Chapter5

Micro-Architectural Level Power and Performance Optimization Related to Multiplier-Accumulator

Previously, we have searched how to optimize a function unit (MAC) by using low power techniques at circuit level. In this chapter, we discussed that the optimization method of micro-architectural level which related to the MAC.

In Sec. 5.1, we overview a common DSP processor. A representative MAC unit will be presented in Sec.5.2. Furthermore, Sec. 5.3 presents some analytical equation of pipelining. Sec. 5.4 exploits parallelism on MAC unit. Finally, some micro-architecture optimization techniques of MAC will be discussed.

5.1 Common DSP Processors Architecture

DSP processors are microprocessors designed to perform digital signal processing - the mathematical manipulation of digitally represented signals. Today's DSP processors are sophisticated devices with powerful capabilities. In this section, we introduce the features of common DSP processors, explain some of the important concepts of it, and focus on the organization of common DSP processors.

Common DSP processors have many RISC-like features [5.1]. The major difference is that common DSP processors execute several operations in parallel while the RISC processors use heavily pipelined function unit. Therefore, the latency of a instruction in the RISC processors may be much longer than the DSP Processors.

Common DSP processors are generally characterized by the following architectural features:

1. A fast on-chip MAC unit that can perform multiplier-and-accumulator type operations within one instruction cycle. An instruction cycle which is generally one or two clock cycles long depending on its pipeline stage.
2. Several functional units that perform several parallel operations, including memory accesses and address calculations. The functional units have usually their own set of registers and most instructions
3. Several large on-chip memory units used to store instructions and data.

4. Several on-chip system buses to increase memory transfer rate.
5. Support for special addressing modes, especially modulo and bit-reversed addressing needed in the FFT. It is a dedicated hardware for address calculations.
6. Support for low-overhead loop and fast interrupt handling.
7. Standby power-saving capability. Only the peripherals and the memory are active in this mode.

In the classical von Neumann architecture the ALU and the control unit are connected to a single memory that stores both the data values and the program instructions. The main disadvantage is that memory bandwidth becomes the bottleneck in such an architecture. Therefore, the most common architecture which be used by the standard DSP processor is the Harvard architecture. Two separate memories are used in the classical Harvard architecture as shown in Figure 5.1. One of the memories is used independently for data while the other is used for instructions.

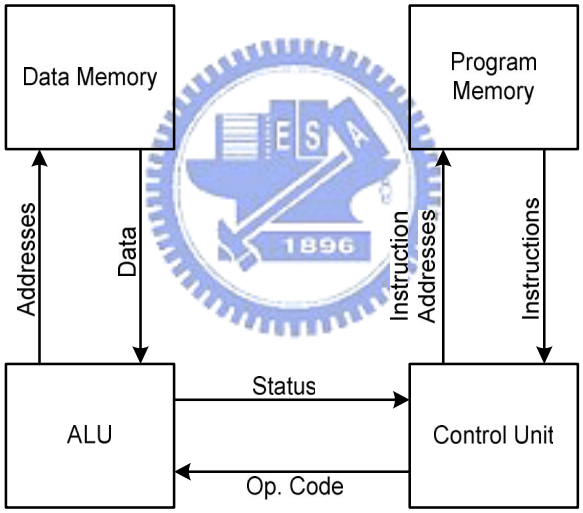

Figure 5.1 Harvard architecture

Most DSP processors share some common basic features designed to support high performance, repetitive, intensive tasks. The most evident of these features is the ability to perform one or more multiplier-accumulator operations in a single instruction cycle. The multiplier-accumulator operation is useful in DSP algorithms such as digital filter, and Fourier transforms. In order to achieve a single-cycle MAC operation, DSP processors integrate multiplier-accumulator unit into the main datapath of the processor. Some recent DSP processors provide two or more multiplier-accumulator units to increase its parallelism.

Another feature shared by DSP processors is the ability to complete several accesses to memory in a single instruction cycle. This allows the processor to fetch an instruction while simultaneously fetching operands and/or storing result of a previous instruction to memory. To support simultaneous access of multiple on-chip buses, multiple-port on-chip memories or register files are needed for such applications.

A third feature of DSP processors is one or more dedicated address generation units. This unit forming the addresses required for operand accesses in parallel with the execution of arithmetic instructions to speed arithmetic processing on DSP processors. Most DSP processors also provide special support for efficient looping of performing repetitive computations. This allows the programmer to implement a for-next loop without expending any instruction cycles for updating and testing the loop counter.

Finally, most DSP processors incorporate one or more serial or parallel I/O interfaces, and specialized I/O handling mechanisms such as low-overhead interrupts and direct memory access (DMA) to allow data transfers efficiently and achieve low-cost, high performance input and output.

The rising popularity of DSP functions in multimedia applications has led to consider implementing DSP on general-purpose processors such as desktop CPUs. For example, the MMX instruction set extensions are included into the Intel Pentium processors. However, because general-purpose processor architectures generally lack features for simplifying DSP programming, software development is sometimes more tedious than DSP processors and result in difficulty of code maintaining. Thus, if general-purpose processors are used only for signal processing, they are rarely cost-effective compared to DSP processors designed specifically for the dedicated task. Therefore, the system designer should continue to use traditional DSP processors for DSP intensive applications.

5.2 MAC Units for Common DSP Processors

Multiplier-accumulator functional units can be used to compute vector dot products very efficiently. MAC units, therefore, is useful in a large class of important signal processing algorithms. Figure 5.2 shows a skeleton of MAC units in DSP processors.

Figure 5.3 shows a representative MAC unit of common DSP processors [5.2]. The MAC unit performs two basic functions: multiply and multiply-accumulation. The functionality of MAC unit is chosen by its configuration state which is stored in a register. The two input operands of the MAC unit are 16-bit signed integers. The MAC unit has a 40-bit accumulator which allowing it to accumulate up to 256 32-bit products without overflow.

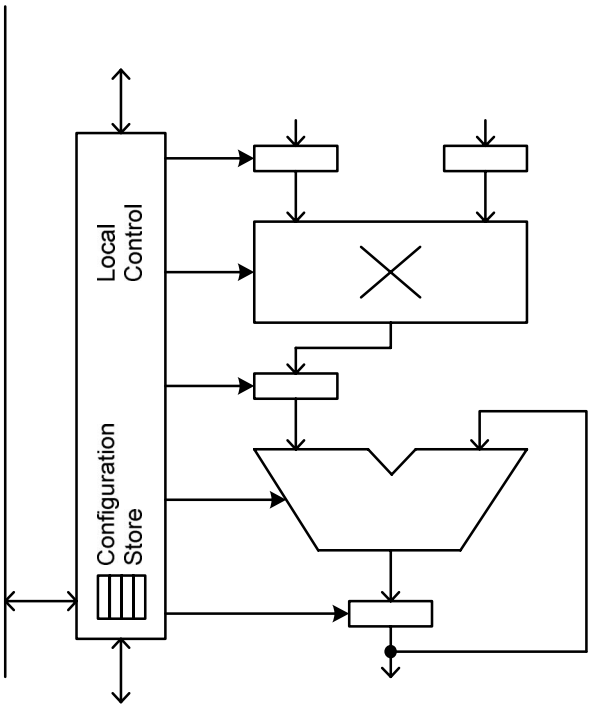

Figure 5.2 Skeleton of a MAC unit in DSP processors

The clock signals, CK1,CK2, and CK3, are generated by the asynchronous handshake controller of MAC unit. The MAC unit has two pipeline stages. In the first stage, the Booth Encoder with a column compression tree to add 8 partial products which are generated by the Booth encoder. The output of column compression tree is in carry-save format and consists of a 32-bit sum vector and a 32-bit carry vector. The 40-bit output of the accumulator register, which is also in carry-save format, is added with the output of the column compression tree to generate the result of the first pipeline stage. For a multiply operation, this result is loaded into the pipeline register, CK3 is enable. Moreover, the CK2 clock is disable, and 40-bit carry and sum vectors are set to zero. For a multiply-accumulation operation, the result of the first pipeline stage is loaded only into the accumulator register, CK3 is disable. In the second stage, the multiply or multiply-accumulation operation will be completed by carry propagation adder (CPA). The output of the CPA is shifted right by designated position (specified by the configuration state of the MAC unit), and the least significant 16-bits of the shifted result form the output of the MAC unit.

