Comparative Evaluation of Multicore Dataflow DSPs with Different Arithmetic Units

Kenji Ichijo, Sora Deguchi and Akiko Narita
Graduate School of Science and Technology, Hirosaki University, Japan

***Abstract*-- DSPs and accelerators using FPGA technology and dataflow approach offer the potential for high performance in many applications including IoT-connected consumer electronics and so on. A dataflow approach provides the potential for exploiting parallelism inherent in programs. In our laboratory, we have developed a ring interconnected multicore dataflow DSP called LSC-Based DSP. In this work, we design five different kinds of arithmetic units including adder-subtractor, multiplier and divider. We implement LSC-Based DSPs with five different arithmetic units on an FPGA and evaluate them by comparison with both the logic resource usage and the execution time.**

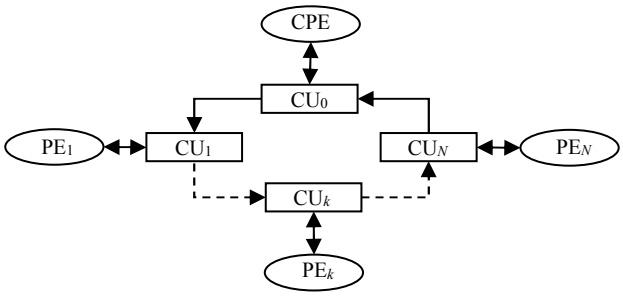
I. INTRODUCTION

DSPs and accelerators using FPGA technology and dataflow approach offer the potential for high performance in many applications including IoT-connected consumer electronics, edge computing and so on. A dataflow approach provides the potential for exploiting parallelism inherent in programs. According to the firing rules of the dataflow execution model, an instruction is executed as soon as its operands become available. Thus, a dataflow multicore processor executes several instructions in parallel, if their instructions fire on each core of the processor. In our laboratory, we have developed a ring interconnected multicore dataflow DSP called LSC-Based DSP. We implement it on an FPGA development board and evaluate its performance. In our previous work [1], [2] and so on, we mainly focus on improving the throughput of interconnection network in LSC-Based DSP to achieve the better performance. So there are many types of interconnection networks in LSC-Based DSP, while the number of operations in LSC-Based DSP is two, they are addition and multiplication operation. In our work [3], we supplement a subtraction operation to LSC-Based DSP in order to provide a higher degree of programmability.

In this work, we supplement a division operation to LSC-Based DSP, and design five kinds of arithmetic units from unit 1 to unit 5 including adder-subtractor, multiplier and divider. The adder-subtractor in all these arithmetic units is a parallel one. The difference between these units depends on a multiplier and divider. Unit 1 contains a parallel multiplier A (ParaMulA) and a parallel divider (ParaDiv). Unit 2 incorporates a parallel multiplier A (ParaMulA) and a serial divider (SeriDiv). Unit 3 retains a parallel multiplier B (ParaMulB) and a serial divider (SeriDiv). Unit 4 has a serial multiplier A (SeriMulA) and a serial divider (SeriDiv). Unit 5 owns a serial multiplier B (SeriMulB) and a serial divider (SeriDiv). We describe LSC-Based DSP designs with five different arithmetic units in VHDL. We implement each LSC-Based DSP on an FPGA development board, and measure the logic resource usage and maximum clock frequency. We execute test programs including addition, subtraction, multiplication and division instructions on our DSP, and measure the execution time of the program. Consequently, we evaluate our designed LSC-Based DSPs by comparison with both the logic resource usage and the execution time.

II. LSC-BASED DSP

We have developed various kinds of ring interconnected multicore dataflow DSP called LSC-Based DSP. The basic structure of LSC-Based DSP is shown in Fig. 1. LSC-Based DSP consists of an external communications processing element (CPE) and several dataflow processing elements (PEs). All dataflow PEs have an arithmetic unit. Each processing element has a communications unit (CU) and all CUs comprise a uni-directional ring interconnection network for transferring data among PEs by packet switching. This interconnection has the ability to avoid packet conflicts. A program is composed of a packet stream compiled from a dataflow graph. Available packets on each dataflow PE via the ring interconnection network are processed according to the firing rules of dataflow execution model. Fig. 2 shows two packet formats of LSC-Based DSP. The packet length is fixed at 56 bits. Fig. 2(a) is the format of instruction packet consists of seven fields. Fig. 2(b) is the format of data packet consists of three fields. One field is five bytes and the others are one byte. The first two fields, PE and PN, are common in two types of packet formats, which are distinguished by the most significant bit of the PN field. The PE field specifies a dataflow PE which must receive the packet. The PN field is used to check whether packets are available and executable. In the instruction packet, the FC filed identifies the operation code and the pair of fields DPE and DPN present



CU_0, … , CU_N: Communications Units
CPE: Communications PE
PE_1, … , PE_N: Dataflow PEs

Fig. 1. Basic structure of LSC-Based DSP.

PE	PN	FC	DPE1	DPN2	DPE2	DPN2

(a) Instruction packet

PE	PN	Fixed point data value (40 bits)

(b) Data packet

Fig. 2. Packet formats.

dependencies among instructions and operands in a dataflow graph.

