An Equalizing and Channel Coding Processor for GSM Terminals

Minoru Okamoto*, Toshihiro Ishikawa**, Shinichi Marui*, Masayuki Yamasaki*,
Katsuhiko Ueda*, Nobuo Asano**, Mitsuru Uesugi**,
Yoshiko Saitoh**, Yukihiro Fujimoto**, Susumu Furushima**

*Semiconductor Research Center, Matsushita Electric Ind. Co., Ltd.
3-1-1, Yagumo-Nakamachi, Moriguchi, Osaka, 570, Japan
**Engineering division, Matsushita Communication Ind. Co., Ltd.
600, Saedo-cho, Tuzuki-ku, Yokohama, 224, Japan

Abstract

A new DSP architecture for equalizing, channel coding/decoding and encryption/decryption required by GSM hand portable terminals is presented. In the DSP, which is called EQCHAN (Equalizer and Channel coding/decoding processor), these tasks are managed in common units, that is, the data processing unit (DPU) and the bit manipulation unit (BMU). The LSI that contains EQCHAN was designed using 0.8 μm CMOS technology and its die size is 123mm^2. The power consumed in the LSI is 60mW at 3.6V under continuous communication mode and this value is sufficient for a portable terminal. In this paper, we describe the detail architecture of EQCHAN.

1. Introduction

As post analog mobile phone systems, digital mobile phone systems such as the GSM system in Europe and the PDC system in Japan are becoming popular. In these systems, small size, long operating autonomy and low cost for hand portable terminals are important. The key technology to resolve these problems is system integration on one LSI chip. Fortunately many functions of digital mobile phone set are processed in digital and it is relatively easy to integrate many functions in a digital chip than in an analog chip. Until now, many LSIs for GSM terminals were developed [1-2]. It is unpractical, however, to prepare logic circuits for each logical function, because if one designed LSIs in this way the die size will become huge and the cost will be extremely high.

We developed a new DSP architecture, called EQCHAN, that realizes equalizing, channel (de)coding, and encryption/decryption, and we designed a LSI which contains EQCHAN, ADC, DAC, and controller that generates control signals such as TDMA window control signals, AGC control signal, AFC control signal, RF power ramping control signal on one chip (see Figure 1). This paper describes the architecture of EQCHAN.

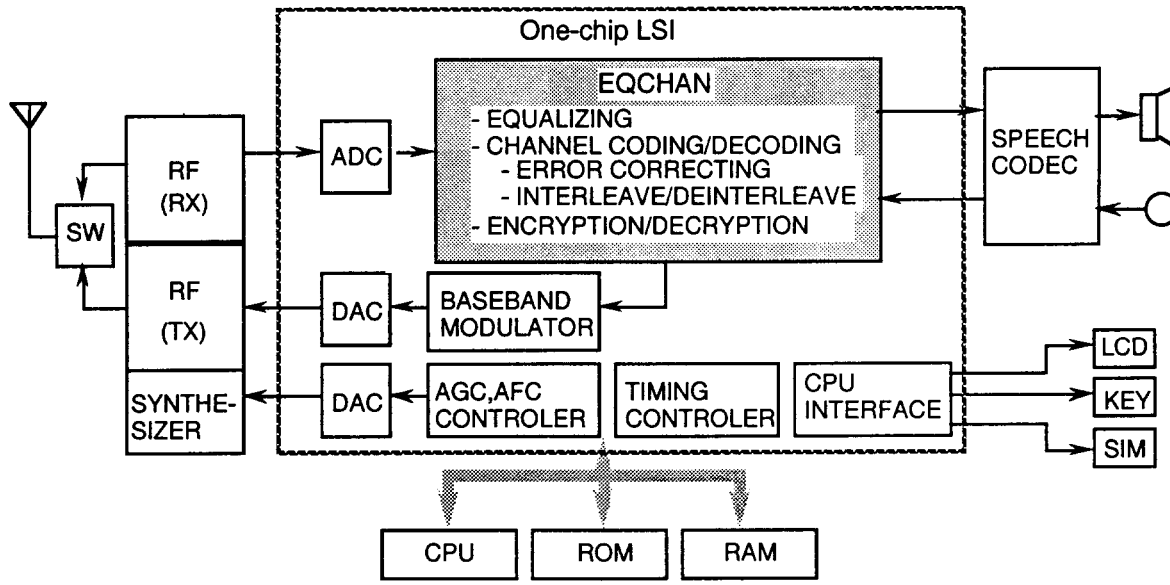

Fig.1 GSM Mobile System

0-7803-2431-5/95 $4.00

2. Why We Chose A DSP Architecture?

2.1 The Timing of The Tasks

We investigated the timing to realize the tasks required by a GSM portable terminal [3], such as equalizing and channel (de)coding. As a result, we got the idea that almost all the baseband processes can be performed sequentially except the process of input data from ADC.

