Area-Efficient Digital Baseband Module for Bluetooth Wireless Communications

Myoung-Cheol Shin, Seong-Il Park, Sung-Won Lee, Se-Hyeon Kang, and In-Cheol Park
Division of Electrical Engineering, KAIST, Daejeon, Korea

Abstract

This paper describes a small and portable digital baseband module developed for Bluetooth wireless technology. To achieve portability and the small size, much of the Bluetooth baseband layer protocols are implemented in software running on the embedded microcontroller while the minimal tasks of low-level baseband processing, UART and USB interfaces, and audio CODEC are performed on the dedicated hardware blocks. The fully synthesizable baseband module was fabricated in a 0.25μm CMOS technology occupying 2.25×2.25mm^2 area

1. Introduction

Due to progress in related technologies in the past decades, the Bluetooth specification [1] was developed in 1999 to substitute cables connecting portable or desktop devices and to build low-cost wireless networks. It emphasizes low complexity, power consumption, and cost target [2], [3]. It is crucial to implement digital baseband processing in hardware and desirable to integrate the whole system on a chip to achieve the power and cost objective [3].

The tasks that the Bluetooth baseband module should perform vary significantly depending on the application. The baseband module should therefore be very flexible and programmable not to waste the hardware resources and processing power in any case.

Several baseband hardware cores have been developed either as a part of a system or as an IP (intellectual property) [4], [5]. Their size, however, are large because they perform almost all tasks in massive hardware while idling their microcontrollers [5], or because they use dual-port internal SRAMs [4].

This paper presents a simple, small, and portable Bluetooth baseband module that is suitable for use as an IP core. To reduce hardware area and gain more flexibility, the programmable embedded microcontroller performs as many tasks as possible and the hardware blocks implement only the most essential hardware functions. The module is made up of a logic part of only 85k gates and a 4kB single-port SRAM. Yet it performs all the essential Bluetooth baseband, link controller, link manager, and host controller interface (HCI) [1] functions including point-to-multipoint communications, multi-slot packets, encryption, scatternet, etc.

2. Overall Architecture

As depicted in Figure 1, the baseband module consists of five major functional units: the microcontroller subsystem, baseband unit, UART, USB, and audio CODEC.

This work was supported by the Korea Science and Engineering Foundation through the MICROS center at KAIST, Korea.

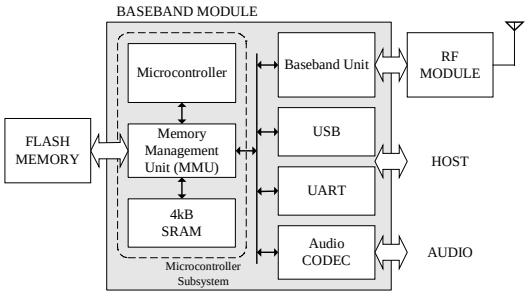

Figure 1. Block diagram of the Bluetooth baseband module

The primary clock input of the module is 48MHz. Whereas the USB runs at the 48MHz external clock to gain synchronization with 12MHz bitstream, the other units use a 12MHz clock that is generated by dividing the 48MHz clock to save power consumption.

3. Microcontroller Subsystem

The microcontroller controls the other units via memory-mapped I/O interface and interrupts. The other important task of the microcontroller is to run Bluetooth link manager, HCI, and a part of link controller protocol software. The microcontroller performs the complex part of the link control that require flexibility such as decision-making on received baseband packets and context switching between links, while the baseband unit performs bit-intensive, time-critical part. Although the link control tasks add some processing burden to the microcontroller, it still has excess processing power that may be used to run an entire protocol stack up to an application program for simple applications.

We have developed a clone of Advanced RISC Machines ARM7TDMI core [6] as the microcontroller of our Bluetooth baseband module. The clone was employed in order to take advantage of small die size and good code density of the ARM7TDMI, and to utilize its development environment. Only public-domain documents and information of the core are used to develop the clone.

The memory management unit (MMU) in Figure 1 manages memory interface and memory-mapped I/O interface of the microprocessor. One of the most important tasks of the MMU is direct memory access (DMA) of peripheral units. We moved the large distributed buffers into the internal SRAM and implemented a DMA, which results in a great area reduction. Compared to the distributed buffer architecture, the logic gate count is reduced by 35.7%. The MMU also provides flash memory programming capability. At power-up, a dedicated pin is used to select the loading of a new program from the UART interface.

We use a single-port 4kB on-chip SRAM, which is half as large as a dual-port SRAM of the same capacity. As the ARM7 architecture does not access the RAM while fetching instructions from the flash memory, the DMA can be easily implemented with a small single-port SRAM.

4. Baseband Unit

The baseband unit performs the bit-intensive baseband functions that are power-efficient if implemented in hardware, i.e. Bluetooth bitstream processing and encryption. In addition, the most time-critical portion of the link controller tasks, such as low-level timing control and frequency hop calculation, are processed by the baseband unit. The complex part of the link controller requiring flexibility runs on the microcontroller in order to make the hardware small and less complex. The baseband unit conforms to the latest version of the Bluetooth specification [1] and supports all of the six ACL, four SCO, and four common packet types.