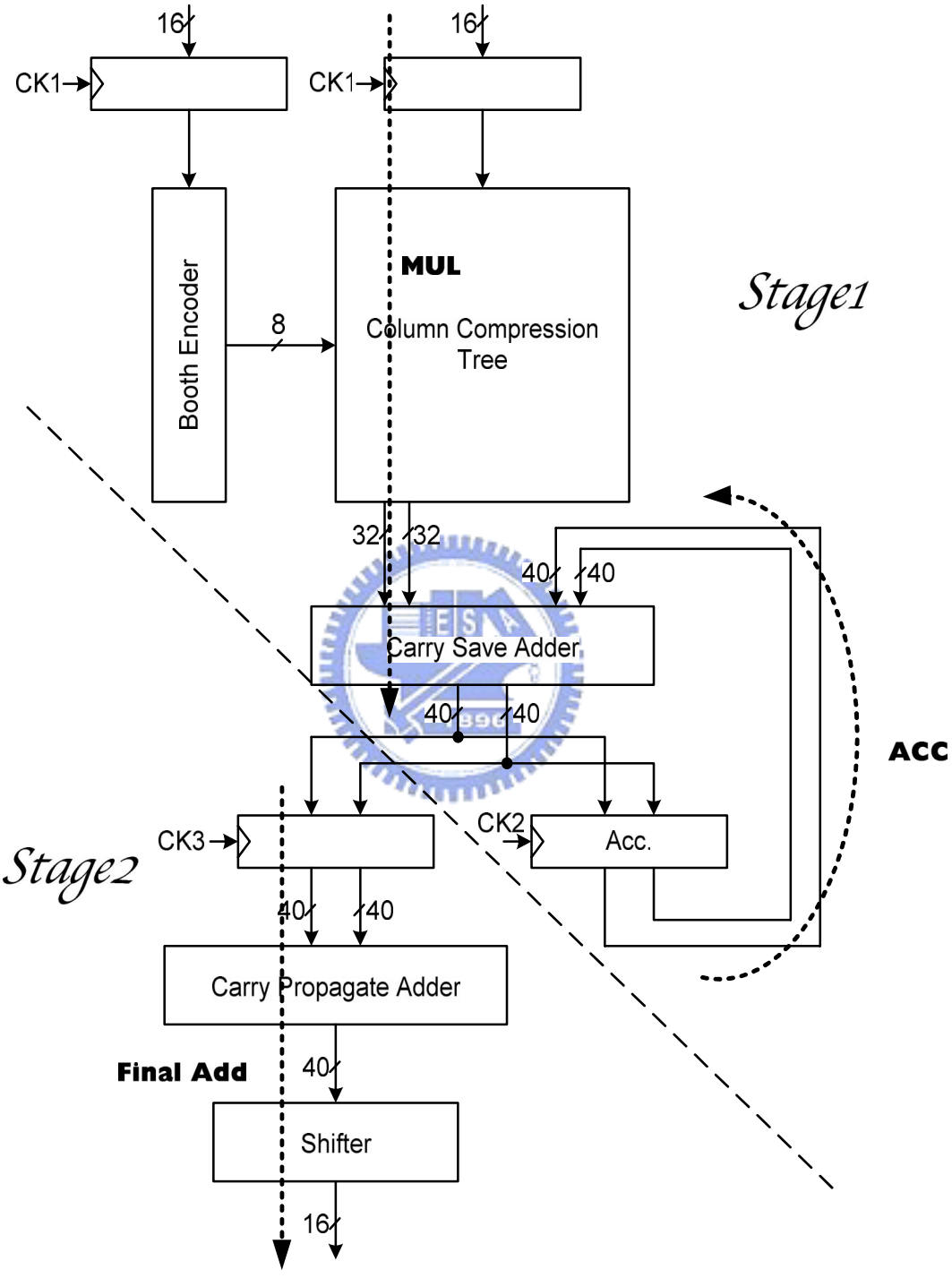

Figure 5.3 Block diagram of the MAC functional unit

5.3 Power-Optimum Pipelining

Figure 5.4 shows schematics of the normal, pipeline, and parallel circuit design. Parallelism and pipelining are used to alleviate timing constrains on combinational circuit A and B when the power of original design is too large. In a parallel design, the area is doubled by applying the two identical hardware in parallel. In pipelining, an additional latches or flip-flops are inserted between circuit block A and B.

A parallel architecture, we will discuss later, could be used to provide excess performance to trade for power, but pipelining has the advantage of lower power and lower area penalty than parallelism. By using the method of pipelining to reduce power, the only limitation is the power overhead of the additional pipeline latches or flip-flop required for each pipeline stage. In this section, we discuss the tradeoffs between pipeline depth, supply voltage, and total power consumption analytically [5.7].

5.3.1 Pipelining versus Supply Voltage

We begin by discussing the effect of pipeline depth with supply voltage. As pipeline stage increases, supply voltage could be scaled down to save power while maintaining speed, because the reduction of logic amount per pipeline stage. The circuit delay can be approximately given by

$$delay \propto (N+k)\times\frac{Vdd}{(Vdd-Vth)^{\alpha}} \tag{5.1}$$

where N is the logic depth per pipeline, k is the timing element delay, a is a velocity saturation factor, Vdd and Vth are supply and threshold voltages respectively.

In deep submicron technology the value of a is close to 1.5. For convenience, we assuming a is 2 to get

$$N+k \propto Vdd-2Vth+\frac{Vth^2}{Vdd} \tag{5.2}$$

Assume that $\frac{Vth^2}{Vdd}$ is close to zero, we can get a simple linear equation between Vdd and N, where $a0$ is a constant:

$$Vdd = a0\cdot N + a1 \tag{5.3}$$

$$\frac{a1}{a0} = k + \frac{2}{a0}Vth \tag{5.4}$$

Equation 5.3 indicates that supply voltage which be scaled as we required is to trade timing of logic depth per pipeline stage for power. More pipeline stages results in supply voltage scaling and lower power consumption.

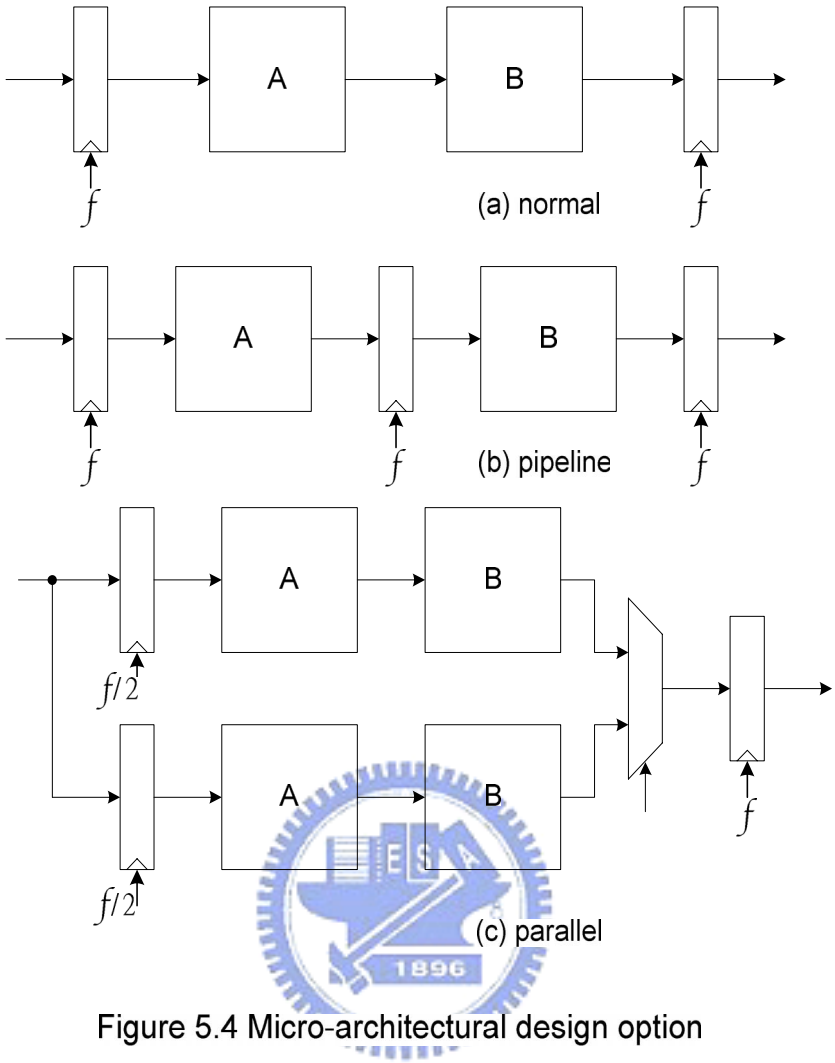

Figure 5.4 Micro-architectural design option

5.3.2 Optimum Logic Depth per Pipeline Stage

The switching power of a pipelined logic stage can be divided into two part, one is combinational circuit block, the other is timing element.

$$P_{switching} = (b_0 + \frac{b_1}{N})V_{dd}^2 \tag{5.5}$$

$$= b_0 a_0^2 (1 + \frac{b_1}{b_0}\frac{1}{N})(N + \frac{a_1}{a_0})^2 \tag{5.6}$$

where b_0 represents the coefficient of timing element, b_1 represents the coefficient of combinational circuit block. Note that the term $(N + \frac{a_1}{a_0})^2$ in Equation 5.6 makes $P_{switching}$ scale down slowly when a_1/a_0 is large.

