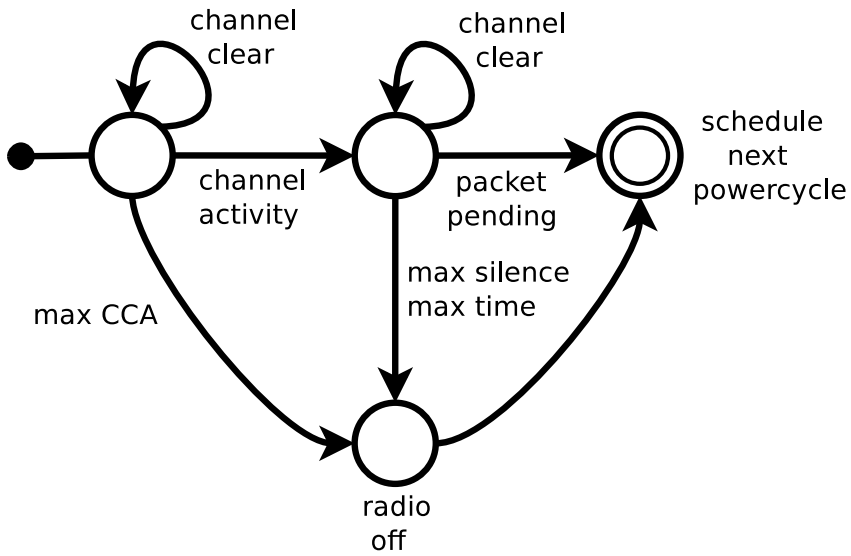
- Scheduler not aware of duty-cycle
- Trigger events at appropriate time
- Schedule timers with min/max time
- Allow the node to sleep for longer periods

Higher level approach

Monitor and visualisation tools → extract higher level model

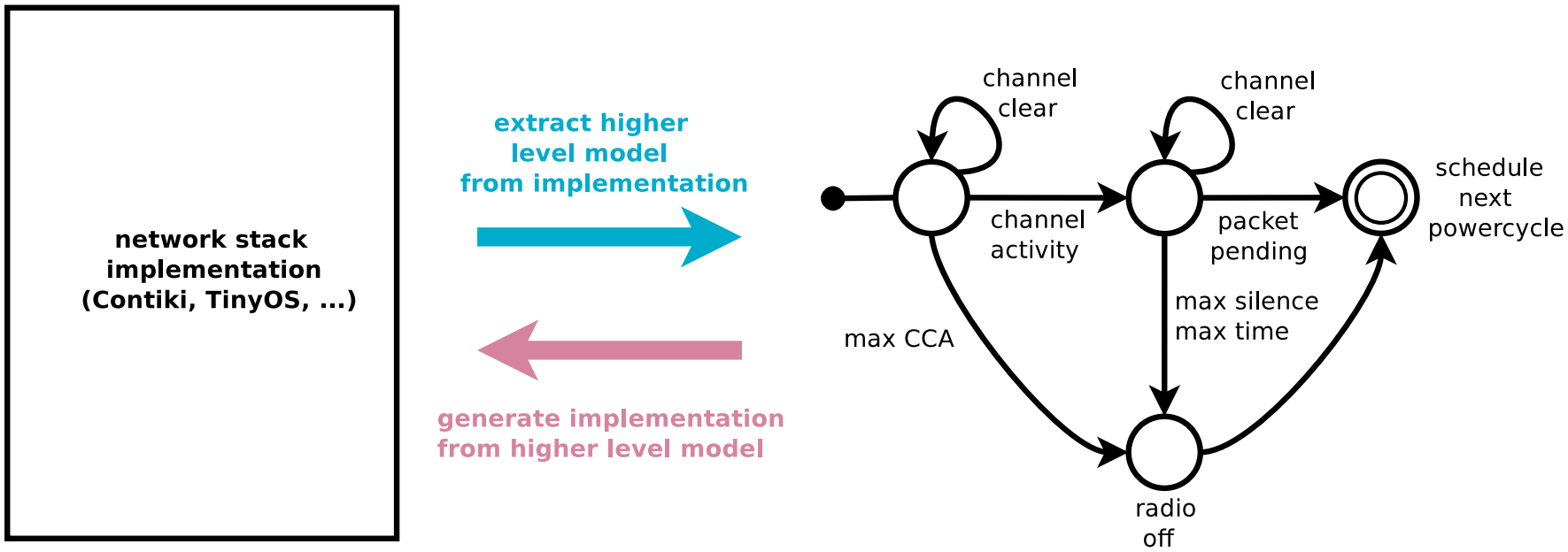
network stack implementation (Contiki, TinyOS, ...)

extract higher level model from implementation



Higher level approach

Monitor and visualisation tools → extract higher level model



Can we generate the implementation from a higher level model ?

