

Digital Baseband SoC for Mobile WiMAX Terminal Equipment

● Miyoshi Saito ● Masahiro Yoshida ● Makoto Mori

(Manuscript received January 29, 2008)

We have developed a digital baseband system-on-a-chip (SoC) that conforms to the next-generation wireless communication standard, Mobile WiMAX, for mobile terminals. This SoC consists of the physical layer (PHY), lower media access control layer (LMAC), and dual processors. The PHY carries out multiple-input multiple-output (MIMO) processing and orthogonal frequency division multiple access (OFDMA) modulation and demodulation. The LMAC implements cryptography functions and controls frame-level data transactions. The dual processors are used for MAC layer processing to achieve high data throughput with a low clock frequency. The PHY and LMAC offer a maximum data reception of 45 Mb/s at an operating frequency of 44.8 MHz, and for the SoC including the firmware, we obtained a prediction of achieving 45 Mb/s. Regarding power consumption, the SoC dissipates 252 mW during 15-Mb/s downlink data reception.

1. Introduction

Mobile WiMAX (IEEE 802.16e)^{1),2)} is expected to become the next-generation communication standard and may well provide high-speed data communication services at a low price. Since it uses orthogonal frequency division multiple access (OFDMA), it allows multiple users to share an OFDMA frame and offers highly efficient frequency use. It uses highly advanced communications technology, such as multiple-input multiple-output (MIMO) technology and beamforming so that large amounts of information can be sent.

In this paper, we introduce our newly developed system-on-a-chip (SoC) that conforms to the Mobile WiMAX Profile³⁾ and offers a single-chip solution for physical and media access control layer processing to mobile stations (MSs). We also explain how we have achieved 45-Mb/s data reception, which is the maximum data rate defined in the profile. Section 2 shows the architecture of the SoC from the viewpoint of processor

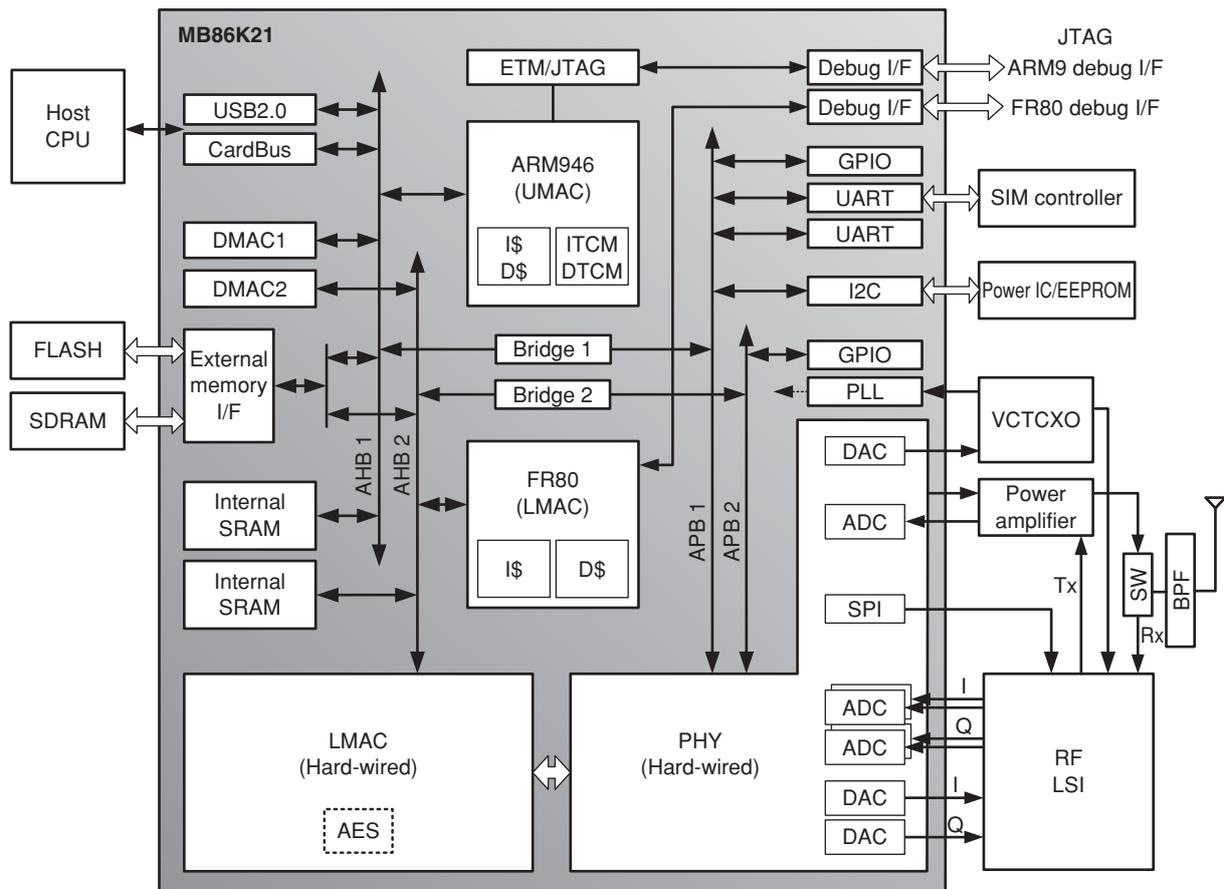
and bus performances, section 3 explains the physical layer (PHY) that achieves processing with small memories, and section 4 describes the processing of the lower media access control layer (LMAC) with low latency. Finally, we show measured results for the SoC from the viewpoint of processor and radio performances.