We assume that the number of latches increase linearly with the number of pipeline stage. In Equation 5.6 the ratio of the parasitic capacitances of combinational circuit block and timing element. When N is much greater than a_1/a_0 and b_1/b_0, $P_{switching}$ becomes as follow:

$$P_{switching} \approx b_0 a_0^2 N^2 \tag{5.7}$$

On the other hand, if N is much smaller than a_1/a_0 and b_1/b_0, $P_{switching}$ becomes inversely proportional to N:

$$P_{switching} \approx b_1 a_1^2 \frac{1}{N} \tag{5.8}$$

The optimum logic depth N* is given by:

$$N^* = \frac{1}{4}\left(\sqrt{\frac{b_1^2}{b_0} + 8\frac{a_1}{a_0}\frac{b_1}{b_0}} - \frac{b_1}{b_0}\right) \tag{5.9}$$

Equation 5.9 indicates that larger parasitic capacitances of timing element lead to less deep pipeline stages. But the timing element delay k which is shown in Equation 5.4 should affect optimum N* and correspondingly optimum power saving more heavily.

5.4 Parallelism Exploitation to Improve Performance

Figure 5.5 categorizes types of parallelism and possible mechanisms for exploitation within DSP processors [5.3]. DSP algorithms usually offer large amount of data level parallelism (DLP). While parallel processing can also be extracted from independent instructions within programs, is called instruction level parallelism (ILP). Advanced DSPs employ VLIW techniques accompany with SIMD feature to increase its parallelism and very few use superscalar approach. Our objective in this section is to discuss the architecture of MAC unit in different types of parallel DSP processors. We will mainly focus on VLIW DSP processors and the one with SIMD feature.

VLIW based DSP processors become more and more notable because they enable the development of high-level language compilers that generate efficient code which will be especially helpful in reducing development time for the DSP programmers. On the other hand, the superscalar architectures would require more hardware complexity because the superscalar architectures need a hardware logic block to find instruction-level parallelism among instructions of codes, hence it's a unfavorable design option of DSP architecture.

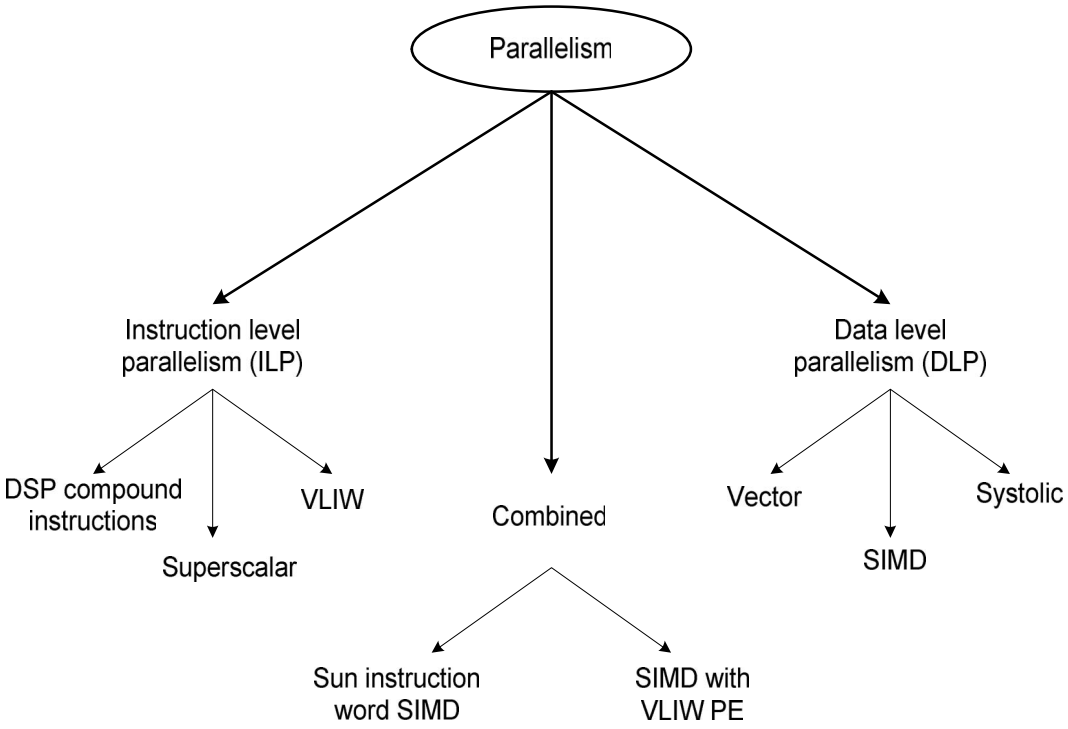

Figure 5.5 Parallelism exploitation in DSP processors

In recent advanced DSP processors design, the application-specific enhancements are also integrated into its instruction set architecture. Such application-specific enhancements are valuable when their application are actually in use. But they do nothing to enhance the performance of other applications and result in wasting chip size as well as energy consumption. Therefore, the choice of such enhancements have to be make carefully and in a balanced way.

Figure 5.6 shows a datapath block diagram of eight-way VLIW architecture of TI® C64X DSP core [5.4]. The eight function units in the C64X datapath can be divided into two groups of four, each function unit in one datapath is almost identical to the corresponding unit in the other datapath. The function unit which related to MAC operations are described in Table 5.1 [5.5]. The C64X multiplier unit is capable of performing two 16-bit or four 8-bit multiplies per cycle and optionally add the result together, which delivers 2400 16-bit Million Multiply Accumulations per Second (MMACs) or 4800 MMACs. Such SIMD instructions are provided to take advantage of data-level parallelism (DLP) result in performance improvement of media streams operations.

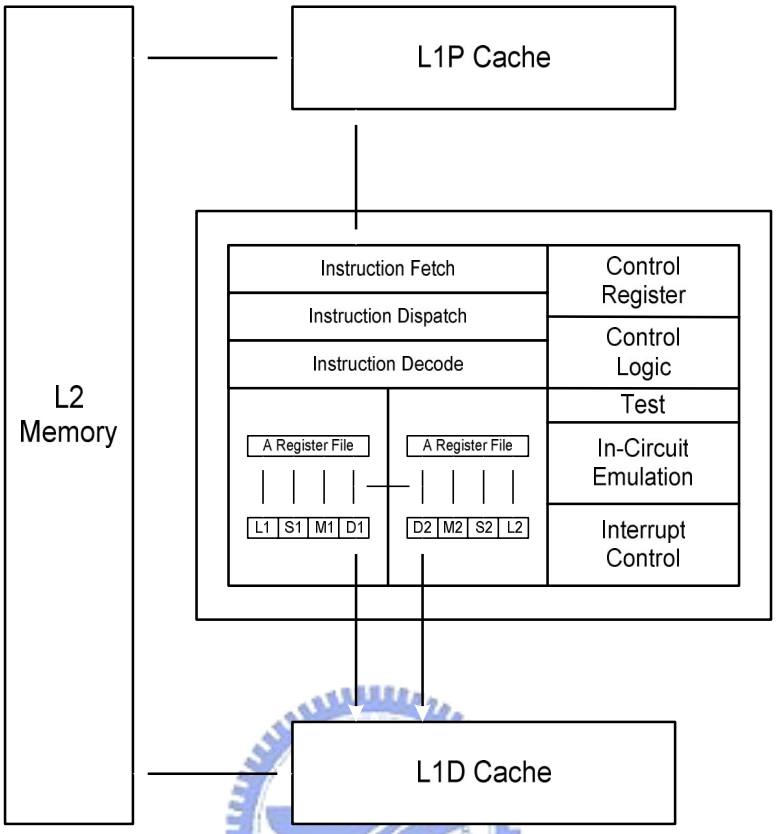

Figure 5.6 Block diagram of C64X DSP core

Table 5.1 MAC operations of C64X DSP processor

Function Unit	Fixed-Point Operations	Fixed-Point Operations
.M unit (.M1, .M2)	16 x 16 multiply operations 16 x 32 multiply operations Quad 8 x 8 multiply operations Dual 16 x 16 multiply operations Dual 16 x 16 multiply with add/subtract operations Quad 8 x 8 multiply with add operation Bit expansion Bit interleaving/de-interleaving Variable shift operations Rotation Galois Field Multiply	32 X 32-bit fixed-point multiply operations Floating-point multiply operations

In [5.6], a representative SIMD featured MAC unit is presented. One-cycle 16 x 16 and 32 x 16 MAC instructions are implemented for increasing the throughputs of many DSP algorithms. It is a coprocessor of the Intel® XScale™ RISC processor. This coprocessor make several additional 16-bit DSP features to meet the specific need of various applications. Figure 5.7 illustrates one of the SIMD instruction which performs two 16 x 16 multiplications and a 40-bit addition. Dual signed 16x16 (SIMD) multiplier-accumulators multiply the high/high and low/low 16-bits of a packed 32-bit multiplier and another packed 32-bit multiplicand to produce two 16-bits products which are both sign-extended to 40-bits and then both added to the 40-bit accumulator. Figure 5.8 illustrates another one which is more complicated involves 16x16 signed multiplier-accumulators multiply either the high/high, low/low, high/low, or low/high 16-bits of a 32-bit multiplier and another 32-bit multiplicand to produce a full 32-bit product which is sign-extended to 40-bits and then added to the 40-bit accumulator. The difference between these two SIMD instructions is that the MUXs which are inserted into the later SIMD instructions.

