A PIPELINE FFT PROCESSOR

Weidong Li
Electrical Engineering Dept.
Linköping University
Linköping
SE-581 83 Sweden

Lars Wanhammar
Electrical Engineering Dept.
Linköping University
Linköping
SE-581 83 Sweden

Abstract: In this paper, we discuss the design and implementation of a high-speed, low power 1024-point pipeline FFT processor. Key features are flexible internal data length and a novel processing element. The FFT processor, which is implemented in a standard 0.35 μm CMOS process, is efficient in term of power consumption and chip area.

1. INTRODUCTION

The Fast Fourier transform (FFT) and its inverse (IFFT) is one of the fundamental operations in the field of digital signal processing. The FFT/IFFT are widely used in various areas such as telecommunications, speech and image processing, medical electronics and seismic processing, etc. Recently, the FFT/IFFT is used as one of the key component in OFDM-based wideband communication systems, like xDSL modems and wireless mobile terminals.
In this paper we present the design of a 1024-point FFT/IFFT processor which computes a 1024-point FFT including I/O within 40 μs. The target application is a prototype for a wide coverage mobile radio modem with ten users with 2 Mbit/s each. This prototype is developed in a consortia consisting of three Swedish universities and three Ericsson companies. A brief review on FFT architectures is described in section 2. Implementation issues are discussed in section 3, and, finally, we present a performance estimate in section 4.

2. FFT ARCHITECTURE

FFT processors can be divided into three main classes:

I. Pipeline FFTs. They utilize concurrent processing of different stages to achieve high throughput [1].
II. Column FFTs. Each stage in the FFT is computed with a set of processing elements and the result is fed back to the same processing elements for the computation of the next stage.
III. Fully parallel FFTs. The operations in the signal-flow graph are mapped isomorphically to a hardware structure. The implementations are hardware intensive and is with current technology not practical for large FFTs.

0-7803-5650-0/99/$10.00 © 1999 IEEE

2.1 Radix-4 Single-path Delay Commutator Architecture

In order to minimize the power consumption, a trade-off between the complexity of the butterfly processing elements and the data communication between processing elements should be done. A large radix, i.e. complex processing element, results in fewer processing elements and fewer transactions between process elements and memories in each stage and the number of stages is also reduced. Furthermore, the multiplications are the most power dissipating operations as observed from the previous work by J. Melander [2] and T. Widhe [3]. The choice on architecture with fewer multiplications is therefore important to reduce the total power consumption. Architectures with radix-4 is in this respect superior to architectures with radix-2.

In our target application, the input data arrive continuously. Hence, the FFT processor must buffer the input and output data in order to obtain a uniform throughput. We have therefore selected a Radix-4 Single-path Delay Commutator (R4SDC) architecture. This architecture was first proposed by Bi and Jones [4] and has been implemented in [5]. This architecture improves the efficiency of butterfly element with a modified butterfly element. The architecture of a 16-point R4SDC is shown in Figure 1, and the memory size for the first and second stage is 24 and 6, respectively.

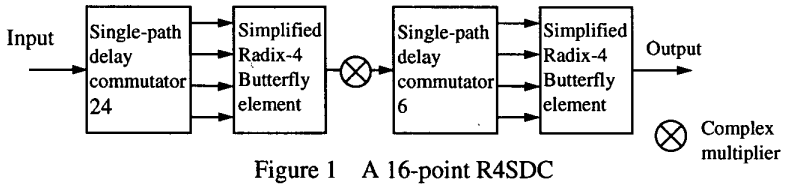

Figure 1 A 16-point R4SDC

2.2 Simplified Butterfly Element

The conventional Radix-4 butterfly element requires 8 adders/subtractors and some trivial multiplications with ± 1 or $\pm j$. Since the input and output data rate is uniform, the butterfly element can be simplified. The simplified butterfly element requires only 3 adders/subtractors and processes one output data at a time to match the input data rate. However, this requires more memory to temporarily save the input data.