2. SoC architecture

2.1 Overview of SoC

A block diagram of the SoC is shown in **Figure 1**. The SoC consists of (1) an ARM946E-S^{4) note 1)} processor that implements upper-layer media access control (UMAC) layer processing and a host interface process, (2) LMAC hardware and an FR80⁵⁾ processor that handle LMAC layer processing, and (3) PHY hardware that carries out PHY processing.

note 1) ARM and ARM946E-S are the trademarks of ARM Limited.



- ADC: Analog-to-digital converter
- AES: Advanced encryption standard (algorithm)
- BPF: Band-pass filter
- DAC: Digital-to-analog converter
- ETM/JTAG: Embedded trace macrocell, Joint Action Group
- I/F: Interface
- PLL: Phase locked loop
- Rx: Receiver
- SPI: Serial peripheral interface
- SW: Switch
- Tx: Transmitter
- VCTCXO: Voltage controlled temperature compensated crystal oscillator

Figure 1
Block diagram of SoC.

2.2 Interfaces

The SoC has CardBus and USB2.0 interfaces as host interfaces. It also has a synchronous dynamic random access memory (SDRAM) interface, a flash memory interface, and an I2C interface (EEPROM: electrically erasable programmable read only memory) as external memory interfaces. In addition to

these, it supports various interfaces such as general-purpose input/output (GPIO), universal asynchronous receiver/transmitter (UART), and debugging interfaces.

2.3 Dual processor system for MAC

In order to manage a complex media access control layer (MAC) process and achieve

high-speed data transfer, the system must have high processing ability. However, this SoC has a strict limitation on power consumption because it is a baseband chip for mobile terminals. Therefore, we chose to use a dual processor system. This lets us reduce the clock frequency, which allows us to use low-leakage process technology for our implementation. As a result, we achieved a low static power consumption as well as a low dynamic power consumption. In addition, we divided the MAC process into LMAC, which has a critical response time, and UMAC, whose timing restriction is not so severe. One of the processors handles the LMAC process and the other one handles the UMAC process. A performance estimation of the dual processor system is described below.

2.4 Estimate of performance for MAC process

This section shows how we determined the hardware resources including the bus structure and memory size for those dual processors. From the necessary computational complexity and bus activity for 45-Mb/s downlink data reception, which is our target performance, we optimized as described below. Here, we just mention, as an example, the performance estimation of the ARM processor for the UMAC process. The same methodology is applied to the FR80 processor for the LMAC process.

2.4.1 Items to consider for processor performance

- Fundamental processor load

We evaluated the processing capacity necessary for the UMAC process theoretically as follows. The UMAC process comprises a protocol process and a data transfer process. First, we estimated the number of assembler steps for the protocol and the data transfer process, respectively. Then, we multiplied these numbers by the cycles-per instruction (CPI) to get the number of processor cycles. From that, we estimated the

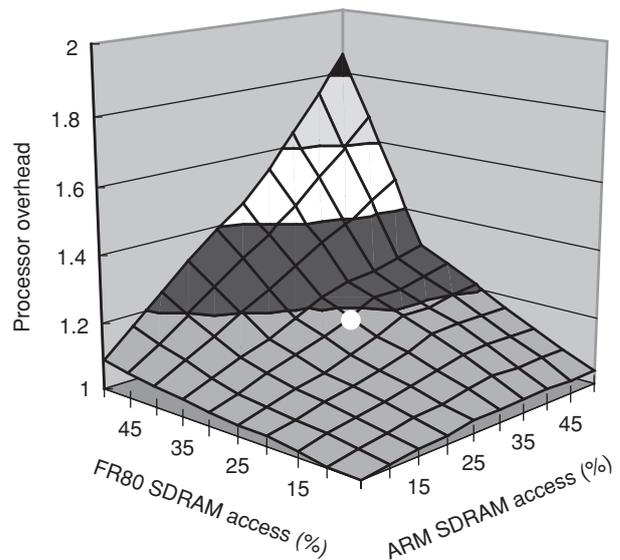


Figure 2
Overhead caused by SDRAM access competition.

fundamental processor loading rate for the target transmission speed (downlink of 45 Mb/s).

- SDRAM access conflict

The processors share external SDRAM, so a bottleneck could occur here. Therefore, we evaluated the performance by virtual prototyping using an electronic system-level tool. We calculated the SDRAM access frequency of the processors by analyzing the behavior of the UMAC and LMAC processes and simultaneously simulated them in the virtual prototyping environment.

The processing overhead of the ARM processor versus the SDRAM activity for both ARM and FR80 processors is shown in **Figure 2**. This graph shows that the estimated overhead is about 20% when the SDRAM activity caused by SDRAM access from each processor is 30%, which is the maximum activity for a single processor. This overhead is acceptable considering the processor work load.

- Built-in memory and cache

In addition to the average processor loading rate, the instantaneous performance, such as interrupt response, is important. Since access

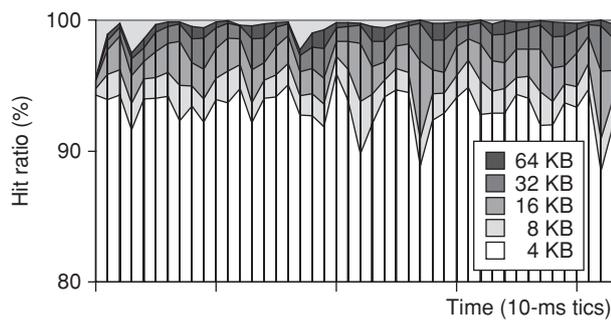


Figure 3
Instruction cache hit rate.

to external SDRAM has an overhead caused by conflict between the two processors, as described above, it is difficult to ensure instantaneous performance. For this reason, in order to ensure instantaneous performance, each processor has a local memory on its own local bus, which does not suffer from interference from the other processor.