The combination of other basic MAC instruction and SIMD instructions allows a programmer to create tight code for handling media streams.

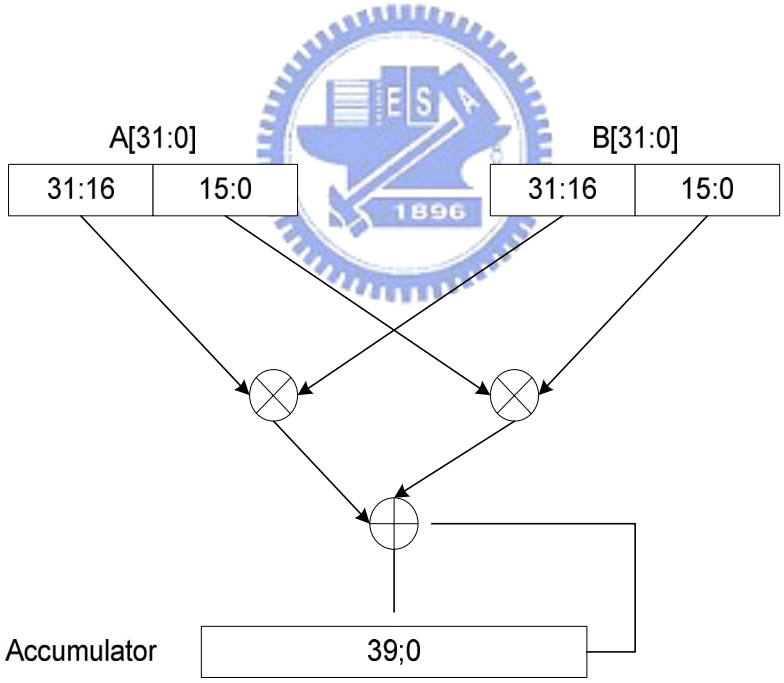

Figure 5.7 Only one combination of 16 bit entities of SIMD instruction that performs 16x16 multiplications and a 40-bit addition

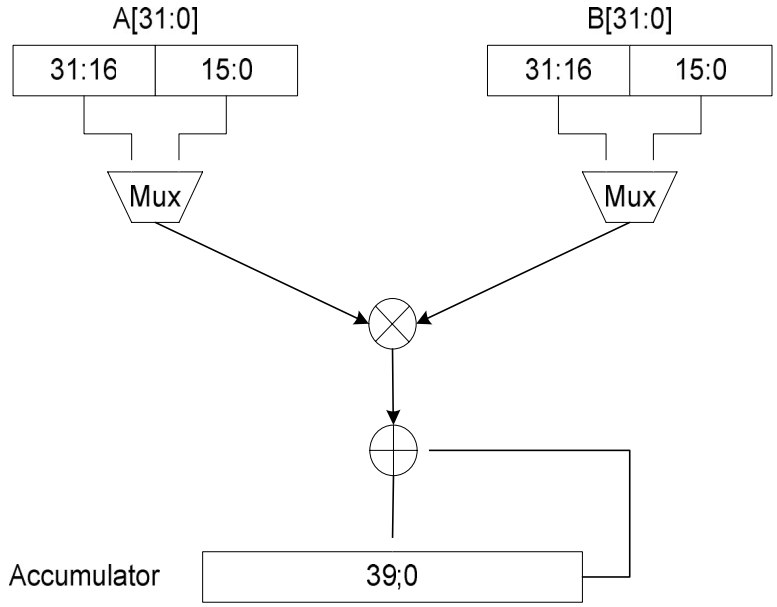

Figure 5.8 Four combinations of 16 bit entities of SIMD instruction that performs 16x16 multiplications and a 40-bit addition

5.5 Reconfigurable Power-Aware Architecture Design

Recently, there are growing demands of multimedia applications which require intensive arithmetic computations on variable precision data [5.8]. Multiply and accumulation often has the largest impact on the instruction cycle time of a DSP processor. Therefore, the use of reconfigurable multiplier-accumulator operating on variable precision data can represent a good choice to accommodate these computational requirements [5.9] [5.10] [5.11] [5.12]. In addition, several literatures describes some novel methodologies for designing reconfigurable pipelines [5.13] [5.14] [5.15] that achieve very low power dissipation by disabling and bypassing an appropriate number of pipeline stages whenever data rates are low.

5.5.1 Variable Precision Multiplier Architecture

In this section, two variable precision reconfigurable multiplier-accumulator are presented. Reconfigurability makes our multiplier-accumulator unit able to support parallel signed/unsigned multiply and accumulation on data with different wordlengths. Figure 5.9 and Figure 5.10 shows two manners of implementing variable precision multiplier-accumulator.

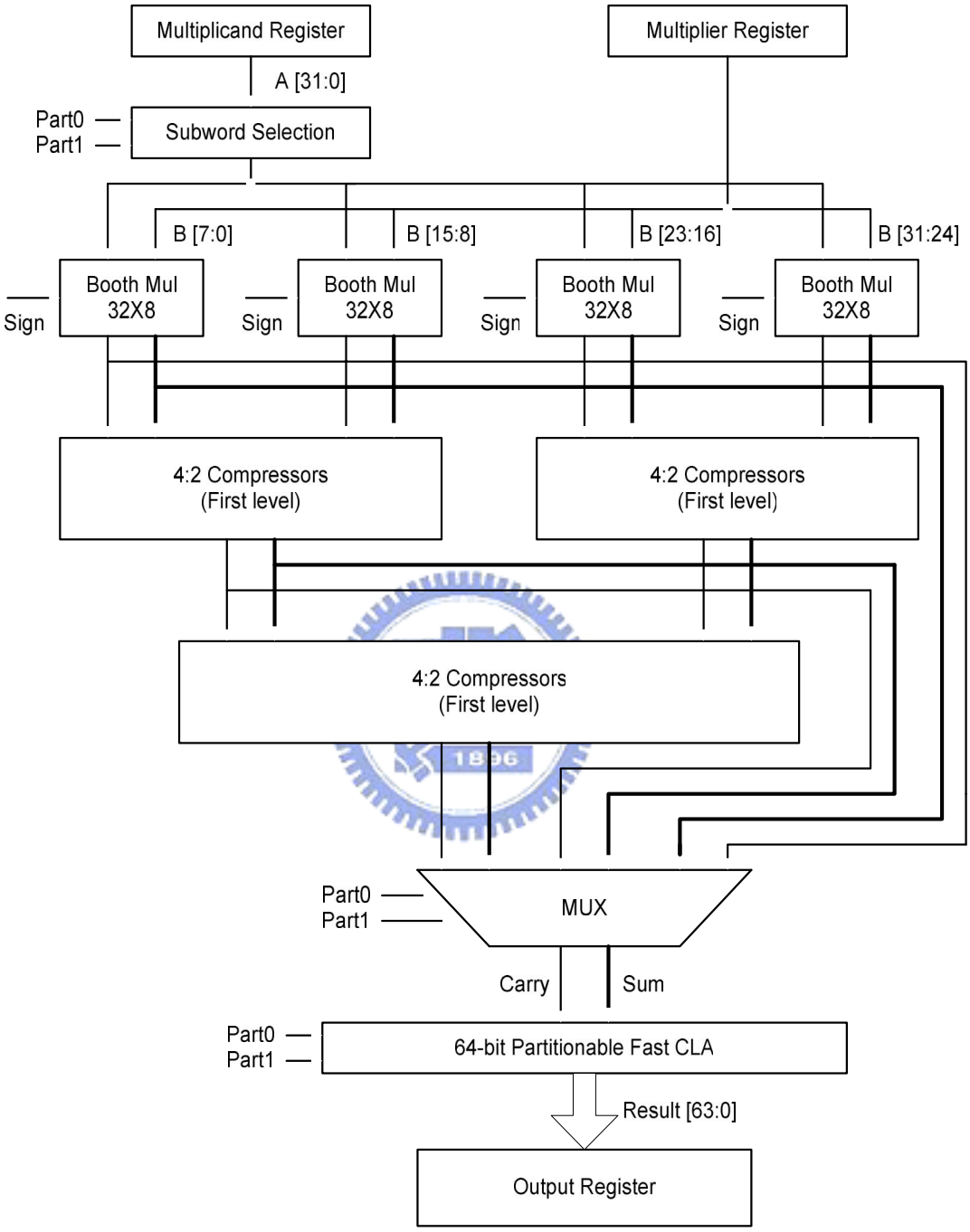

Figure 5.9 Variable precision multiplier architecture

Figure 5.9 is based on the observation that the result of a 32-bit binary multiplication A[31:0] * B[31:0] can be produced as shown in Equation 5.1:

$$\begin{aligned} A[31:0] * B[31:0] = {} & LLS16 (LLS8 (A[31:0] * B[31:24]) + \\ & EX48 (A[31:0] * B[23:16])) + \\ & EX64 (LLS8 (A[31:0] * B[15:8]) + \\ & EX48 (A[31:0] * B[23:16])) \end{aligned} \tag{5.1}$$

where LLSd and EXc indicate a logical left shift by d bit positions and a word extension to c bits, respectively.

