A Minimal Bluetooth-Based Computing and Communication Platform

Jan Beutel
Swiss Federal Institute of Technology
Computer Engineering and Networks Lab
8092 Zurich, Switzerland
beutel@tik.ee.ethz.ch

Oliver Kasten
Swiss Federal Institute of Technology
Distributed Systems Group
8092 Zurich, Switzerland
kasten@inf.ethz.ch

Abstract

In this technical note we present an autonomous wireless communication and computing platform and its applications. The system is based on a Bluetooth communication module and a microcontroller. It is designed for a minimum use of resources while still being flexible. This platform is being used to set up large ad hoc networks, e.g. for collaborative remote sensing. In general, it can be used as a small but generic wireless networking node.

1. Introduction

Recently, networking multiple small devices in an uncoordinated and uncentralized fashion by wireless means has generated much interest. The Bluetooth technology [1] is an emerging communication standard that provides ad hoc configuration of master/slave piconets up to eight units. It allows data rates up to several hundred kilobytes per second. We have designed and implemented an autonomous programmable computing unit with Bluetooth communications. The system is being used to implement a collaborative sensor network [2] and serves as a testbed for ad hoc networking protocols in networks comprising large numbers of mobile autonoumous nodes.

A brief overview of the design considerations and the hardware implementation of such a network node is given in sections 2 and 3. Details of the power consumption of the system components are discussed in section 3. The final section 4 deals with the systems operating software, protocols, and applications.

2. Design Considerations

Commercial Bluetooth solutions are available as fully self-contained transceiver modules. They are designed to be used as add-on peripherals. They feature an embedded CPU, different types of memory, as well as baseband and radio circuits. The modules offer a generic Host Controller Interface (HCI) to the lower layers of the Bluetooth protocol stack while the higher layers of the protocol and applications must be implemented on the host system. Since the in-system CPU and memory are not available for user specific implementations, even a minimal standalone Bluetooth node needs an additional host CPU to execute applications and the corresponding higher layers of the Bluetooth protocol.

Having networks of hundreds of autonomous mobile nodes in mind, devices need to be carefully designed. Some of the features we considered in the design are:

- In-circuit programmable platform
- Component count
- Overall system size
- Sensor and user interface
- Single voltage with power management

3. System Overview

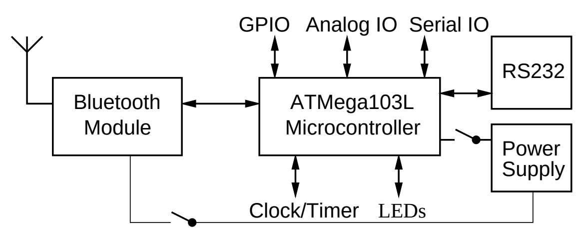

Figure 1. Schematic system overview

To run applications and the higher Bluetooth protocol layers, the Atmel ATMega103L SOC microcontroller with embedded memory was chosen. It features an 8-bit RISC core with up to 4 MIPS at 4 MHz, 128 Kbytes Flash memory and

This work has been partly funded by the European Commission as part of the Smart-Its project (contract No IST-2000-25428) and the Swiss Federal Office for Education and Science (BBW No. 00.0281).

4 Kbytes SRAM, a serial interface as well as several power modes. The Bluetooth module [3] is a fully shielded subsystem that is attached to a serial port of the microcontroller. External serial ports are used for data transfer and in-circuit programming. An external antenna is mounted with the required ground plane onto the 4x6 cm PCB substrate.

A voltage regulator is used to supply the necessary operating voltage from a small battery pack to the main components individually. This allows exact monitoring of power consumption and duty cycles.

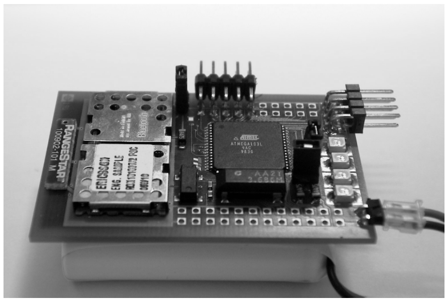

Figure 2. System mounted on battery pack

The system power consumption for different operating modes is given in table 1. It shows that the dominant component in such a wireless network node today is the Bluetooth module. The values presented clearly show that Bluetooth is not yet ready for deployment in real world scenarios. However, improved Bluetooth products [4] and advanced power management will eventually reduce power consumption considerably. Our system design allows for easy replacement of the Bluetooth tranceiver module, once improved modules are available on the market.

Table 1. System power consumption at 3.3 V

CPU Power Down, Bluetooth detached	9.9 mW
Running, Bluetooth detached	26.4 mW
Running, Bluetooth Transmit/Receive mode	108.9 mW
Running, Bluetooth Inquiry mode	148.8 mW

4. System Software and Applications

The system software provides low-level drivers, a scheduler and the host portion of the Bluetooth protocol stack. There are drivers for serial ports, analog to digital converter, general purpose IO, random number generator, system clock, and sensors. The scheduler provides event-driven scheduling of system and application tasks.

We ported the host portion of the Bluetooth protocol stack from an open source Linux implementation [5] to our microcontroller environment. The Linux version of the Bluetooth stack required multithreading capabilities and access to the serial port. On our system these functions are taken care of by the scheduler and the low-level drivers. The main obstacle in porting was the limited memory provisions of the microcontroller, since the original protocol stack was not optimized for memory consumption.

Devices can autonomously communicate using Bluetooth wireless technology. Supported layers are HCI and the Logical Link Control and Adaption Protocol (L2CAP). Inquiry, connection establishment to other devices, and disconnection procedures have been implemented and tested.

We are developing an application that deduces the topology of mobile Bluetooth devices based on the ability to inquire other Bluetooth devices within range. Whenever an unknown device enters inquiry range, its presence is detected. This information is then disseminated throughout the known part of the network. Approaches and first results are described in [6]. Furthermore, the system can also be used as a wireless interface peripheral.

Future work on the system software will concentrate on minimizing memory usage of the Bluetooth stack and adding the Service Discovery Protocol (SDP). In the application domain, the Smart-Its project [2] will make use of the system to form mobile ad hoc networks of collaborative sensors.

5. Conclusions

The implementation of a small, standalone communication platform using the Bluetooth protocol has been described. The feasibility of scaling the Bluetooth protocol stack to an embedded device with limited resources has been demonstrated. Even though the current implementation can hardly be deployed in real world scenarios due to its high power consumption, it serves well as a demonstraton platform for research in mobile and ad hoc connected networks (MANETs) and distributed sensor networks.

References

[1] Bluetooth Special Interest Group, *Specification of the Bluetooth System v1.1*, December 2000.

[2] The Smart Its Project, http://www.smart-its.org.

[3] Ericsson Microelectronics, *ROK 101 008 Bluetooth Module Datasheet*, 2001.

[4] Frank Op't Eynde et. al., "A Fully-Integrated Single-Chip SOC for Bluetooth," in *Proceedings of IEEE International Solid-State Circuits Conference - ISSCC 2001*, February 2001, pp. 196–197.

[5] Axis Communications, "Axis Bluetooth Driver Software," http://sourceforge.net/projects/openbt/.

[6] Thomas Moser and Lukas Karrer, "The EventCollector Concept," M.S. thesis, ETH Zürich, Distributed Systems Group, 2001.