The instruction cache size was evaluated using a virtual prototyping environment. We measured the fluctuation in hit rate when the cache capacity was changed (**Figure 3**). In particular, we focused on a temporary decline in hit rate and chose an adequate cache size to keep above 95%. On the basis of the experimental results, we defined the instruction cache to be 32 KB and data cache to be 16 KB.

2.4.2 Estimation of processor performance

The practical working load of the ARM processor with all of the above conditions is shown in **Figure 4**. We took into account cache misses, memory latency, and SDRAM access conflict with the fundamental processor load to get the practical working load. We obtained a processor margin of 48% for 45-Mb/s downlink.

3. Physical layer (PHY)

3.1 Overview of PHY

The PHY establishes a wireless physical layer connection with base stations (BSs). For downlink, it receives and decodes OFDMA signals. For uplink, it modulates subcarrier data

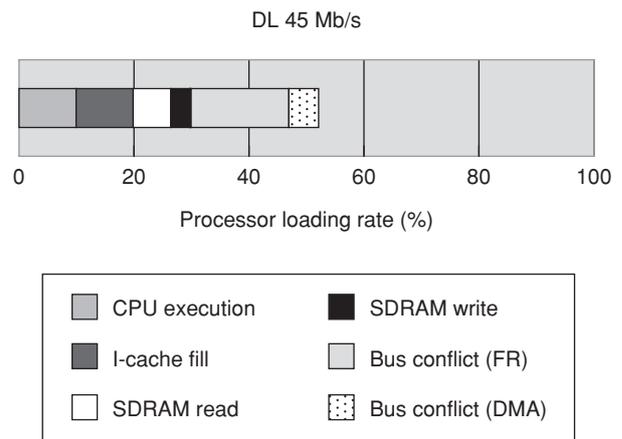


Figure 4
Estimated processor load.

and generates the transmit OFDMA signal.

3.2 Frame structure of Mobile WiMAX

The OFDMA frame structure used for Mobile WiMAX is shown in **Figure 5**. The horizontal axis shows the OFDMA symbol number that corresponds to time, and the vertical axis shows frequency. The frame consists of several tens of OFDMA symbols and is divided between a downlink (DL) subframe and an uplink (UL) subframe. Logically, the frame starts from a preamble symbol followed by a frame control header (FCH) burst and a downlink map (DL-MAP) burst. The FCH burst includes a modulation and coding scheme (MCS) for decoding the DL-MAP burst. The next region of the DL-MAP in the DL subframe is divided into several data bursts that carry user messages and management messages. The DL-MAP describes the allocation and MCSs of the data bursts. Because the specification^{1,2)} or the profile³⁾ has no limitation on the allocation in a frame, an MS should support any data burst allocation for connection to any BS. This requires that an MS that has the flexibility to decode any kind of frame structure.

3.3 PHY downlink processing

Normally, to obtain flexibility, each data

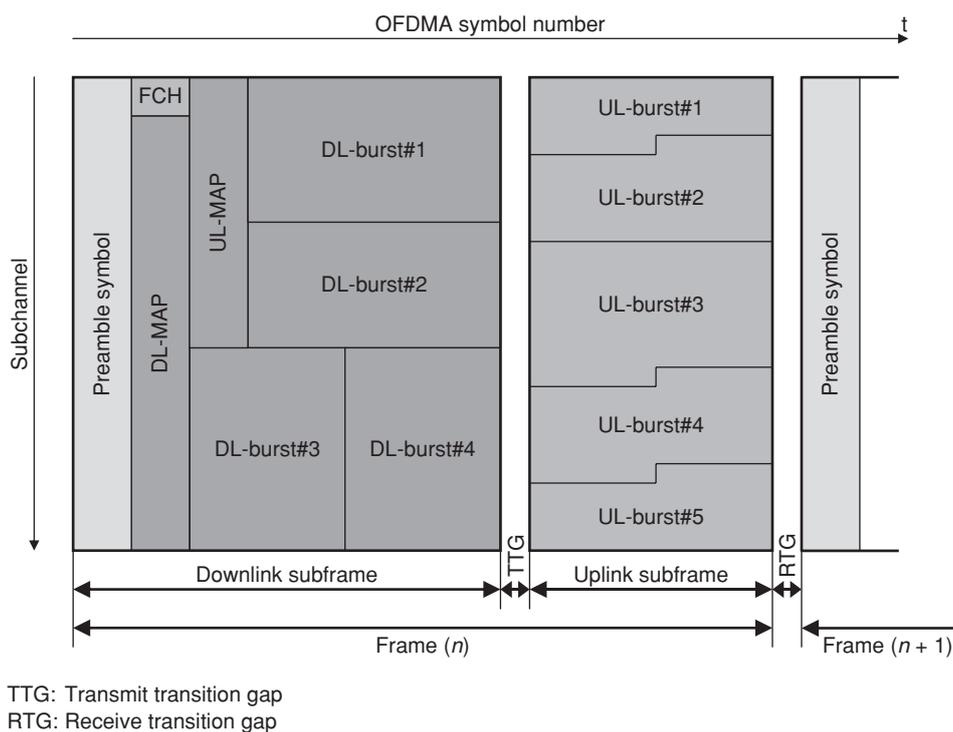


Figure 5
Frame structure.

burst is extracted from received downlink data and decoded, after all of the downlink data has been stored in a memory. However, this method has two issues for 45-Mb/s downlink data reception:

- 1) A large frame memory is needed to store all of the downlink data that corresponds to 45 Mb/s.
- 2) Processing latency is high because the decoding of data bursts cannot be started until all of the data has been stored in memory.