Equation 5.1 shows that a 32*32-bit multiplier can be realized by using four 32*8-bit multipliers. The four independent results obtained in this way can be easily combined to generate the whole 64-bit result. The main advantage obtained using this approach resides in the possibility of performing also two parallel 16*16-bit multiplications or four 8*8-bit independent multiplications.

What operand wordlength the multiplier has to operate on is established by two control signals, Part1 and Part0, and what kind of data have to be elaborated (signed or unsigned) is indicated by a third control signal Sign. The possible operation modes of the variable precision multiplier are summarized in Table 5.2.

Table 5.2 Control words supported by the configurable multiplier

Part1	Part2	Sign	Control Word
0	0	0	4 packed 8-bit unsigned mult
0	0	1	4 packed 8-bit signed mult
0	1	0	2 packed 16-bit unsigned mult
0	1	1	2 packed 16-bit signed mult
1	0	0	32-bit unsigned mult
1	0	1	32-bit signed mult

Figure 5.10 is based on recursive multiplier [5.16] that the result of a n-bit binary multiplication A[n:0] * B[n:0] can be produced as follow.

Mathematically, the recursive algorithm may be proved by first considering two unsigned n-bit operands, the multiplier A and multiplicand B:

$$A = \sum_{k=0}^{n-1} a_k \cdot 2^k \qquad B = \sum_{k=0}^{n-1} b_k \cdot 2^k \tag{5.2}$$

By dividing each of the two operands into 2m-bit values, where m = n/2, we obtain:

$$A = \sum_{k=0}^{m-1} a_k \cdot 2^k + \sum_{k=m}^{2m-1} a_k \cdot 2^k \qquad B = \sum_{k=0}^{m-1} b_k \cdot 2^k + \sum_{k=m}^{2m-1} b_k \cdot 2^k \tag{5.2}$$

A and B may now defined as: $A = A_L + A_H$

$$B = B_L + B_H \tag{5.3}$$

The overall multiplication of A and B is given by:

$$\begin{aligned} P &= A \cdot X \\ &= (A_L + A_H) \cdot (B_L + B_H) \\ &= A_L \cdot B_L + A_L \cdot B_H + A_H \cdot B_L + A_H \cdot B_H \\ &= P0 + P1 + P2 + P3 \end{aligned} \tag{5.4}$$

Therefore, the overall multiplication may be reduced to four smaller multiplications, and this process may be repeated using even smaller base multipliers. Figure 5.10 shows the recursive multiplier architecture which with one level of recursion will be used as the foundation for the reconfigurable architecture. For variable precision of multiplication, this scheme utilizes 2-bit control signal to select one of four precisions of operation. Since all of the necessary components for each precision of operation are present in the design, there will be no reconfiguration time required, enabling the device to switch one of four precision of operation.

5.5.2 Power Aware Variable Pipeline Stage Architecture

This subsection discussed a reconfigurable pipelined architecture that achieves high performance and low power dissipation by adapting its structure to computational requirement. In [5.15], whenever throughput requirements are low, register stages are selectively disabled by gated clocks and bypassed by multiplexers. Figure 5.11 shows 4 stage pipelined reconfigurable structure. The throughput of a conventional pipelined structure is fixed at one operation cycle, while the throughput of this configurable pipelined structure may be set to one operation every one, two, or four cycles, depending on the input data rates. As Figure 5.11 shows, three register stages are disabled by gated clocks and bypassed through multiplexers, thus saving a significant fraction of the datapath's total power dissipation in the reconfigured datapath. The pipeline depth of the datapath is dynamically controlled depending on the throughput requirement and therefore very beneficial in processing many of such kind of applications.