We shall discuss it in detail. In the process for receiving data, first the data from ADC are separated to the imaginary part and the real part (I and Q element). Then the two pairs of the data are stored into memory (here, we defined the above process "process A"). Next, the data held in memory are equalized and then decrypted and channel decoded according to the type of information stream, such as TCH/FS, SACCH, etc. Tasks required for transmitting data, that is encryption and channel coding, are performed at the same time as the process A. To assign the timing like the above, there is enough time to complete these tasks in one TDMA frame, that equals 4.615 ms.

Therefore, if a proper unit is prepared for the process A, it is possible that the rest processes such as equalizing, channel (de)coding, encryption and decryption are managed in a common unit.

2.2 DSP vs. Logic Circuits

Before moving on to the discussion about our architecture, it is helpful to describe the general specification of DSP architecture and the preparation logic circuits for each logical function (here, we call it "logic circuits").

In the logic circuits, generally the die size becomes large. There are several reasons for this. One is that each logic circuit is designed for a few functions, so even if there are similar processes in some functions such as shifting bits and addition of the data, each logic circuit cannot help containing the same structure. Another reason is that many and various functions must be performed in a portable terminal, so the amount of circuits is large. In that case, however, it is easy to reduce the power consumption. It is known that the power consumption of CMOS devices is proportional to the operating frequency F, the load capacity C and the square of the supply voltage V. Each logic circuit performs a few functions, so the operating frequency F needed in the circuit can be small and it leads to reduce the supply voltage V.

In the DSP architecture, the die size is apt to be small, because a DSP can perform many functions in common processing units. However these units run at high frequency F and consume relatively large power consumption.

2.3 Our Choice

In order to reduce the power consumption in a DSP, there are some effective means at the architecture design level [4]. We attached great importance to realizing these tasks by one, small LSI. Therefore, we chose the way of DSP architecture, and we conquered the problem of power consumption using the following solution.

In order to reduce operating frequency F, we developed the high performance processing unit that can perform the necessary process in small dynamic steps. The details about this are described in the next chapter. We also designed the efficient instruction sets that control many processing circuits in one machine cycle. Figure 2 shows the example of the instruction sets.

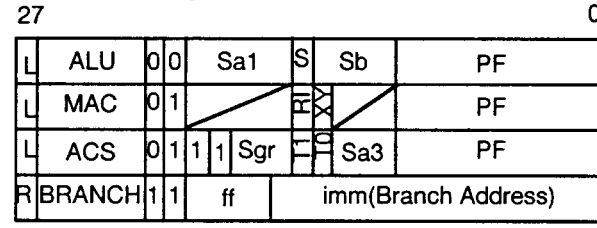

Fig.2 Examples of Instruction Format

Furthermore, we prepare some clock control facilities, for example, stopping the supply of clock to the unit that is not necessary to be operated. Also we designed the architecture that can transfer the data between I/O unit and memories at any time, even if the operating clock input from external pin is stopped.

By means described above, we designed the new DSP architecture that is small size and low power consumption.

3. EQCHAN Architecture

3.1 Overview

Figure 3 shows the block diagram of the EQCHAN. The EQCHAN is constructed with five main blocks; the data processing unit (DPU), the bit manipulation unit (BMU), the memory unit (MEU), the program control unit (PCU), and the I/O interface (IOU). DPU and BMU are in the next chapter.

The MEU consists of two plane memory X and Y, and pointers for these two memory planes. For equalizing process, each memory can store a set of real and imaginary part of data, so it is possible to read four data onto four data buses in one machine cycle.

The PCU controls the whole unit in EQCHAN. Its function is to decode instructions output from the instruction memory, to control program sequence, and to generate control signals to DPU, BMU and so on. There are 33 types of instruction. All instructions consist of 28 bits and performed in one machine cycle (see Figure 2).

