Single-Chip Multi-Processor Integrating Quadruple 8-Way VLIW Processors with interface timing analysis considering power supply noise

Satoshi Imai

Atsuki Inoue

Motoaki Matsumura Ken

Kenichi Kawasaki

Atsuhiro Suga

System LSI development laboratories Fujitsu Laboratories Ltd. 4-1-1 Kamikodanaka, Nakhara-ku, Kawasaki, Japan Tel : +81-44-754-2783 Fax : +81-44-754-2691

e-mail : {imai.satoshi-02,inoue.atsuki,matsumura.motoa,k.kawasaki,suga.atsuhiro}@jp.fujitsu.com

Abstract - This paper introduces a 51.2Gops, 1.0GB/s-DMA single-chip multi-processor integrating quadruple cores and proposes a new power integrity analysis. Our multi-processor is designed to decode MP@HL streams without any dedicated circuits. To achieve such high performance, data throughput as well as processing capability is important, requiring a large number of high speed I/Os. However, this makes for a high level of power supply noise. We then applied an interface timing margin analysis tool that took power supply noise into account, and succeeded in putting reasonable restrictions on LSI design, as well as that for the printed circuit board. As a result, we succeeded in operating the processor at 533MHz with the 2ch 64bit main memory IF at 266MHz and 64bit system bus at 178MHz.

I. Introduction

There are two major methodologies for increasing the processing performance of processor: namely increasing the frequency and increasing the total number of processing elements in parallel. Processors for commercial products must satisfy three requirements: high performance, low pricing and low power consumption. For example, a processor for a personal computer achieves high performance; however, it requires a forced air cooling device or an expensive sealing package for the chip to work within a 100W power consumption limit. Use of such expensive the highly competitive devices for commercial supercomputer market is not possible. Generally, the price level for a built-in processor is set much lower than general purpose processors. Thereby, we have to use a less expensive package and cool it without using a fanning device. To meet this condition, we have to make a low priced LSI with low power consumption: approximately one fiftieth that of a PC microprocessor. Hence, we decided to achieve high performance and low power consumption by keeping the fundamental idea of FR-V and increase the total number of processing elements in parallel. We decided not to depend on increasing the frequency. The FR550[5] used VLIW and SIMD with eight instruction level paralleled processing and four data level paralleled processing (Fig. 1). Moreover, without increasing the frequency, it was necessary to run multiple chunks of instructions larger than those at the instruction level in parallel to boost the level of performance by a few times. As semiconductor process technology entered the 90nm age, the multi-core processor FR1000 with VLIW processor that we had planned at the beginning was finally realizable [6].

Our target was the realization of media processing, such as images and audio, exclusively by software. This kind of media processing was previously performed by configuring the control CPU with dedicated logic (ex. ASIC)[7]. Until now, a supercomputer or computing server divides the task into plural parallel tasks, mainly utilizing the loop parallelism in source code. However, a media processing program such as MPEG decoding includes in itself parallel tasks such as IDCT functions and a large chunk of slice level tasks. So we decided to equip the multiple core VLIW processor with our processor. A thread level (which is a larger chunk than instruction level) parallel processing was performed on multiple processor cores, and the instruction level parallel processing was performed in each processor core using VLIW architecture as well as the existing FR-V processor had done. As a result, we achieved higher performance and lower power consumption. For example, the decoding of MPEG2 MP@HL, which was six times the processing volume of MPEG2 MP@ML, consumed only about twice as much as power as the decoding of MP@ML streams by FR550.

Recently, power supply noise has become a critical design issue in LSI design. In particular, the problem of simultaneous switching noise (SSN) by I/O buffers is becoming serious. This is caused by the following two factors.

The first is an increase of SSN. Higher I/O bandwidth of system interface between LSIs causes an increase of dI/dt (I is the drive current of I/O buffers) as well as an increase of

the current itself. Because SSN is proportional to dI/dt and total inductance of power supply line, an increase of dI/dt leads an increase of SSN.

The second factor is a decrease in the noise tolerance of the circuits. The more advanced process technology becomes, the lower power supply voltage becomes in order to maintain transistor reliability. Lower voltage brings lower power consumption. However the noise margin of the circuits decreases significantly at the same time.

SSN not only causes false operation of logic circuits but also adds a delay penalty to the timing of the system interface between LSIs, such as high speed memory interface. So it is essential to evaluate the power supply noise and the delay penalty accurately in order to feed these results back to PCB designers.



Fig. 1. Effects of processor-core parallelism on the limits of VLIW architecture

II. Hardware Architecture

The FR1000 processor is a multi-core processor loaded on one chip with four VLIW processor cores inside. Each VLIW processor core is able to execute up to eight parallel processes at one time.

