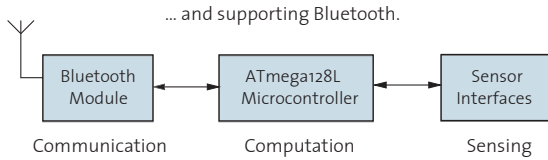


# Bluetooth Smart Nodes for Mobile Ad-hoc Networks

Jan Beutel, Oliver Kasten  
 Swiss Federal Institute of Technology (ETH) Zurich  
 8092 Zurich, Switzerland  
 beutel@tik.ee.ethz.ch, kasten@inf.ethz.ch

## A Distributed Platform for Sensor Nodes

A device magnitudes smaller than a PDA ...  
 equally flexible and programmable ...



Autonomous wireless communication and computing platform based on a Bluetooth radio module and a microcontroller.

### Smart everyday objects by attaching sensor nodes

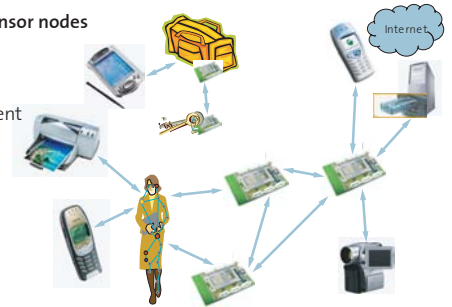
- Self aware
- Context sensitive
- Cooperative
- Integration into computing environment

### Ad-hoc networking scenarios

- Scalable multi-hop routing
- Integrated application protocols

### Consumer electronics integration

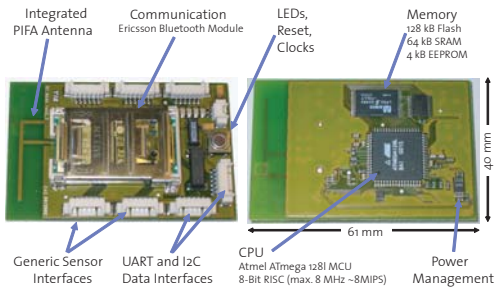
- Wearable and perceptual computing
- Backend connectivity



## BTnode - The Hardware Architecture in Detail

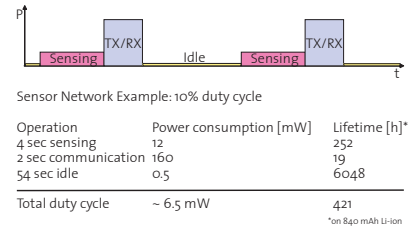
### Hardware Requirements

- Small form factor, low component count
- Standardized wireless interface
- Flexible and cost effective deployment of large quantities of networking nodes



Power Consumption [mW]	max	typ
Bluetooth Inquiry/CPU Active	250	160
Bluetooth Init/CPU Active	95	67
Bluetooth Off/CPU Idle	15	12
Bluetooth Off/CPU Sleep	6	<0,5

Bluetooth power consumption varies up to 300% depending on revision and operating mode.



### Testbed Deployment

- Current deployment ~200 units with 16 groups
- Unit cost \$ 110

The BTnodes have been developed and distributed to researchers in cooperation of the NCCR-MICS and the Smart Its Project.

## Event Driven System Software

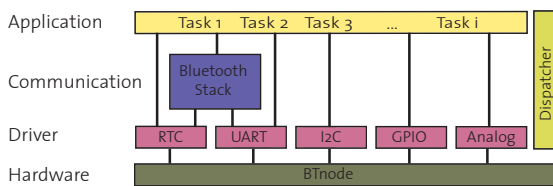
### System Software Overview

Lightweight OS made up of **drivers that are interrupt driven** and a **dispatcher for scheduling** application and driver tasks. Similar but simpler than TinyOS [Culler et al.].

### Software Features

- Low level drivers and libraries for peripherals and interfaces
- Event driven application model facilitates coarse grained cooperative multithreading
- 30 k codesize in ROM, 1-2k in RAM, with 128 byte UART buffers for communication

### Software Architecture



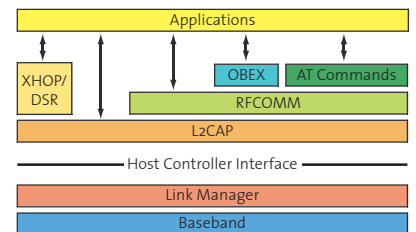
### Development Environment

- Standard C language GNU GCC crosscompiler development environment
- In system programmable and remote update over Bluetooth/bootloader
- Emulation environment on Linux
- Successful jumpstart in under a day

### Bluetooth Protocol Stack

- Baseband processing on subsystem
- Reduced host controller stack
- Elementary functions of link layer
- High level Bluetooth interface

Custom protocols can be easily integrated into this framework to support simple applications without RTOS knowledge.



## Some Demo Applications

### Multihop Source Routing on Bluetooth

- Reduction of the CMU DSR protocol allows a **message passing type multihop routing**
- Communication across the Bluetooth piconet borders supports >8 nodes
- Script like command language in the payload allows remote commands

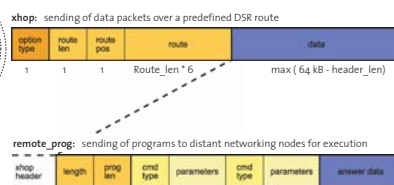
### Example Remote Topology Discovery

- xhop(A,B,C,D,E)
- inquiry
- return result
- xhop(E,D,C,B,A)

Result: (D,F,G)

### Performance

- 1-2 sec per hop, depending on inquiries
- ~10 sec to transmit, write and reboot 80kB firmware using selective flooding



### Communication with Bluetooth enabled Appliances: GSM Mobile Phones

BTNodes can communicate with other Bluetooth enabled devices using standard Bluetooth profiles for SMS (RFCOMM and AT commands) and object push (OBEX).

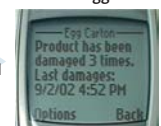
### Example Product Monitoring

In our demo example we can be alerted of sensor events such as shock or heat via SMS from a BTnode that acts as a smart tag. Vice versa an application can inquire a BTnode enabled 'smart object' for data and status.

### BTnode enabled Egg Carton



### SMS from Egg Carton



### Interactive Dialog