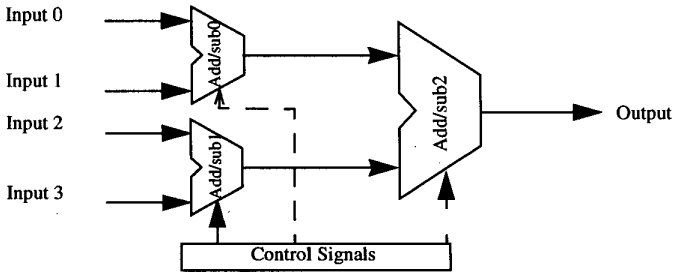

Figure 2 Simplified Butterfly Element

2.3 Complex Multiplier

The complex multiplier is the key component in the datapath. The direct implementation of complex multiplier requires 4 real multipliers. the number of real multipliers can be reduced to 3 with a simple transformation at the cost of extra additions. In the FFT processor, the twiddle factors are known in advance, which can be simplified the complex multiplier with Distributed Arithmetic (DA) [12]. The hardware complexity for the complex multiplier with DA in our design is about 2 real multipliers with a small modification in the partial product generation.

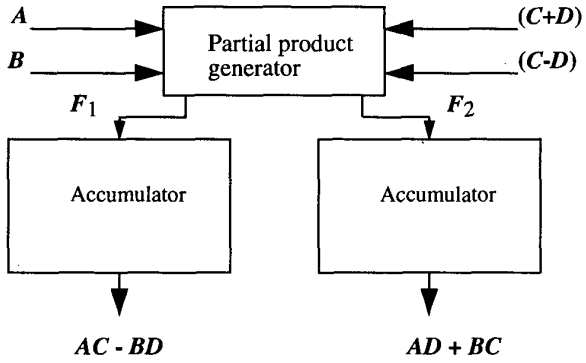

Figure 3 Complex Multiplier Using DA

A complex multiplier using DA is shown in Figure 3, where C and D are real and imaginary part of the twiddle factor. Table 1 shows the partial products, where $\mathbf{a}_i$ and $\mathbf{b}_i$ are the i-bit of multiplicands A and B. §

$\mathbf{a}_i$	$\mathbf{b}_i$	F_1	F_2
0	0	- (C - D)	- (C + D)
0	1	- (C - D)	(C - D)
1	0	(C + D)	- (C - D)
1	1	(C - D)	(C + D)

Table 1: Partial Product Generation for Complex Multiplier

2.4 Memory

The memories dominate in terms of chip area. It is important both from power consumption and chip area point of view to reduce the memory size. This can be done in two ways: the first is to minimize the internal word length, and hence, reduce the memory size; the second is to select an area efficient memory structure. The in-

ternal word length reduction shows that memory size can be reduced up to 14% by using different word lengths in the stages [6]. A dual-port memory is required since the read and write operation must be performed in one clock cycle.

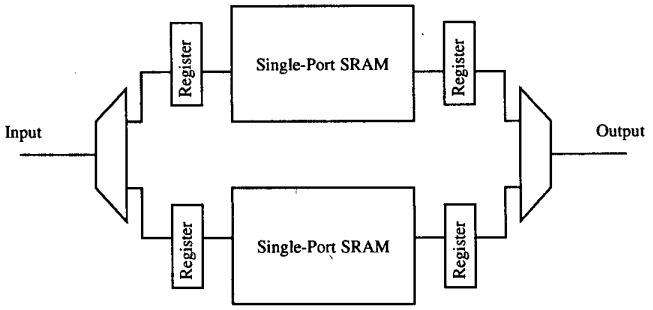

Figure 4 Single-port RAM Based Commutator

However, a dual-port memory cell requires 33% more chip area than a single-port memory cell. We therefore use two single-port memories to replace a dual-port memory [15] as illustrated in Figure 4.

3. IMPLEMENTATION CONSIDERATIONS

In order to obtain low power consumption with a standard 0.35 μm CMOS process which is normally operated at 3.3 V, we operate the FFT processor at a lower than normal power supply voltage [7], i.e. only 1.5 V.