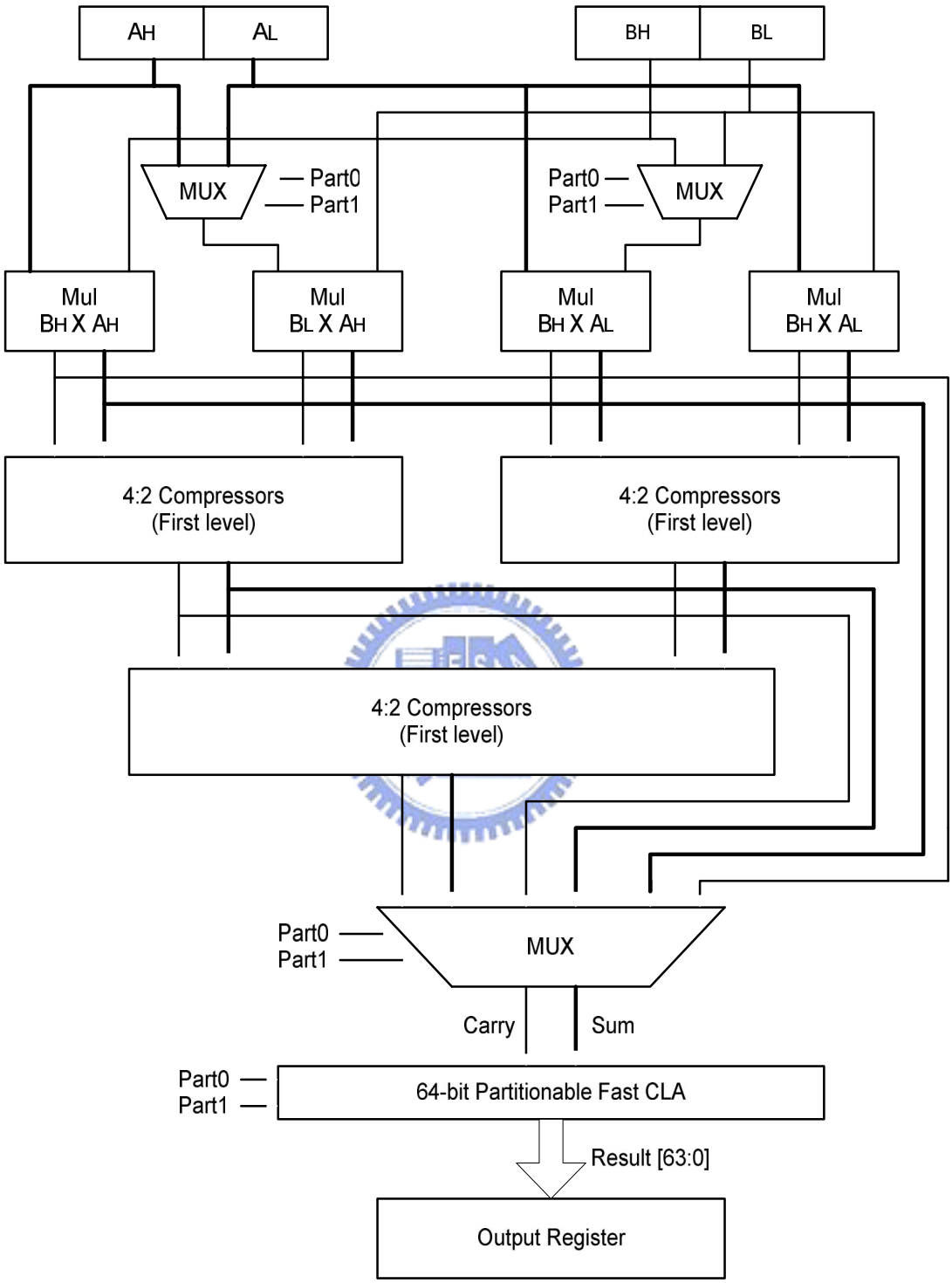

Figure 5.10 Recursive variable precision multiplier architecture

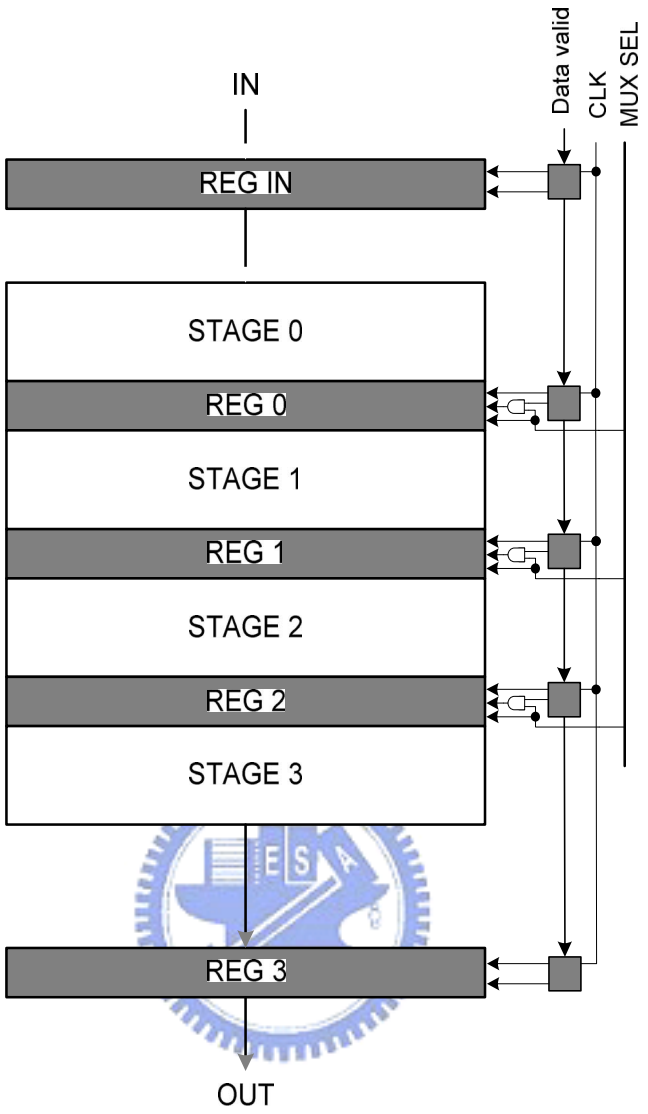

Figure 5.11 Reconfigurable 4-stage pipeline

5.6 Conclusions

In this chapter, we have discussed the MAC unit for common DSP processors. We also examined micro-architectural optimization methods of power and speed. The key of power optimum pipelining is the determination of proper supply voltage and logic depth per pipeline stage. The VLIW architecture DSP processors combined with SIMD featured instructions increase both instruction level parallelism (ILP) and data level parallelism (DLP) for achieving high performance. Finally, the reconfigurable architecture of MAC are discussed. Variable precision and variable pipeline stage of multiplier are discussed briefly.

Chapter6

Conclusions

6.1 Summary

In this thesis, we have investigated high-speed micro-architecture design as well as circuit-level optimization techniques for achieving high-speed and low-power MAC. We have addressed design problems from two aspects: Firstly, to achieve high-speed the efficient multiplier micro-architecture is considered. Secondly, to achieve low-power the circuit-level optimization method is proposed. For micro-architecture efforts, we considered several existing recoding scheme for partial product generation. Two classes of combinational multipliers are considered: linear array multiplier and tree multipliers. Three efficient micro-architectures of parallel adders are evaluated. For circuit-level efforts, logical effort model is used as a way of comparing XOR gates. Circuit topologies and circuit style of 5-2 compressors are compared in terms of power and speed. Proposed power-speed optimization techniques move the original design point as close as to optimum design point in terms of power and speed.

6.2 Future Work

As an attempt to design further high-speed and low-power MAC micro-architecture, one possible direction is higher radix Booth recoding scheme. We have only considered radix-4 recoding as it is simple and popular choice. High-radix recoding further reduces the number of partial products and thus has the potential of power saving. The difficulty of designing higher radix recoding is to generate hard partial products such as 3X. This may introduce additional delay and design complexity which can be trade for power saving. Thus, there is power/delay/area tradeoff in high radix recoding.

In order to truly minimize the power in a chip, it is necessary to optimize all design layers simultaneously to achieve the optimum balance between power and performance [2.21]. However, in this study, we have only considered power-speed optimization at circuit-level, which is clearly not the case in globally optimum. In fact, extra degrees of freedom at micro-architectural level, such as parallelism, pipelining, and reconfigurable design, allow for power-delay optimization over a wider range. Another possible direction is to develop reconfigurable MAC micro-architecture such as reconfigurable

pipeline stage of MAC or partitionable MAC. Reconfigurable pipeline stage is often desirable in power-efficient applications. While partitionable MACs are becoming more important because the data precisions are very widely in different applications [6.1]. These reconfigurable micro-architectures provide wider tradeoff space for power and performance.

References

References of Chapter1

[1.1] Wei Hwang, “New trends in low power SoC design technologies,” SOC Conference, 2003. Proceedings. IEEE International [Systems-on-Chip], 17-20 Sept. 2003 Pages:422

[1.2] International Technology Roadmap for Semiconductors 2001 edition, Semiconductor Industry Association, http://public.itrs.net

References of Chapter2

[2.1] Murakami, H.; Yano, N.; Ootaguro, Y.; Sugeno, Y.; Ueno, M.; Muroya, Y.; Aramaki, T.; “A multiplier-accumulator macro for a 45 MIPS embedded RISC processor” Solid-State Circuits, IEEE Journal of , Volume: 31 , Issue: 7 , July 1996 Pages:1067 – 1071

[2.2] Elguibaly, F.; “A fast parallel multiplier-accumulator using the modified Booth algorithm” Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on] , Volume: 47 , Issue: 9 , Sept. 2000 Pages:902 – 908

[2.3] Ohsang Kwon; Nowka, K.; Swartzlander, E.E.; “A 16-bit×16-bit MAC design using fast 5:2 compressors” Application-Specific Systems, Architectures, and Processors, 2000. Proceedings. IEEE International Conference on , 10-12 July 2000 Pages:235 - 243

[2.4] Fujino, M.; Moshnyaga, V.G.; “Dynamic operand transformation for low-power multiplier-accumulator design” Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on, Volume: 5 , 25-28 May 2003 Pages:V-345 - V-348 vol.5

[2.5] Bartlett, V.A.; Grass, E.; “A low-power concurrent multiplier-accumulator using conditional evaluation” Electronics, Circuits and Systems, 1999. Proceedings of ICECS '99. The 6th IEEE International Conference on, Volume: 2 , 5-8 Sept. 1999 Pages:629 - 633 vol.2

[2.6] Kabuo, H.; Okamoto, M.; Tanaka, R.; Yasoshima, H.; Marui, S.; Yamasaki, M.; Sugimura, T.; Ueda, K.; Ishikawa, T.; Suzuki, H.; Asahi, R.; “A 16 bit low-power-consumption digital signal processor using a 80 MOPS redundant binary MAC” VLSI Circuits, 1995. Digest of Technical Papers, 1995 Symposium on , 8-10 June 1995 Pages:63 – 64

[2.7] Seung-Min Lee; Jin-Hong Chung; Hyung-Seok Yoon; Lee, M.M.-O.; “66 M/70 mW HS and ultra-low power 16×16 MAC design using TG for web-based multimedia system” ASICs, 1999. AP-ASIC '99. The First IEEE Asia Pacific Conference on , 23-25 Aug. 1999 Pages:151 – 153

[2.8] Seung-Min Lee; Jin-Hong Chung; Hyung-Seok Yoon; Mike Myung-Ok Lee; “High speed and ultra-low power 16×16 MAC design using TG techniques for web-based multimedia system” Design Automation Conference, 2000. Proceedings of the ASP-DAC 2000. Asia and South Pacific , 25-28 Jan. 2000 Pages:17 – 18

[2.9] Lee, M.M.-O.; Seung-Min Lee; ”A high performance MAC design using proposed low power IP-cells” ASIC, 2001. Proceedings. 4th International Conference on , 23-25 Oct. 2001 Pages:596 – 598

[2.10] Izumikawa, M.; Igura, H.; Furuta, K.; Ito, H.; Wakabayashi, H.; Nakajima, K.; Mogami, T.; Horiuchi, T.; Yamashina, M.; “A 0.25-μm CMOS 0.9-V 100-MHz DSP core” Solid-State Circuits, IEEE Journal of , Volume: 32 , Issue: 1 , Jan. 1997 Pages:52 – 61

[2.11] Bum-Sik Kim; Dae-Hyun Chung; Lee-Sup Kim; “A new 4-2 adder and booth selector for low power MAC unit” Low Power Electronics and Design, 1997. Proceedings., 1997 International Symposium on , 18-20 Aug. 1997 Pages:100 – 103

[2.12] Parameswar, A.; Hara, H.; Sakurai, T.; “A swing restored pass-transistor logic-based multiply and accumulate circuit for multimedia applications” Solid-State Circuits, IEEE Journal of , Volume: 31 , Issue: 6 , June 1996 Pages:804 – 809