As stated above, the processor core on the FR1000 is a FR550 compatible processor. The instructions executed on the FR550 processor core consist of integer arithmetic instructions, floating point arithmetic instructions, and the media instructions with 16 bit fixed point arithmetic operations. Media instruction can process four or eight operations in parallel with the SIMD method. A processor core which can execute eight parallel instructions at the same time can execute twenty eight operations at the same time in one cycle. Consequently, the FR1000 processor is capable of executing 112 operations at the same time in one cycle.

A block diagram of this processor is shown in Fig. 2. In aggregate view, the chip consists of four processor cores, a main memory controller with two channels, a DMA controller (DMAC) for transfer between local memory units, a DMA controller (DMAC) for transfer to outside, and a 64 bit system bus interface. If the chip is configured with a

system that requires the processing of a huge volume of image data such as HDTV, a high speed/high precision printing system or a graphic system, it is very important to provide high speed capability to transfer data including I/O accesses, and to transfer data between memory units. High performance arithmetic processing capability is also very important. For example, in an arithmetic instruction, if the data transfer time from memory to arithmetic unit takes 90 cycles and the operation in the processor core is executed in 10 cycles, then ninety percent of total processor capability is controlled by the data transfer. The bandwidth of memory is not necessarily cause for concern with a single processor, however it becomes a major factor in a multi-core processor.

Consequently, we adopted the configuration below to avoid data transfer becoming a bottleneck in performance capability.

1) Four processor cores, two 64 bit channels of main memory interface at 266MHz, and with one of the system bus interfaces at 178MHz. Connections were made using cross bar methodology

2) To reduce external memory accesses, each processor core is equipped with 128KB SRAM (local memory unit) as local storage.

3) The DMA controller is functionally divided into DMAC for internal data transfer (internal DMAC) and DMAC for external data transfer (external DMAC), and each DMAC runs independently at the same time. The internal DMAC is used to control data transfer between processor cores, processor core and external memory, and memory and memory. The external DMAC is used to control the data transfer between memory and system bus.

4) In addition to the cross bar for data transfer described above, a communication control mechanism is installed for dedicating the instruction data transfer between processors.

The local memory unit, built into each processor core, is connected with a dedicated cross bar. All cores can access all local memory units. The internal DMAC and external DMAC described above are equipped with 16 channels respectively. With these bus architectures, the FR1000 processor can simultaneously process data transfer between the built-in local memory units, data transfer between areas on the memory, and data transfer between the memory and an external device. As shown Fig. 3, the data transfer speed between memory and external device reaches up to 1GB/s thanks to this architecture.



Fig. 2. Block Diagram of FR1000



Fig. 3. Data Transfer performance

III. Performance Evaluation

Fig. 5 shows the increase in application performance when optimizing the MPEG2 MP@HL decoding software. The vertical axis represents the operating frequency required for decoding the MPEG2 MP@HL. First, we applied MPEG2 MP@ML decoding software for single core to MP@HL decode and found that more than 1GHz of operating frequency is required. Next, we simply divided the decoding software into four cores using the method shown in Fig. 4. Large streaming data was divided into slice level data and processing for each slice level was assigned to each core. However, we could not increase the performance at all if we simply modified the program to fit the multi-core processor. We still needed over 1GHz of frequency (Single HL to 4PE HL in the Fig. 5). After we analyzed the memory access profile, we found that heavy memory access came from plural cores to the same main memory channel at the same time, which brought about a bottle neck in performance. This situation easily came about because many functions were accessing just a single memory unit at the same time in a multi-core environment. Accordingly, to reduce the load of memory accesses, we changed the memory map for MPEG2 decode processing and equalized the load of data accesses to memory as well. We thereby succeeded in increasing the performance to twice that of before this adjustment, requiring operating frequency for the MP@HL decoding to be reduced to approximately 500MHz.



Fig. 4. Parallel processing of MPEG@HL streaming data on four cores



Fig. 5. Performance optimization result on decoding MPEG-2 MP@HL

IV. Power integrity analysis

A. Power supply noise problem

Power supply noise inside LSI is becoming a critical design issue in realizing a high speed LSI with a large number of high speed I/Os, such as the FR1000 described in the previous chapter. We need to evaluate the power supply noise generated from LSI properly and its influence on the timing margin of high speed I/Os to provide reasonable design guidelines to PCB designers. The major concerns for LSI designers are as follows.

1) The SSN of I/O circuits is becoming large enough to interfere with other circuit operations and/or increase timing fluctuations.

2) The timing margin of system interfaces between LSIs is

decreasing as demand for higher I/O bandwidth is increasing.

In the case of FR1000, there are 230 SSTL I/O buffers for the two channels of memory interface which operate at a 266Mbps data rate, and 160 conventional CMOS I/O buffers for the system bus interface toggling at 178MHz.