The pipelined FFT processor can be divided into three main building blocks: memory, simplified butterfly elements, and complex multipliers. The control unit is also considered in this section.

3.1 Memory

As the supply voltage is as low as 1.5 V, the speed degration will be severe for the memories design. Since the SRAM has a lower access time than DRAM at low supply voltage, we select to use SRAM. An SRAM consists of three building blocks: memory cell matrix, peripheral circuits, and the decoder.

The decoder can be realized by using a hierarchical architecture, which reduce both the delay and the activity factor. The row decoder can use either NOR-NAND decoder or tree decoder. Tree decoder requires fewer transistors, but suffer from speed degration due to the serial-connection of pass-transistors, which could increase the delay (it becomes worse for lower power supply voltage). The NOR-NAND decoder has a regular layout but requires more transistors. In small decoders the tree decoder is preferred and the NOR-NAND decoders is preferred for larger decoders.

The memory cell is a 6T full CMOS memory cell because it is more robust at

1.5V. To reduce the word line delay we use metal wires instead of poly wires.

The sense amplifiers (SA) are one of the most power dissipating parts in the peripheral circuits. To reduce the power consumption, pulsed sense enable signal is used to reduce the active time. The conventional current mirror sense amplifier has both low gain and large delay time for low voltage operation. We modified therefore an STC D-flip-flop [8] to form a two stage latch type sense amplifier. The sense amplifier is functional when the supply voltage is as low as 0.9 V. The *Hspice*™ simulation shows that this SA dissipates 11.4 μW at 1.5 V, 25 MHz.

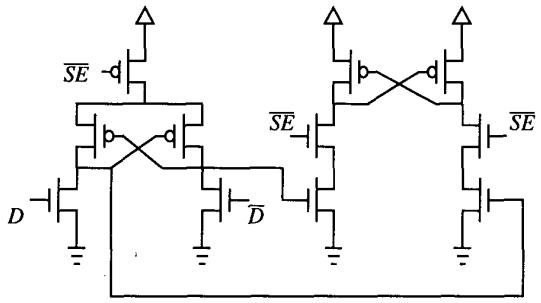

Figure 5 STC D-flip-flop

3.2 Complex Multiplier

The complex multipliers dissipates almost 80% of total power in the previous work [2] [3]. So it is important to design a low power multiplier.

Multipliers can be divided into three types: bit-parallel, bit-serial, and digit-serial. Although the bit-serial, or digit-serial, multiplier has often less chip area than that of bit-parallel, it requires a high-speed clock. To achieve high throughput, the bit-serial, or digit-serial multiplier often needs several parallel units, which increases activity factor for the local clock. To meet the speed requirement, we therefore select a bit-parallel structure.

The fastest multi-operand tree is the Wallace tree, but the Wallace tree has complex wiring and is therefore difficult to optimize and the layout becomes irregular. The overturned-stairs tree [9], which has a regular layout and the same performance as the Wallace tree when the data word length is less than 19, is used in the design of the complex multipliers [10].

The straight forward way to generate partial products is derived directly from the formulation of DA [12]. It requires a delay of two XOR-gates and a 2-to-1 multiplexer, which is shown in Figure 6(i). A new scheme (See Figure 6(ii)) is suggested in [13] which uses only a 4-to-1 multiplexer and two inverters. As the final adder, a

Brent-Kung's adder [14] is used in order to meet the timing constraint.

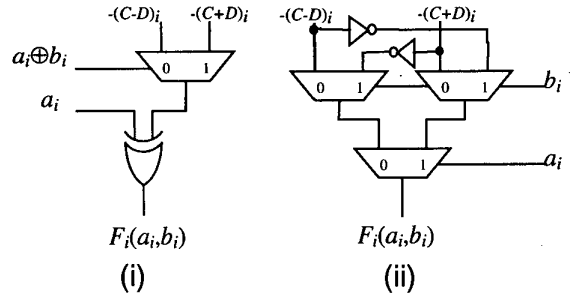

Figure 6 Partial Product Generation Circuits

The selection of full adder is important to the whole multiplier. The conventional static CMOS adder with large stack height, is too slow (See Figure 7). We therefore select an adder called Reusens full adder [11](Figure 8). This adder is fast and compact, but requires buffers at the outputs.