To resolve these issues, we chose not a data burst but an OFDMA symbol as the unit of decoding. This can reduce both memory size and latency to one third. We utilized the features that each data burst is composed of forward error correction (FEC) blocks and that each FEC block spans several OFDMA symbols. If one OFDMA symbol is decoded, the complete FEC block cannot be obtained because the FEC block is divided by

the OFDMA symbol boundary. Therefore, we chose to use an “FEC-block buffer” instead of the frame memory, in order to reconstruct the FEC block from the divided FEC block. Although our approach additionally requires a “symbol buffer”, explained later, it can reduce the total required memory size to one third. In addition, because FEC processing can start as soon as a few OFDMA symbols have been decoded, processing latency is also reduced to one third.

Our approach also requires a burst reconstruction process because the FEC blocks in different data bursts are processed in parallel. This process is explained in section 4.5.

Here, we show the details of our approach. A block diagram of the PHY downlink processing block is shown in **Figure 6**. In our architecture, the required memories are the symbol buffer and the FEC-block buffer. The symbol buffer stores several OFDMA symbols that are result data from a fast Fourier transform (FFT). If the

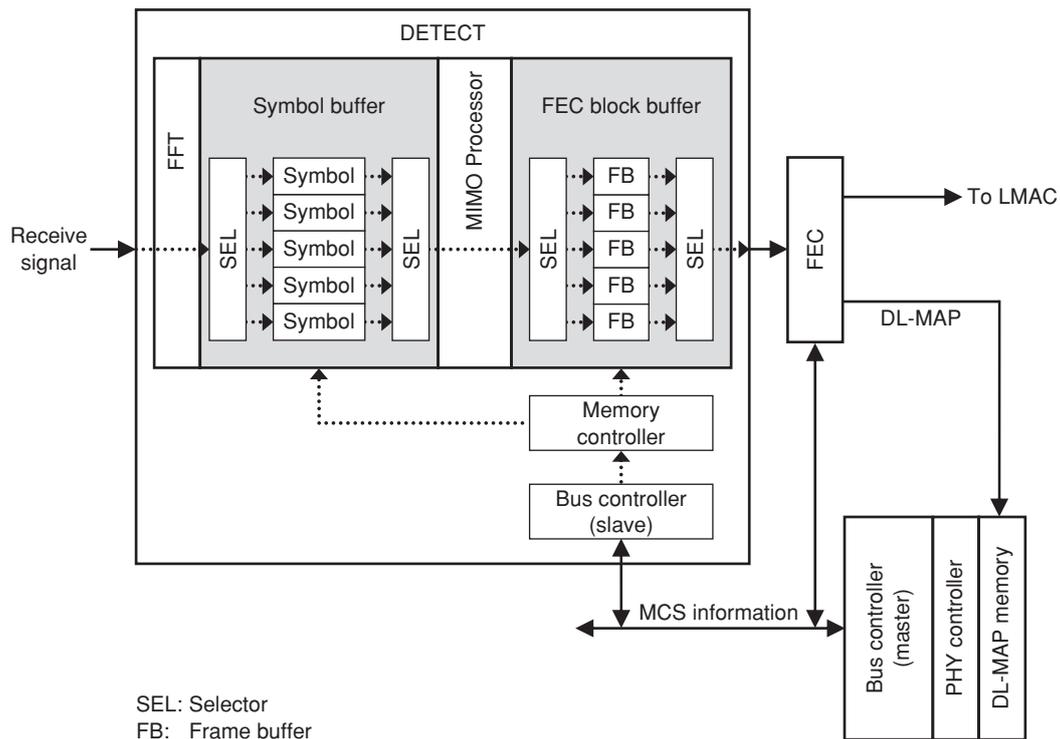


Figure 6
PHY downlink processing block.

FEC-block buffer has enough capacity to write FEC-block data, a memory controller reads an OFDMA symbol from the symbol buffer and feed it into a MIMO processor. Output data from the MIMO processor is divided into FEC-blocks based on the MCS information about each burst, and queued into the FEC-block buffer. After data division to the FEC block, an FEC processing unit performs error correction for every FEC block and sends corrected FEC blocks to the LMAC.

3.4 Details of PHY block

A block diagram of the PHY block is shown in **Figure 7**. This section overviews the internal components of the PHY block.

PHY controller: A PHY controller analyzes the DL-MAP and controls hardware in the PHY block.

SYNC: A SYNC (synchronization) unit detects the frame start timing and determines the FFT window position. It also detects the

frequency offset for a BS and corrects it by automatic frequency control. In addition, it controls a radio frequency module, i.e., automatic gain control, etc.

DETECT: A DETECT unit performs channel estimation based on pilot subcarriers, correction for amplitude and phase, and subcarrier demodulation. It also performs space-time code and MIMO processing.

FEC: An FEC unit corrects errors in demodulated subcarrier data based on error-correcting code. The unit has a Viterbi decoder and convolutional turbo code decoder.

UL-PHY: A UL-PHY unit generates OFDMA signals to transmit. This unit receives transmit data subcarriers from the LMAC and performs modulation. It also performs an inverse FFT and peak-to-average-power ratio operation.