The major factors reducing the timing margin are clock jitter and skew, the difference of the signal trace length on the package (PKG) and the printed circuit board (PCB), x-talk noise on PKG and PCB, and the I/O delay penalty caused by power supply noise. It is necessary to take all these factors into consideration to achieve robust design for processing and supply voltage variations. In particular, budgeting the interface timing, including the delay penalty by the SSN, is essential because the influence of the noise on timing is too large to be ignored.

In this chapter, we explain our method for analyzing power supply noise and delay penalty as well as their analysis models. Then, we discuss the analyzed results of the delay penalty for the FR1000's memory interface caused by power supply noise [1] [2].

B. Power supply noise analysis model

The power supply noise analysis model is a combination of three components: the DIE, PKG, and PCB models. To analyze the power supply noise accurately, not only PKG and PCB but also the DIE model should be generated properly, because the nature of the noise source is important to this analysis.

We proposed the LSI power model based on the power unit abstraction for the DIE, as shown in Fig. 6 [3]. The size of each power unit is typically chosen as 100 or 200 um squared. The model is composed of four parts: the power supply line network, capacitors, current sources, and I/O buffers. The power supply network is modeled as series LR circuit in each power unit. The capacitors represent the capacitance between power and ground such as decoupling capacitor, gate capacitance, and junction capacitance. The current sources represent the switching behavior of the logic gate and RAM inside each power unit. The I/O buffers are modeled separately in each pin as a transistor level netlist. We developed an automatic extraction tool from the layout design.

The PKG and PCB model are composed of power and ground supply network, signal traces, receivers, and decoupling capacitors. The signal traces are extracted as w-element or LRC ladder circuit, and the receivers are modeled as capacitor element.



Fig. 6. Power supply noise analysis model

C. Timing Analysis method

The analysis method of power supply noise is separated into two phases as shown in Fig. 7.



Fig. 7. Timing analysis method

In the power supply noise analysis phase, noise source and power supply network models are extracted from each set of layout data of DIE, PKG, and PCB. The power supply voltage waveforms of DIE, PKG, and PCB can be simulated by combining these models with appropriate stimulus.

In the timing analysis phase, the transmission line model is extracted from each set of layout data of DIE, PKG, and PCB. Then, the power supply voltage waveforms obtained in the previous noise simulation are applied to this model and the delay penalty of each I/O is evaluated. (Fig. 7, Fig. 8) In order to obtain the worst case delay penalty value, the generating timing of power supply noise is swept within possible timing window, as shown in Fig. 9. The signal arrives at the receiver either earlier or later depending on the applied supply noise timing. The largest timing fluctuation in the receiver is defined as the worst case delay penalty.



Fig. 8. Delay penalty analysis model

Finally, interface timing analysis is performed to check the timing margin considering this penalty, as well as others such as x-talk penalty, clock jitter, clock skew and so on.



Fig. 9. Delay penalty by power supply noise analysis method

D. Timing analysis of memory interface

We analyzed the delay penalty for the memory interface of FR1000 using the method described in the previous section. Because each signal operates at a different frequency and has different load capacitance and allowable timing margin, we classified the signals into four groups: address (Address), clock (Clock), data (DQ) and data strobe (DQS), and analyzed the I/O delay penalty for each group at first.

TABLE I shows the worst case delay penalty analysis results at the worst case process condition, supply voltage and temperature condition. The delay penalty of Address and Clock is larger than those of DQ and DQS. This is because the load capacitance of Address and Clock is larger than those of DQ and DQS, which causes the slew rate of Address and Clock to be larger than those of others. Generally, signals with a large slew rate are easily affected by power supply noise.

TABLE I
Worst case delay penalty results of memory interface

Signal	Delay penalty [ns]		
Sigilai	+⊿Delay	-⊿Delay	
DQ	0.284	0.220	
DQS	0.285	0.220	
Address	0.447	0.303	
Clock	0. 440	0.380	

Next, we discuss the timing analysis results of our memory interface. Figure 10 shows the write timing diagram of DQS and DQ at a 266Mbps data rate. Ideally, 1.88ns, which is the DQS-DQ phase difference on the driver buffer, is also kept at the memory input pins. The ideal setup margin of DQ against DQS becomes 1.38ns, assuming a 0.5ns setup time (tDS) taken from the main memory data specification sheet. Actually, this margin is reduced by various factors such as clock jitter and skew, a difference in the signal trace between DQS and DQ, and the I/O delay penalty caused by power supply noise. In particular, the delay penalty of DQS and DQ reduces the setup margin to 0.504ns (the sum of 284ps and 220ps) at the worst case process condition. In a word, the delay penalty reduces the setup margin by as much as 36.5%.

In TABLE II, other timing restrictions are summarized as well as the result of DQ-DQS timing analysis. From these results, we found that the delay penalty by power supply noise significantly influences the timing margin of the memory interface in the worst case. It is important to take it into account when providing appropriate design guidelines to PCB designers.