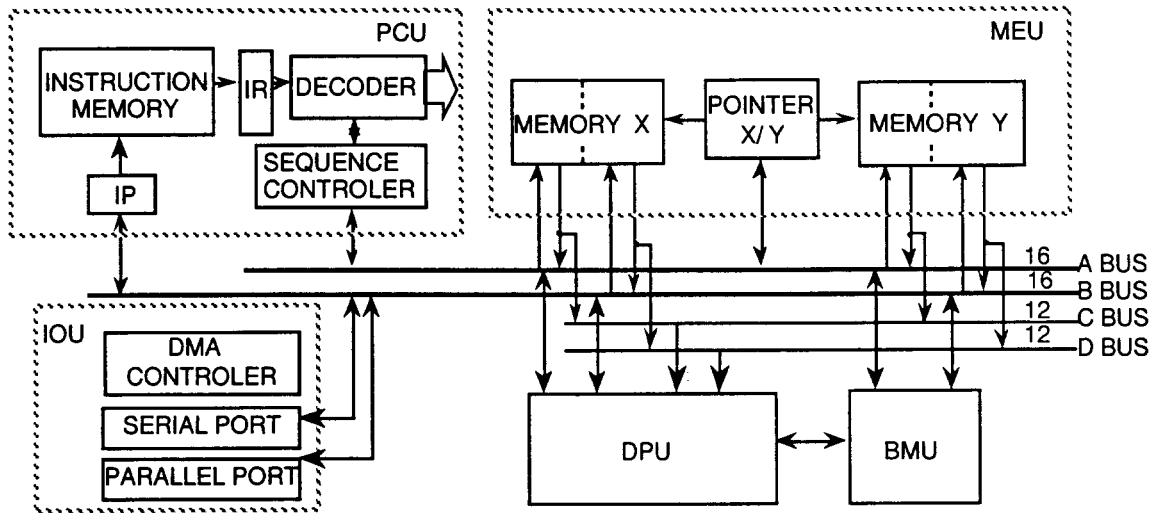

Fig.3 Block diagram of EQCHAN

The IOU has one half-duplex serial port and 16-bit parallel port. The DMA controller transfers data between the serial port and the memory X in one machine cycle.

3.2 DPU

The DPU has two 12x12-bit multipliers (MULA and MULB), a 16-bit ALU, a 16-bit adder, a comparator for ACS calculation (ACSCMP), 16-bit registers that is constructed 8 word (GR), and so on. Figure 4 shows the block diagram of DPU.

This unit manages the equalizing process, the (de)interleaving process, the generation of the Cyclic Redundancy Check (CRC) code, and the Viterbi decoding process.

We describe one example of the equalizing process. The calculation of equalizing process [5] consists of multiply-accumulate, multiply, addition, and so on. An example is shown below.

$$\Sigma(ArBr-AiBi),\ \Sigma(ArBi+AiBr) \qquad (1)$$

Ar,Br are real parts and Ai,Bi are imaginary parts of complex data. In expression (1), MULA and MULB calculate Ar x Bi and Ai x Br, and ALU adds ArBi and AiBr. The result (ArBi+AiBr) is stored into r0 that is one of the registers. At the next machine cycle, MULA, MULB, and ALU calculate the next data like the same as above, and adder adds r0 and r1. The r1 has held zero before this machine cycle. These processes are performed like pipeline operating, so DPU can calculate expression (1) in one machine cycle.

In generating the CRC code, the information data and the data of generator polynomial that are stored in MEU are input into r0 and r1. Then each data held in r0 and r1 are output to ALU. The ALU shifts the data of r0, that have information data, and calculate exclusive-or of r0 and r1, if the most significant bit (MSB) of r0 equals one. By repeating these procedures, the CRC code is generated.

Viterbi decoding process consists of branch metric calculation, comparison two path metrics, selection of less path metric, and updating path metric. It is realized by addition, comparison, and selection (ACS) process. In ACS calculation the ALU and the adder add two pairs of data and ACSCMP compares two results that the above unit outputs. Therefore the DPU can perform ACS calculation in one machine cycle.