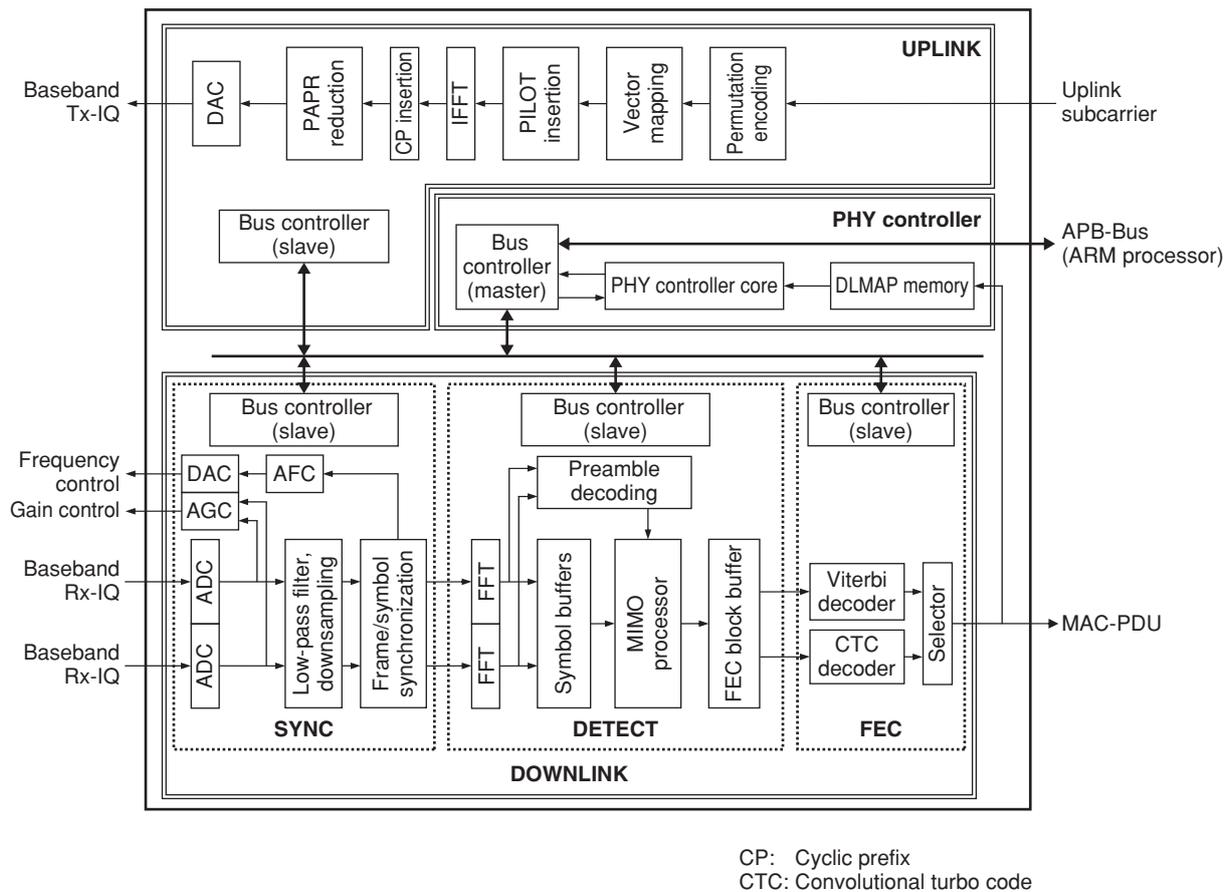


Figure 7
PHY baseband processor.

4. Lower media access control layer (LMAC)

4.1 Overview of LMAC

The LMAC is located between the UMAC and PHY. It converts PHY data into the format used for UMAC and vice versa and also performs data encryption and decryption. For the UL data processing, it processes FEC encoding and manages the UL hybrid automatic repeat request (HARQ) process and constructs the UL subframe structure. The LMAC also performs a MAC-level automatic repeat request (ARQ) process.

4.2 Interfaces of LMAC

The processing flow of sending and receiving data in LMAC is shown in **Figure 8**. Also

shown are the interfaces between layers. The left hand side of the figure shows the packet processing flow and the right hand side shows the layers. A MAC service data unit (MAC-SDU) packet is the interface between UMAC and LMAC. Typically, it is an IP packet. The interfaces between LMAC and PHY are the subcarrier data of the OFDMA symbol and decoded FEC blocks for UL and DL, respectively. The interfaces between LMAC hardware and LMAC firmware are the MAC-PDUs before encryption and after decryption for UL and DL, respectively. This hardware/firmware partitioning within LMAC is decided from the standpoint of required throughput. Because encryption and decryption processes applied to MAC-PDUs using an