[2.13] Ivan Edward Sutherland, Robert F. Sproull, David Harris “Logical Effort: Designing Fast CMOS circuits”, Morgan Kaufmann; 1st edition (1999)

[2.14] Brodersen, R.W.; Horowitz, M.A.; Markovic, D.; Nikolic, B.; Stojanovic, V.; “Methods for true power minimization” Computer Aided Design, 2002. ICCAD 2002. IEEE/ACM International Conference on, 10-14 Nov. 2002 Pages:35 - 42

[2.15] J. Rabaey, A. Chandrakasan, and B. Nikolic, “Digital Integrated Circuits - A design Perspective “, 2/e Prentice Hall 2003

[2.16] Hamada, M.; Ootaguro, Y.; Kuroda, T.; “Utilizing surplus timing for power reduction” Custom Integrated Circuits, 2001, IEEE Conference on. , 6-9 May 2001 Pages:89 – 92

[2.17] Lackey, D.E.; Zuchowski, P.S.; Bednar, T.R.; Stout, D.W.; Gould, S.W.; Cohn, J.M.; “Managing power and performance for system-on-chip designs using Voltage Islands” Computer Aided Design, 2002. ICCAD 2002. IEEE/ACM International Conference on , 10-14 Nov. 2002 Pages:195 – 202

[2.18] Zyuban, V.; Strenski, P.; “Unified methodology for resolving power-performance tradeoffs at the microarchitectural and circuit levels” Low Power Electronics and Design, 2002. ISLPED '02. Proceedings of the 2002 International Symposium on , 12-14 Aug. 2002 Pages:166 – 171

[2.19] V. Stojanovic, D. Markovic, B. Nikolic, M. Horowitz, R. Brodersen, “Energy-Delay Tradeoffs in Combinational Logic using Gate Sizing and Supply Voltage Optimization,” to appear in Proc. ESSCIRC, Sept. 2002.

[2.20] Sakurai, T.; Newton, A.R.; “Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas” Solid-State Circuits, IEEE Journal of, Volume: 25 , Issue: 2 , April 1990 Pages:584 – 594

[2.21] Markovic, D.; Stojanovic, V.; Nikolic, B.; Horowitz, M.A.; Brodersen, R.W.; “Methods for True Energy-Performance Optimization” Solid-State Circuits, IEEE Journal of , Volume: 39 , Issue: 8 , Aug. 2004 Pages:1282 - 1293

References of Chapter 3

[3.1] C.S. Wallace, “A Suggestion for a Fast Multiplier,” IEEE Trans. Computers vol. 13, no.2, pp.14-17, Feb. 1964.

[3.2] Fried, R.; “Minimizing energy dissipation in high-speed multipliers” Low Power Electronics and Design, Proceedings., 1997 International Symposium on , 18-20 Aug. 1997 Pages:214 - 219

[3.3] Goto, G.; Inoue, A.; Ohe, R.; Kashiwakura, S.; Mitarai, S.; Tsuru, T.; Izawa, T.; “A 4.1-ns compact 54×54-b multiplier utilizing sign-select Booth encoders” Solid-State Circuits, IEEE Journal of , Volume: 32 , Issue: 11 , Nov. 1997 Pages:1676 - 1682

[3.4] Wen-Chang Yeh; Chein-Wei Jen; “High-speed Booth encoded parallel multiplier design” Computers, IEEE Transactions on , Volume: 49 , Issue: 7 , July 2000 Pages:692 – 701

[3.5] A.D. Booth, “A Signed binary multiplication technique,” Quarterly Journal of Mechanics and Applied Mathematics, pp. 236-240, June 1951.

[3.6] O.L. MacSorley, “High-Speed arithmetic in Binary Computers,” Proceedings of the IRE, vol.49, pp.67-91, Jan.1961

[3.7] Michael J.Flynn , “Advanced computer Arithmetic Design” , John Wiley & Sons, Inc. press, 2001.

[3.8] Behrooz Parhami , Computer Arithmetic Algorithms and Hardware Designs, Oxford Univ. Press, 1999.

[3.9] Itoh, N.; Naemura, Y.; Makino, H.; Nakase, Y.; Yoshihara, T.; Horiba, Y.; “A 600-MHz 54×54-bit multiplier with rectangular-styled Wallace tree” Solid-State Circuits, IEEE Journal of , Volume: 36 , Issue: 2 , Feb. 2001 Pages:249 - 257

[3.10] Dadda, L. 1965. “Some schemes for parallel multipliers.” Alta Frequenza , vol. 34, pp. 349–356.

[3.11] V. G. Oklobdzija, D. Villeger, and S. S. Liu, " A Method For Speed Optimized Partial Product Reduction And Generation Of Fast Parallel Multipliers Using An Algorithmic Approach", IEEE Transactions on Computers, Vol. 45, No.3 , March 1996.

[3.12] V. G. Oklobdzija, P. Stelling " Design Strategies for the Final Adder in a Parallel Multiplier", Twenty-Ninth Annual Asilomar Conference on signals, Systems and Computers, Pacific Grove, California, October 29 - November 1, 1995.

[3.13] B.R. Zeydel, V.G. Oklobdzija, S. Mathew, R.K. Krishnamurthy, S. Borkar, “A 90nm 1GHz 22mW 16x16-bit 2's Complement Multiplier for Wireless Baseband ”, Proceedings of the 2003 Symposium on VLSI Circuits, Kyoto, JAPAN, June 12 - 14, 2003.

[3.14] Behrooz Parhami , " Computer Arithmetic, Algorithms and Hardware Designs ", Oxford press, 2000.

[3.15] J. Sklansky, "Conditional-sum addition logic" IRE Transactions on Electronic Computers, pp. 226-230, June 1960.
[3.16] Lindkvist, H.; Andersson, P. "Techniques for fast CMOS-based conditional sum adders" Computer Design: VLSI in Computers and Processors, 1994. ICCD '94. Proceedings., IEEE International Conference on , 10-12 Oct. 1994 Pages:626 - 635
[3.17] Zhijun Huang, "High-Level Optimization Techniques for Low-Power Multiplier Design." Ph.D. dissertation, University of California Los Angeles, Aug. 2003.
[3.18] N.H.E. Weste, K. Eshraghian, Principles of CMOS VLSI Design. Addison-Wesley Publishing Company, 1993.
[3.19] W.-C. Yeh and C.-W. Jen, “High-speed Booth encoded parallel multiplier design,” IEEE Trans. Comput., vol.49, no.7, pp.692-701, July 2000.
[3.20] Jones, R.F., Jr.; Swartzlander, E.E., Jr., "Parallel counter implementation" Signals, Systems and Computers, 1992. 1992 Conference Record of The Twenty-Sixth Asilomar Conference on, 26-28 Oct. 1992 Pages:381 - 385 vol.1
[3.21] De Angel, E.; Swartzlander, E.E., Jr., "Low power parallel multipliers" VLSI Signal Processing, IX, 1996., Workshop on , 30 Oct.-1 Nov. 1996 Pages:199 - 208
[3.22] R. Brent and H. Kung, "A regular layout for parllel adders," IEEE Trans. Computers, Vol. C-31, No.3, pp260-264, March 1982.
[3.23] Jan M. Rabaey , et al., "Digital Integrated Circuits - A design Perspective ", 2/e Prentice Hall 2003
[3.24] P.M Kogge and H.S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations" , IEEE Trans. Computers, Vol. C-22, No. 8, pp.786-793, Aug. 1973.
[3.25] R.E. Ladner and M.J. Fischer, "Parallel prefix computation", Journal of ACM, Vol. 37, No.4, pp.831-838, Oct. 1980
[3.26] T. Han and D. Carlson, "Fast area-efficient VLSI adders," Proc. 8th Symp. Comp. Arith., pp.49-56, Sept. 1987.
[3.27] Kuo-Hsing Cheng; Shu-Min Chiang; Shun-Wen Cheng; “The improvement of conditional sum adder for low power applications”, ASIC Conference 1998. Proceedings. Eleventh Annual IEEE International , 13-16 Sept. 1998 Pages:131 - 134

References of Chapter 4

[4.1] Jyh-Ming Wang; Sung-Chuan Fang; Wu-Shiung Feng; ‘New efficient designs for XOR and XNOR functions on the transistor level” Solid-State Circuits, IEEE Journal of , Volume: 29 , Issue: 7 , July 1994 Pages:780 - 786

[4.2] Hanho Lee; Sobelman, G.E.; “New low-voltage circuits for XOR and XNOR” Southeastcon 97. Engineering New Century. , Proceedings. IEEE, 12-14 April 1997

[4.3] Hung Tien Bui; Yuke Wang; Yingtao Jiang; “Design and analysis of low-power 10-transistor full adders using novel XOR-XNOR gates” Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on, see also Circuits and Systems II: Express Briefs, IEEE Transactions on, Volume: 49, Issue: 1 , Jan. 2002