Conclusion

WSN

- RTOS complex and low-level
- Energy constraints
- Need a higher-level approach for netstack optimization

Monitor

- Hardware and software approach
- MSP430 emulation in MSPSIM
- RegMon approach really fast
- WiredMon for real nodes but slow
- Visualisation tools

Conclusion

Optimizations

- Possible optimizations derived from the visualisation tools
- Delayed sleep
- Short packets
- Optimistic timer

Further work

- Generate implementation from higher level model
- Derive implementations from DSLs

Conclusion

Internet of Things

- Also data mining, security, ...
- Still many technical challenges
- 50 billions devices in 2020

Chances are that you will be developing code for the Internet of Things in the future!

Questions?

- **E-Mail:**

  david.hauweele@umons.ac.be
- **Site:**

  http://www.hauweele.net/~gawen
- **Slides:**

  http://www.hauweele.net/~gawen/files/sem-umons2016.pdf

References I


Contiki.
`http://www.contiki-os.org`, 2012.


TinyOS.
`http://www.tinyos.net`, 2012.


M. Buettner, G. V. Yee, E. Anderson, and R. Han.
X-MAC: a short preamble mac protocol for duty-cycled wireless sensor networks.
In Proceedings of the 4th international conference on Embedded networked sensor systems, pages 307–320. ACM, 2006.


W. Dron, S. Duquennoy, T. Voigt, K. Hachicha, and P. Garda.
An emulation-based method for lifetime estimation of wireless sensor networks.
In DCOSS, pages 241–248. IEEE, 2014.

References II

A. Dunkels.
RIME — a lightweight layered communication stack for sensor networks.
In Proceedings of the European Conference on Wireless Sensor Networks (EWSN), Poster/Demo session, Delft, The Netherlands. Citeseer, 2007.

A. Dunkels.
The ContikiMAC Radio Duty Cycling Protocol.
Technical report, 2011.
SICS.

E. Egbogah and A. Fapojuwo.
A survey of system architecture requirements for health care-based wireless sensor networks.
Sensors, 11(5):4875–4898, 2011.

References III

J. Eriksson, A. Dunkels, N. Finne, F. Osterlind, and T. Voigt.
MSPSim – an extensible simulator for msp430-equipped sensor boards.
In EWSN, Poster/Demo session, page 27, 2007.

J. McCulloch, P. McCarthy, S. M. Guru, W. Peng, D. Hugo, and A. Terhorst.
Wireless sensor network deployment for water use efficiency in irrigation.
In Proceedings of the workshop on Real-world wireless sensor networks, pages 46–50. ACM, 2008.

G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler.
Transmission of IPv6 packets over IEEE 802.15.4 networks.
RFC 4944 (Draft Standard), 2007.

References IV


L. M. Oliveira and J. J. Rodrigues.
Wireless sensor networks: a survey on environmental monitoring.
Journal of communications, 6(2):143–151, 2011.


J. Polastre, J. Hill, and D. Culler.
Versatile low power media access for wireless sensor networks.
In Proceedings of the 2nd international conference on Embedded networked sensor systems, pages 95–107. ACM, 2004.


Z. Shelby, K. Hartke, C. Bormann, and B. Frank.
Constrained application protocol (CoAP), draft-ietf-core-coap-13.
IETF, 2012.


X. Shen, Z. Wang, and Y. Sun.
Wireless sensor networks for industrial applications.
In Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on, volume 4, pages 3636–3640. IEEE, 2004.

References V

T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander.
RPL : IPv6 routing protocol for low power and lossy networks.
RFC 6550 (Draft Standard), Mar. 2012.