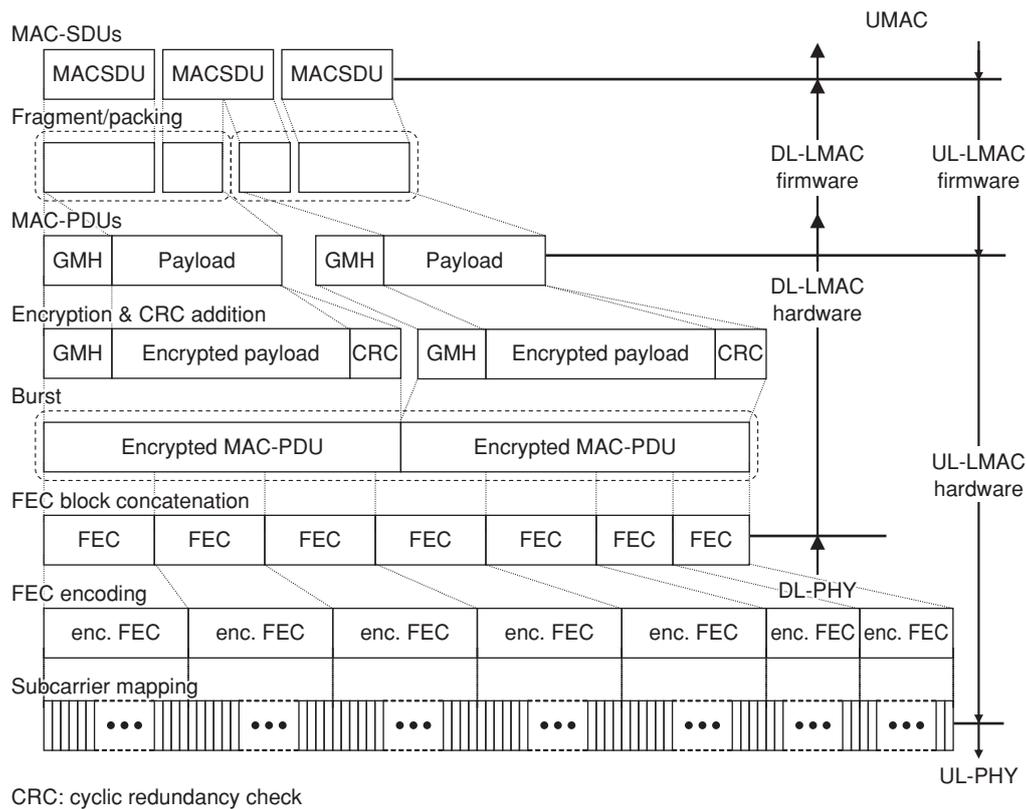


Figure 8 LMAC data handling and inter-layer interfaces.

AES-CCM^{note 2)} algorithm⁶⁾ require a lot of computational power, they must be done in hardware to meet the required throughput, which reaches about 45 Mb/s. From the viewpoint of frame processing, the role of the LMAC hardware is data processing for each frame, and that of LMAC firmware is hardware control and management of multiple-frame sequences such as the ARQ and HARQ retransmission mechanisms.

4.3 Flow of LMAC data processing

In the UL processing, the LMAC firmware carries out fragment/packing for MAC-SDUs received from UMAC and re-constructs the MAC-PDUs. In this re-constructing, a header

called the Generic MAC Header (GMH) is added to each MAC-PDU. The LMAC hardware uses the AES-CCM algorithm to encrypt the payload of the MAC-PDU. At this stage, the burst is composed of encrypted MAC-PDUs. Next, it is divided into FEC blocks, and convolutional turbo coding is carried out for each FEC block. The encoded burst is allocated to a UL burst in a UL subframe, according to the UL-MAP. The DL processing is almost the inverse processing of that for the UL, but the interface with the PHY is the decoded FEC blocks.

4.4 LMAC hardware architecture

There are two large performance demands for LMAC hardware.

- 1) High throughput: 45-Mb/s data processing must be carried out with minimum delay.

note 2) AES: advanced encryption standard; CCM: counter with cipher block chaining message authentication code.

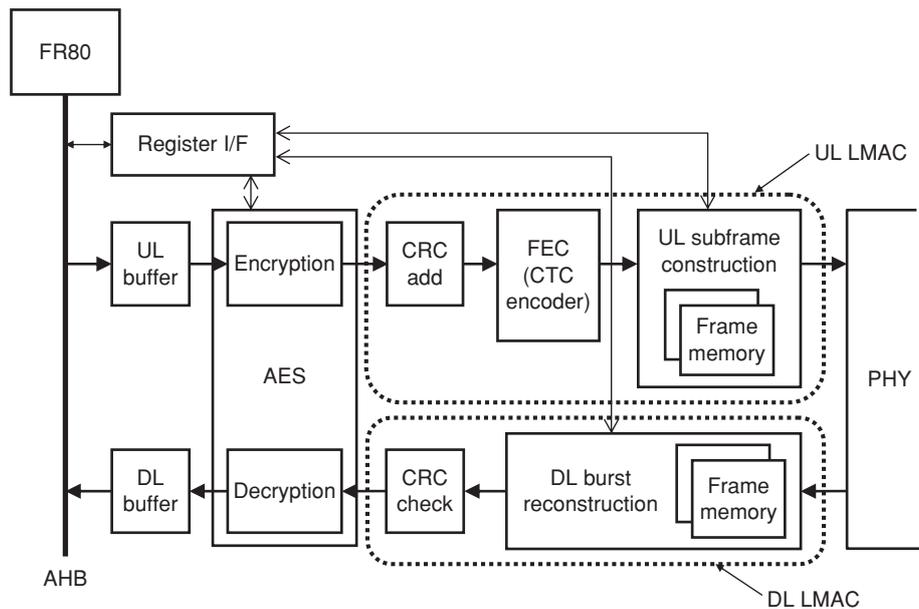


Figure 9
LMAC block diagram.

- 2) Fast feedback operation: It is necessary to transmit the control information by the next UL subframe on the basis of data received in the current DL subframe. Such information includes the acknowledge or negative acknowledge (ACK/NACK) information of DL HARQ or the measured carrier-to-interference-plus-noise ratio. It requires DL and UL processing with low latency.

To meet these requirements, three techniques are used. A block diagram of the LMAC hardware is shown in **Figure 9**. The hardware is composed of the DL processing block, UL processing block, and cryptographic processing block. LMAC firmware is run on the FR80 processor.

The first technique is the separation of the DL and UL processes. Because time division duplexing is used in Mobile WiMAX, the DL and UL processing overlap in the time domain in the LMAC layer. Separating the DL and UL can eliminate this process interference.

The second technique is that the crypto-

graphic block, i.e., the AES block, operates at double the clock frequency of other blocks in order to improve throughput. Since the AES encryption/decryption algorithm in the cryptographic block is not carried out in a fully pipelined manner, and 10 or more cycles are required for each 128-bit input data, this process is a bottleneck in both throughput and latency in LMAC processes.