[4.4] Bum-Sik Kim, Dae-Hyun Chung, Lee-Sup Kim, “A new 4-2 adder and booth selector for low power MAC unit” Low Power Electronics and Design, 1997. Proceedings., 1997 International Symposium on, 18-20 Aug. 1997

[4.5] Shen-Fu Hsiao; Ming-Roun Jiang; Jia-Sien Yeh; “Design of high-speed low-power 3-2 counter and 4-2 compressor for fast multipliers” Electronics Letters , Volume: 34 , Issue: 4 , 19 Feb. 1998 Pages:341 - 343

[4.6] Margala, M.; Durdle, N.G.; “Low-power low-voltage 4-2 compressors for VLSI applications” Low-Power Design, 1999. Proceedings. IEEE Alessandro Volta Memorial Workshop on, 4-5 March 1999 Pages: 84 - 90

[4.7] Radhakrishnan, D.; Preethy, A.P.; "Low power CMOS pass logic 4-2 compressor for high-speed multiplication" Circuits and Systems, 2000. Proceedings of the 43rd IEEE Midwest Symposium on , Volume: 3 , 8-11 Aug. 2000 Pages:1296 - 1298 vol.3

[4.8] Prasad, K.; Parhi, K.K.; “Low-power 4-2 and 5-2 compressors” Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on, Volume: 1 , 4-7 Nov. 2001 Pages:129 - 133 vol.1

[4.9] Jiangmin Gu; Chip-Hong Chang; “Ultra low voltage, low power 4-2 compressor for high speed multiplications” Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on , Volume: 5 , 25-28 May 2003 Pages:V-321 - V-324 vol.5

[4.10] Jiangmin Gu; Chip-Hong Chang; “Low voltage, low power (5:2) compressor cell for fast arithmetic circuits” Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on, Volume: 2 , 6-10 April 2003 Pages:II - 661-4 vol.2

[4.11] Goel, S.; Elgamel, M.A.; Bayoumi, M.A.; “Novel design methodology for high-performance XOR-XNOR circuit design” Integrated Circuits and Systems Design, 2003. SBCCI 2003. Proceedings. 16th Symposium on , 8-11 Sept. 2003 Pages:71 – 76

[4.12] Radhakrishnan, D.; “Low-voltage low-power CMOS full adder” Circuits, Devices

and Systems, IEE Proceedings , Volume: 148 , Issue: 1 , Feb 2001 Pages:19 - 24
[4.13] Fang-Shi Lai, Wei Hwang, “Design and implementation of differential cascode voltage switch with pass-gate (DCVSPG) logic for high-performance digital systems” Solid-State Circuits, IEEE Journal of, Volume: 32 , Issue: 4 , April 1997 Pages:563 - 573
[4.14] Lawrence T. Clark, Rakesh Patel, Timothy S. Beatty, “Managing Standby and Active Moide Leakage Power in Deep Sub-micron Design,” ISLPED’04, Aug. 9-11, 2004.
[4.15] Clark L.T., Shay Demmons, Deutscher N., Ricci F., “Standby power management for a 0.18/spl mu/m microprocessor” ISLPED’02, Aug. 12-14, 2002.
[4.16] Roy K., Mukhopadhyay S., Mahmoodi-Meimand H., “Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits,” Proceedings of the IEEE , Volume: 91 , Issue: 2 , Feb. 2003 Pages:305 – 327
[4.17] Shigematsu S., Mutoh S., Matsuya Y., Yamada J., “A 1-V high-speed MTCMOS circuit scheme for power-down applications,” VLSI Circuits, 1995. Digest of Technical Papers., 1995 Symposium on , 8-10 June 1995 Pages:125 – 126
[4.18] Wei L., Chen Z., Roy K., Johnson M.C., Ye Y., De V.K., “Design and optimization of dual-threshold circuits for low-voltage low-power applications” Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , Volume: 7 , Issue: 1 , March 1999 Pages:16 – 24
[4.19] Kim C.H., Roy K., “Dynamic VTH scaling scheme for active leakage power reduction,” Design, Automation and Test in Europe Conference and Exhibition, 2002. Proceedings, 4-8 March 2002 Pages:163 - 167

References of Chapter 5

[5.1] Lars Wanhammar, “DSP Integrated Circuits,” San Diego, Calif.: Academic Press, 1999.

[5.2] Arthur Abnous, “Low-Power Domain-Specific Processors for Digital Signal Processing,” Ph.D. dissertation, University of California, Berkeley, 2001

[5.3] Sernec R., Zajc M., Tasic J., “The evolution of DSP architectures: towards parallelism exploitation,” Electrotechnical Conference, 2000. MELECON 2000. 10th Mediterranean, Volume: 2, 2000 Pages:782 - 785 vol.2

[5.4] Agarwala S., Anderson T., Hill A., Ales M.D., Damodaran R., Wiley P., Mullinnix S., Leach J., Lell A., Gill M., Rajagopal A., Chachad A., Agarwala M., Apostol J., Krishnan M., Duc Bui, Quang An, Nagaraj N.S., Wolf T., Elappuparackal T.T., “A 600-MHz VLIW DSP” Solid-State Circuits, IEEE Journal of, Volume: 37, Issue: 11, Nov. 2002 Pages:1532 – 1544

[5.5] “TMS320C6000 CPU and Instruction Set Reference Guide”, Texas Instruments, 2000

[5.6] Yuyun L., David R., Eric H., “VLSI Implementation of a High Performance and Low Power 32-bit Multiply-Accumulate Unit,” Proc., ESSCIRC, 2001.

[5.7] Seongmoo Heo, Krste Asanovic, “Power-Optimum Pipelining in Deep Submicron Technology,” ISLPED’04, Aug. 9-11, 2004.

[5.8] S. Nazareth, R. Asokan. “Processor Architecture for Multimedia”, academic paper, http://www.cs.dartmouth.edu/~nazareth/academic/CS107.pdf , November 2001.

[5.9] Perri, S.; Corsonello, P.; Iachino, M.A.; Lanuzza, M.; Cocorullo, G.; “Variable precision arithmetic circuits for FPGA-based multimedia processors”, Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , Volume: 12 , Issue: 9 , Sept. 2004 Pages:995 – 999

[5.10] Ying Li; Jie Chen; “A reconfigurable architecture of a high performance 32-bit MAC unit for embedded DSP”, ASIC, 2003. Proceedings. 5th International Conference on , Volume: 2 , 21-24 Oct. 2003 Pages:1285 - 1288 Vol.2

[5.11] Mokrian, P.; Ahmadi, M.; Jullien, G.; Miller, W.C.; “A reconfigurable digital multiplier architecture”, Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on , Volume: 1 , 4-7 May 2003 Pages:125 - 128 vol.1

[5.12] Jia Di; Yuan, J.S.; “Run-time reconfigurable power-aware pipelined signed array multiplier design”, Signals, Circuits and Systems, 2003. SCS 2003. International Symposium on , Volume: 2 , 10-11 July 2003 Pages:405 - 408 vol.2

[5.13] S. Kim C. H. Ziesler M. C. Papaefthymiou; “Fine-grain real-time reconfigurable pipelining”, IBM Journal of Research and Development archive Volume 47 , Issue 5-6 (September 2003) Pages: 599 – 609

[5.14] Sangjin Hong; Shu-Shin Chin; Connaway, C.;Variable-rate pipelined multiplier design for reconfigurable DSP applications Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on , Volume: 1 , 4-7 Aug. 2002 Pages:I - 587-90 vol.1

[5.15] Kim, S.; Papaefthymiou, M.C.; “Reconfigurable low energy multiplier for multimedia system design”, VLSI, 2000. Proceedings. IEEE Computer Society Workshop on , 27-28 April 2000 Pages:129 – 134

[5.16] Danysh, A.N.; Swartzlander, E.E., Jr.; “ A recursive fast multiplier ”, Signals, Systems & Computers, 1998. Conference Record of the Thirty-Second Asilomar Conference on , Volume: 1 , 1-4 Nov. 1998 Pages:197 - 201 vol.1

References of Chapter 6

[6.1] Ruby B. Lee, “Computer arithmetic – a processor architect’s perspective,” in Proc. 15th IEEE Symp. Computer Arithmetic, Keynote Presentationl, June 2001.

Vita

PERSONAL INFORMATION

Birth date: August 17, 1979

Birth place: Taipei, Taiwan, R.O.C.

Address: Department of Electronics Engineering
National Chiao Tung University
1001 Ta-Hsueh Road
Hsienchu, Taiwan 30050, R.O.C.

E-Mail Address: ppro.tw@yahoo.com.tw

Web Site: http://home.kimo.com.tw/ppro.tw/

EDUCATION

B.S. [2002] Department of Computer and Communication Engineering, National Kaohsiung First University of Science and Technology.

M.A. [2004] Institute of Electronics, National Chiao-Tung University.