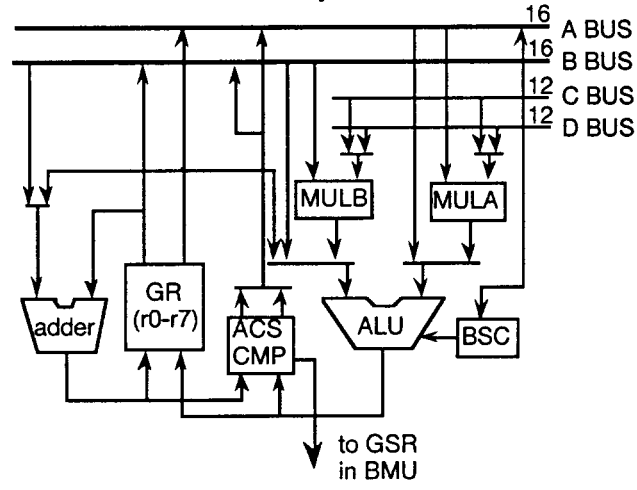

Fig.4 Block diagram of DPU

3.3 BMU

The BMU is for bit operations such as Fire (de)coding, convolution, encryption and decryption that are not adequate to process in the DPU. The main circuit is shift registers and exclusive-or logic. The connection

between shift registers and exclusive-or logic circuits can be changed by a value in a control register (BPCS). Figure 5 shows the block diagram of BMU.

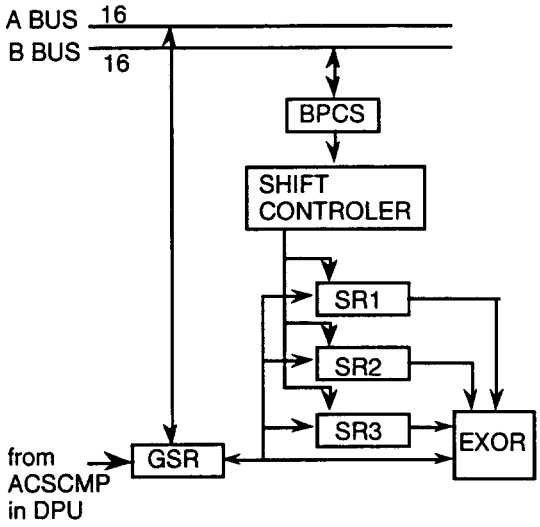

Fig.5 Block diagram of BMU

4. Conclusion

We have presented the architecture of EQCHAN and the outline of the LSI that contains EQCHAN. By the development of the new architecture, as we have described, EQCHAN realizes the process of equalizing, channel (de)coding, and encryption/decryption required by GSM hand portable terminal at 154ns per one machine cycle. The LSI that contains EQCHAN was designed in a 0.8μm CMOS technology and its die size is 123mm^2. The power consumption is 60mW at 3.6V under continuous communication mode. Table 1 shows the key features of EQCHAN.

Table 1 Key Features of EQCHAN

ALU	16bit op 16bit
ADDER	16bit + 16bit
MULTIPLIER	12bit×12bit (2units)
General Register (GR)	16bit×8word
Instruction Memory	28bit×4Kword
Data Memory (memory X, Y)	7Kbyte (total)
Instruction cycle times	154 ns (at 3.6V)

EQCHAN contributes to the low power consumption of the LSI by its high performance architecture, therefore it is very suitable for hand portable GSM terminals.

Acknowledgments

The authors would like to express their appreciation to K.Honma and H.Shibata for their helpful suggestions, and would like to thank T.Kayada for circuit design.

We wish to thank M.Yasutome for her contribution to constructing the simulator and testing of EQCHAN.

References

[1] H.J.Busschaert et al., "A Power Efficient Channel Coder/Decoder Chip for GSM Terminals," IEEE CICC91, pp.7.8.1-7.8.4, 1991.

[2] A.Spielberg et al., "ASIC Receiver for The Pan European Digital Cellular Telephone(GSM)," IEEE VTC92, pp.919-922, 1992.

[3] GSM Recommendations.

[4] K.Ueda et al., "A 16b Low-Power-Consumption Digital signal Processor," IEEE ISSCC93, pp.28-29, 1993.

[5] M.Uesugi et al., "Adaptive Equalization In TDMA Digital Mobile Radio," IEEE GLOBECOM89, pp.95-101, 1989.