The third technique is separation of the control and data paths between the LMAC hardware and the FR80 processor. The control path is the register interface that enables the delivery and receipt of control information between the LMAC hardware and the FR80. The throughput of this path is limited because it allows single read/write access within the LMAC operating frequency of 44.8 MHz, but it offers easy control of the LMAC hardware. The data path is composed of the UL buffer and the DL buffer, which exchange transmit data (UL) and received data (DL). These buffers are shared by the FR80 and the LMAC hardware as a result of using dual port memories so that burst access

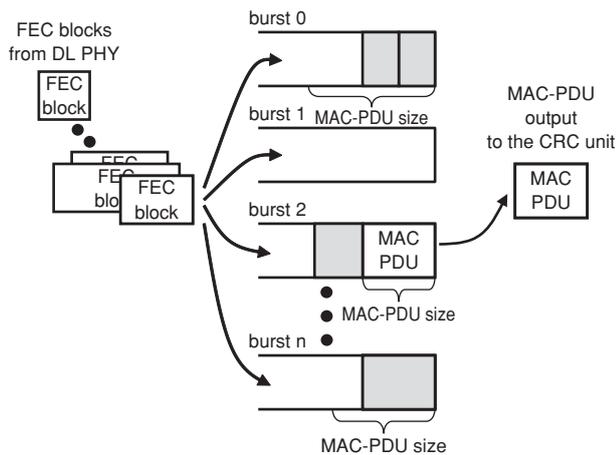


Figure 10 DL burst reconstruction and MAC-PDU extraction process.

at 112 MHz (the operating frequency of the local bus [AHBTM note 3)]) is available. This mechanism achieves high throughput for both sending and received data.

4.5 DL burst reconstruction

In relation to data burst processing in the PHY, as previously mentioned, the LMAC reconstructs data bursts from FEC blocks and extracts MAC-PDUs from the data bursts. An overview of the burst reconstruction and MAC-PDU extraction process carried out in the DL burst reconstruction unit is shown in **Figure 10**. The steps are as follows.

- 1) Sort the received FEC blocks from the PHY to each burst.
- 2) Extract the MAC-PDU in each burst using the MAC-PDU size information described in the GMH.
- 3) Output the extracted MAC-PDUs to the next unit, which is a cyclic redundancy check unit.

5. Measurement

Values measured using an actual SoC are presented in this section.

note 3) AHB: advanced high-performance bus, AHB is the trademark of ARM Limited.

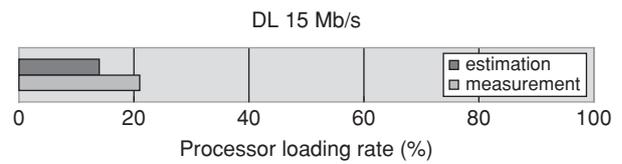


Figure 11 Evaluation of processor load.

5.1 Processor performance

Here, we describe a performance evaluation for an actual SoC. An ETM9TM note 4) was implemented on the ARM, making it possible to monitor software operation on an actual machine. The estimated and actually measured processor loads for a 15-Mb/s downlink are shown in **Figure 11**. We confirmed that this estimation method is appropriate and obtained a prediction that a 45-Mb/s downlink is possible.

5.2 BER and constellation

The environment used for bit error rate (BER) measurement for the PHY performance and the structure of a measured frame are shown in **Figure 12**. The first three symbols contain the preamble, FCH, and DL-MAP. The rest of the DL subframe is a MIMO zone that includes data bursts. The possible data rate with this frame structure is about 45 Mb/s. Measured results for BER and constellation are shown in **Figure 13**. The measurement results confirmed 45-Mb/s throughput and error-free data reception.

6. SoC features

The specifications of the SoC implementing the architecture described above and mounting the PHY/LMAC hardware are listed in **Table 1**.

7. Conclusion

We developed an SoC for terminals that will be the key devices of the next-generation communication standard — Mobile WiMAX.

note 4) ETM9 is the trademark of ARM Limited.