III. ARITHMETIC UNITS

In this work, we design five kinds of arithmetic units from unit 1 to unit 5 for each dataflow PEs of LSC-Based DSP. The adder-subtractor in all these arithmetic units is a parallel one. The difference between these units depends on a multiplier and divider. Unit 1 contains ParaMulA and ParaDiv. Unit 2 incorporates ParaMulA and SeriDiv. Unit 3 retains ParaMulB and SeriDiv. Unit 4 has SeriMulA and SeriDiv. Unit 5 owns SeriMulB and SeriDiv. We choose Altera Cyclone IV EP4CE115 FPGA and Quartus II EDA tool as the implementation environment for developing LSC-Based DSP with their units. Clearly, a parallel arithmetic hardware outputs a result in one clock cycle, while the critical path become longer and the clock frequency become lower. In contrast, a serial arithmetic hardware outputs a result in multi clock cycles, while the critical path become shorter and the clock frequency become higher. Our designed multipliers are following four hardware: ParaMulA, ParaMulB, SeriMulA and SeriMulB. ParaMulA is described by operator "*" in VHDL, and implemented on embedded multiplier resources in above FPGA by Quartus II. ParaMulB is a well-known Wallace tree multiplier. SeriMulA is a well-known Booth multiplier. SeriMulB is a multiplier which outputs a result in half clock cycles of SeriMulA by using dual half sift registers stored every other bit of multiplier bits. On the other hand, our designed dividers are following two hardware: ParaDiv and SeriDiv. ParaDiv is a parallel division hardware generated by using Quartus II Mega Wizard Plug-In Manager. SeriDiv is a serial divider based on non-restoring division algorithm. We describe LSC-Based DSP designs with five different arithmetic units in VHDL. In the VHDL description, we also arrange the control unit on dataflow PEs according to each of our designed five kinds of arithmetic units from unit 1 to unit 5.

IV. EVALUATION AND CONCLUSION

By using Altera Quartus II EDA tool, we synthesize the VHDL design of each of five LSC-Based DSPs, and implement each DSP on Altera Cyclone IV EP4CE115 FPGA (114480 logic resources). Fig. 3 shows the logic resource usage of each DSP in the case that the number of dataflow PEs ranges from 2 to 10. In this figure, each of five LSC-Based DSPs is denoted by the name of each unit. Based Quartus II timing analyzer reports, we decide operation clock frequencies of both CU and PE as shown in Table I. We prepare a test program corresponding to the following equation: $x = [[\{\{(2\times4)+(-4\times0)\}-\{(1\times5)\div(2\times1)\}\}+\{\{(-1-4)+(-3-(-2))\}+\{(-3+2)\times(2\times1)\}\}]-[\{\{(-2\times(-2))\div(-5-4)\}\div\{(-5-4)-(-4\div(-1))\}\}-\{\{(4+4)\times(-2-(-1))\}+\{(-1\div5)\times(-5-1)\}\}]-[\{\{(4\div5)\times(0-2)\}\div\{(4\div5)\div(-1\times1)\}\}-\{\{(2\div3)-(-1-4))\}\div\{(4\times5)\div(-3-(-2))\}\}]\times[\{\{(-5-1)\times(2\div3)\}\times\{(-3-(-2))\div(0-3)\}\}\times\{\{(-5\times5)+(-5\div2)\}\times\{(-2\div4)\times(5-5)\}\}]]$. We measure the execution time of the program on each of five LSC-Based DSPs in the case that the number of dataflow PEs ranges from 2 to 10 as illustrated in Fig. 4. To measure the time, we use our designed measurement circuit embedded on the same FPGA that our DSPs are implemented. Consequently, we conclude that the LSC-Based DSP with unit 5 requires lower resources and achieves the better performance by our results.

REFERENCES

[1] K. Niiyama, K. Ichijo and Y. Yoshioka, "Execution time measurement of a vector operation on LSC-Based DSP, " *Proc. of Int. Conf. on ISPACS*, pp. 1-8, 2009.

[2] K. Ichijo, M. Kubota and A. Narita, "Design and Evaluation of a Multicore Dataflow DSP with Modified Ring Interconnection Networks, " *Information*, Vol.19, No.6(B), pp. 2403-2418, 2016.

[3] K. Ichijo, S Deguchi and A. Narita, "Expansion of Operations in a Multicore Dataflow DSP, " *Proc. of 2016 IEEE Int. Conf. on Consumer Electronics-Taiwan*, pp. 278-279, 2016.

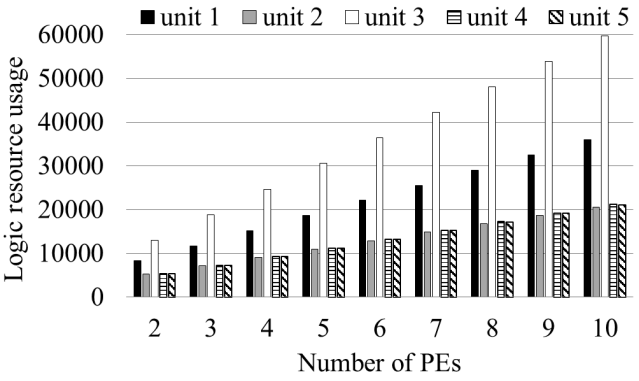


Fig. 3. Logic resource usage.

TABLE I
CLOCK FREQUENCIES [MHZ]

CU	PE with unit 1	PE with unit 2	PE with unit 3	PE with unit 4	PE with unit 5
100	5	25	25	50	50

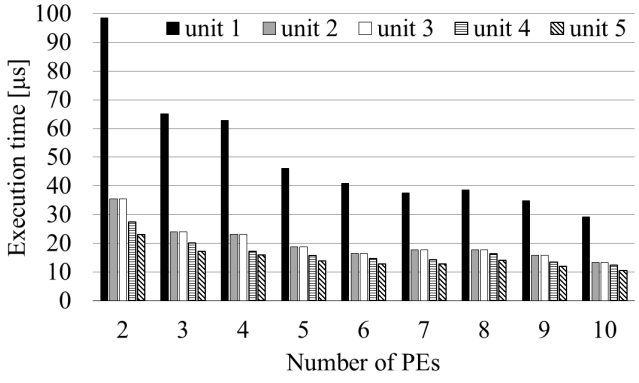


Fig. 4. Execution time.