In a physical design of FR1000, we budgeted ideal timing margin to clock jitter and skew, flight time difference on PKG and PCB, x-talk noise delay penalty on PKG and PCB, and the I/O delay penalty by power supply noise for all of the interface signals. Though the I/O delay penalty was large, we succeeded in designing the FR1000, securing a sufficient interface timing margin by optimizing other timing budgets.



Fig. 10. Timing diagram of DQS and DQ for both driver and receiver

Timing restriction	Ideal margin [ns] ※ 1	Total delay penalty [ns]	Ratio [%] ※ 2
Clock-Address setup timing	2.86	0.827	28.9
Clock-Address hold timing	2.86	0.743	26.0
DQS-DQ setup timing	1.38	0.504	36.5
DQS-DQ hold timing	1.38	0.505	36.6
Clock-DQS skew	1.88	0.665	35.4

TABLE II Delay penalty of each signal group in memory interface

V. Chip implementation

TABLE III provides the specifications of FR1000. This processor was fabricated using a 90nm, nine-metal layer CMOS process technology. The number of transistors is 28M for logic and 55M for RAM. Transistors totalling 83M are integrated on a 10.3mm x 11.9mm die as shown in Fig. 11. The bus logic part, which operates at half core clock frequency, is arranged in center of the die.

The typical power supply voltage is 1.2V, and the power consumption measured to be 3.0W at MP@HL stream decode.

TABLE III Chip Specifications

Core	4 cores with 8-way VLIW architecture
Memory	32 KB+32 KB/core (D-cache, I-cache)
	128 KB/core (Local memory)
DMA controller	16 ch (Internal), 16 ch (External)
Interface	Main mem IF 266 MHz 64 bit x 2ch
	System Bus 178 MHz 64 bit
Technology	90-nm CMOS, 9-metal layers
Transistor count	28M (Logic), 55M (Memory)
Operating frequency	533 MHz @1.2 V
Power consumption	3.0 W @1.2 V, 533 MHz
Package	900-pin FCBGA



Fig. 11. Chip Die micrograph

VI. Conclusion

The FR1000 is designed to achieve 51.2Gops, 1.0GB/s-DMA and have four processor cores, internal DMAC, external DMAC, two 64bit channels of main memory interfaces at 266MHz, and a system bus interface at 178MHz. We demonstrated MPEG2 MP@HL stream decoding without any dedicated circuits with only a 3.0W power dissipation.

We investigated the influence of power supply noise and found the simultaneous switching noise by large numbers of high speed I/O buffers is enough large to reduce the timing margin of a high speed interface. We prepared reasonable PCB design guidelines taking this effect into account. Thus, we succeeded in operating the processor without any LSI re-spin.

Acknowledgements

The authors would like to thank Hiromasa Takahashi, Yukihito Kawabe, Wataru Shibamoto, Atsushi Sato, Tetsutaro Hashimoto, Hideo Miyake, Yasuki Nakamura, Hiroshi Okano, Fumihiko Hayakawa, Shinichiro Tago, Teruhiko Kamigata, Atsushi Tanaka and Takahisa Suzuki for their contributions to this work.

References

 M.Matsumura et al., "Analysis of the Effects of Simultaneous Switching Noise on System Interface Timing Between LSIs," Collected papers of the 18th workshop on circuits and systems in Karuizawa, pp.43-48, 2005
 K.Kawasaki et al., "Single-Chip Multi-Processor integrating Quadruple Processors on 90nm CMOS Process," IEICE Technical Report, Vol.105, No.95, pp.7-12, May.2005
 T.Sato et al., "LSI Noise Model for Power Integrity Analysis," FUJITSU, Vol.55, No.6, pp.608-613, Nov 2004
 Suga et al., "Introducing The FR500 Embedded microprocessor," IEEE Micro, pp. 21-27, July/Aug 2000.
 H. Okano et al., "An 8-way VLIW Embedded Multimedia Processor Build in 7-layer Metal 0.11µm CMOS Technology," ISSCC Dig. Tech. Papers, pp. 374-375, Feb. 2002.

[6] Shiota et al., "A 51.2GOPS, 1.0GB/s-DMA Single-Chip Multi-Processor Integrating Quadruple 8-Way VLIW Processors," ISSCC Dig. Tech. Papers, pp.18-19, 2005.

[7] Irvin M et al., "A new generation of MPEG-2 video encoder ASIC and its application to new technology markets," Broadcasting Convention, International (Conf. Publ. No. 428)12-16 Sept. 1996 Page(s):391 - 396.

<sup>X1. Ideal margin is the margin without considering any delay penalty.
X2. Ratio [%] = (Total delay penalty / Ideal margin) × 100</sup>