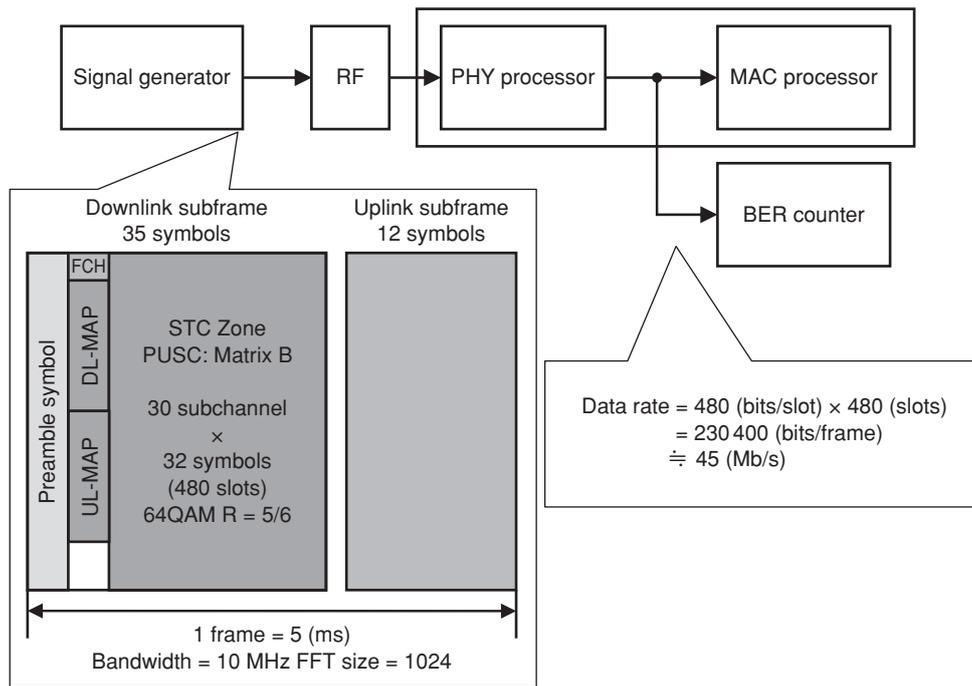
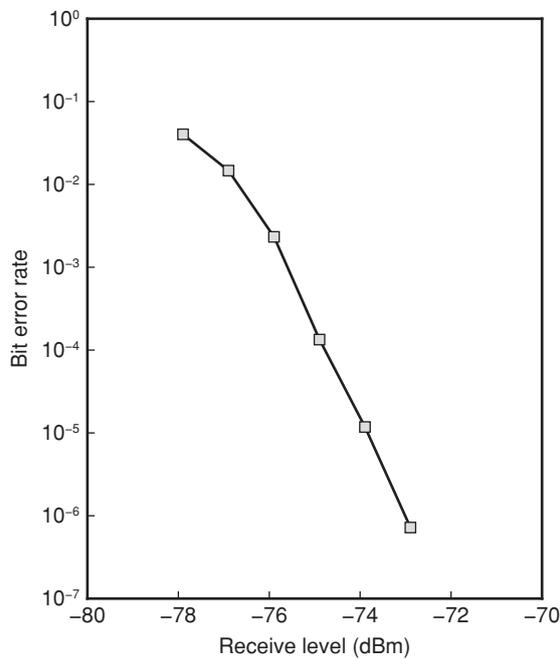
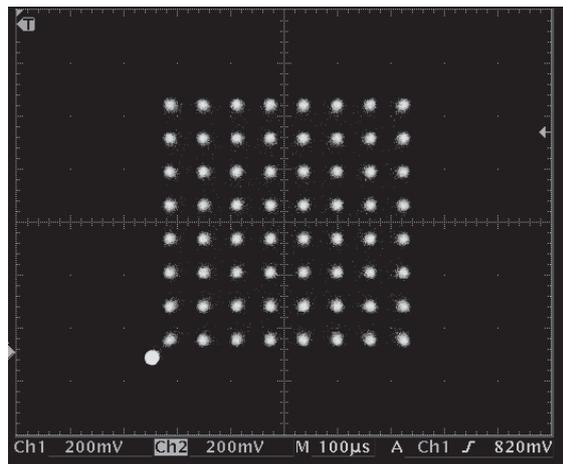


Figure 12
Measurement model.



(a) Bit error rate



(b) Constellation

Figure 13
Measurement result.

Table 1
SoC features.

| | |
|--------------------------|-------------------------------------|
| Process | 90 nm |
| Operating frequency | PHY/LMAC: 44.8 MHz CPUs: 112 MHz |
| Power consumption (core) | 252 mW@15 Mb/s (DL) |
| Power supply | 1.2 V core, 1.8/2.9/3.3 V I/O |
| Package | 610-FBGA 16 mm |

FBGA: Fine-pitch ball grid array

The PHY/LMAC offers a maximum processing capability of 45 Mb/s at 44.8-MHz operation. For the SoC including its firmware, we got a prediction of achieving 45 Mb/s. Future research will

focus on reducing the power consumption even more: the SoC currently dissipates 252 mW during 15-Mb/s downlink data reception.

References

- 1) IEEE Std 802.16e™-2005 and IEEE Std 802.16™-2004/Cor1-2005.
- 2) P802.16-2004/Cor2/D3 (Draft Corrigendum to IEEE Std 802.16-2004).
- 3) WiMAX Forum™ Mobile System Profile, Release 1.0 Approved Specification (Revision 1.4.0).
- 4) ARM Limited: ARM946E-S technical reference manual.
- 5) Fujitsu Limited: FR family instruction set manual.
- 6) NIST Special Publication 800-38C, May 2004.



Miyoshi Saito

Fujitsu Laboratories Ltd.

Mr. Saito received the B.S. and M.S. degrees in Physics from Tokyo Institute of Technology, Tokyo, Japan, in 1987 and 1989, respectively. He joined Fujitsu Laboratories Ltd., Kawasaki, Japan in 1989. After working on research in quantum electronics, he was engaged in research on high-speed DRAMs and high-speed interfaces. Since 1998,

he has been engaged in research and development of embedded processors, reconfigurable logic, software-defined radio, and wireless communication devices. He is a member of the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM).



Makoto Mori

Fujitsu Laboratories Ltd.

Mr. Mori received the B.S. and M.S. degrees in Electronics Engineering from Chiba University, Chiba, Japan in 1995 and 1997, respectively. In 1997, he joined Fujitsu Ltd., Kawasaki, Japan, where he worked on the development of the physical architecture for a high-end processor. He moved to Fujitsu Laboratories Ltd., Kawasaki,

Japan in 2007, where he has been engaged in the development of an SoC for Mobile WiMAX, especially performance analysis and system architecture.



Masahiro Yoshida

Fujitsu Laboratories Ltd.

Mr. Yoshida graduated from Fujitsu College of Technology, Kawasaki, Japan in 1990. He joined Fujitsu Ltd., in 1985, and moved to Fujitsu Laboratories Ltd., Kawasaki, Japan, in 1993, where he worked on the development of image processors. Since 2000, he has been engaged in the development of wireless communication

devices, such as SoCs for ISDB-T and Mobile WiMAX.