The baseband unit is directly connected to an RF module via Ericsson's RF module interface. The control of the RF module is achieved through a serial control interface based on the IEEE standard 1149.1 boundary scan architecture. The interface can be easily retargeted to other RF front-end modules.

5. Host Controller Interface Units

For data transmission between the Bluetooth baseband module and a host such as a PC, PDA, or cellular phone, two serial interfaces, USB and UART, are provided. The USB and UART units constitute the physical part of Bluetooth HCI. To reduce chip area, they implement only the most basic hardware parts while the microcontroller performs the complex flow control.

The USB (Universal Serial Bus) unit complies with USB Specification 1.1 [7] and HCI USB transport layer specification of Bluetooth v1.1 [1], and supports full-speed 12Mbit/s interface. The different types of HCI packets are mapped onto different USB endpoints according to the Bluetooth specification [1]. The packets are stored in the corresponding endpoint buffer memory via DMA.

The UART (Universal Asynchronous Receiver Transmitter) unit is designed based on industry-standard 16C450. It supports from 300bit/s to 1.5Mbit/s, and the default bit rate is 57.6kbit/s. The UART unit includes a packet decoder that finds the HCI packet type and length of the received packets to help HCI processing of the microcontroller.

6. Audio CODEC

Bluetooth specifies three audio coding techniques: Log PCM coding using either A-law or μ-law and CVSD (continuous variable slope delta modulation) [1]. Since the table lookup of log PCM and low-pass filtering required in CVSD to avoid aliasing is appropriate for hardware implementation, we implemented all the three coding methods in a hardware audio CODEC. For simple audio applications, the audio unit interfaces directly to PCM audio devices without using HCI. The interface is designed to pass 8- to 16-bit decoded linear PCM signals.

7. Chip Implementation

We have implemented a prototype baseband module in a 0.25μm CMOS technology. A die microphotograph is shown in Figure 2, and the characteristics of the chip is summarized in Table I. The prototype chip contains no on-chip SRAM.

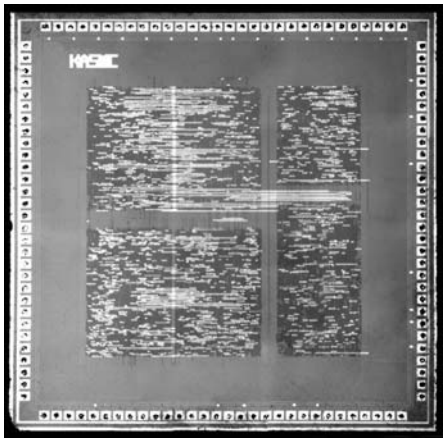

Figure 2. Die microphotograph of the prototype chip

Table I. Major chip characteristics

Technology	0.25μm CMOS 5LM
Core size	2.25×2.25mm^2
Gate count	85,000
Supply voltage	2.5V
Operating clock	48MHz

All the blocks of the chip are described in Verilog HDL and fully synthesizable. The synthesized gate-level design is autorouted to implement the chip. The prototype chip is fully tested using IMS ATS2 test station to verify its functionality and timing.

8. Conclusions

A small, flexible baseband module for Bluetooth wireless connection technology has been presented. We implemented a prototype chip in a 0.25μm CMOS technology. The chip occupies only 2.25×2.25mm^2 core area, and its functionality and timing characteristics are tested using a test station.

As the module is implemented in the way that the most of the tasks are implemented in software except the indispensable hardware-efficient functions, it can be adjusted to various Bluetooth applications without wasting hardware resources and processing power. Since the module is a small, fully synthesizable soft core, it can be easily integrated as an IP core on SOC ASICs for the applications related to Bluetooth communications.

References

[1] Specification of the Bluetooth System, Specification Volume 1, Core, Ver. 1.1, Feb. 2001.

[2] J.C. Haartsen, and S. Mattisson, "Bluetooth—A New Low-Power Radio Interface Providing Short-Range Connectivity," Proceedings of the IEEE, vol. 88, pp. 1651–1661, Oct. 2000.

[3] J. Bray and C.F. Sturman, Bluetooth: Connect Without Cables, Prentice Hall PTR, Upper Saddle River, NJ, 2001.

[4] F. O. Eynde, et al., "A Fully-Integrated Single-Chip SOC for Bluetooth," IEEE International Solid-State Circuits Conference, pp. 196–197, and 446, 2001.

[5] Ericsson Microelectronics, EBCP CherryRed Bluetooth Baseband IP, May 2001.

[6] Advanced RISC Machines, ARM7TDMI Data Sheet, Aug. 1995.

[7] Universal Serial Bus Specification 1.1, Sep. 1998.