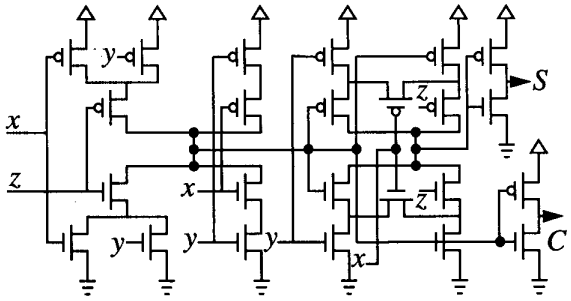

Figure 7 Conventional CMOS Full Adder

The buffer insertion is usually considered as a drawback since it introduces extra delay and increases power consumption. However, in the multiplier design, the buff-

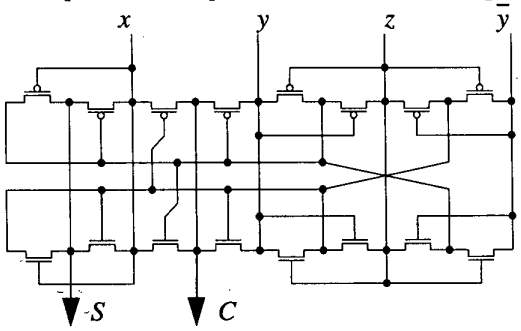

Figure 8 Reusens Full Adder

er insertion is necessary any way to drive the long interconnections. There is no direct path from power supply line to the ground in this full adder, which yields low power consumption. The *Hspice*™ simulation shows that the complex multiplier can operates up to 30 MHz at 1.5 V. The power consumption, which is estimated with *Lsim Power*™, is 15 mW at 25 MHz with 1.5 V power supply voltage.

3.3 Simplified Butterfly Element

The simplified butterfly element utilizes three adders/subtractors and need three control signals (See Figure 2). The processing of data takes place in two steps: the first step is to add/subtract two input data to generate the intermediate results, the second step is to use the intermediate results to compute the output. Note that the only operations used are additions and subtractions. Hence we can simplify the processing as shown in Figure 9 where we have replaced the two adders/subtractors in the butterfly element with a 4/2 compressor and two extra set of XOR-gates.

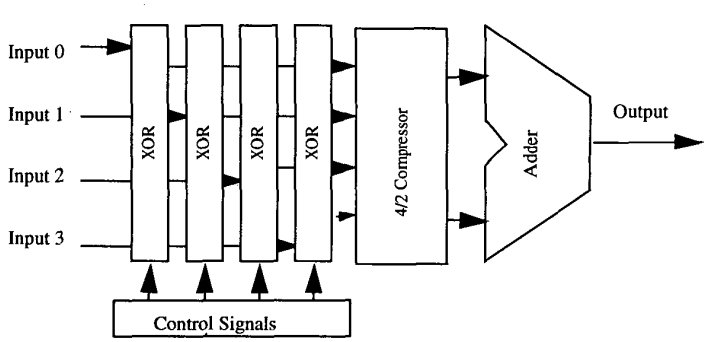

Figure 9 Novel Simplified Butterfly Element

3.4 Control Unit

A global control unit that provides the control signals to the different parts of FFT processor is expensive in term of complexity and it is difficult to manage the timing issues. As the transistor size is scaled down and clock frequencies increase, the problem of routing and associated timing issues become even more difficult. We therefore select a distributed control scheme [15]. The 1024-point FFT processor has five stages and each stage has a local control unit. Only three control signals are transferred between two local control units belonging to adjacent stages. This approach reduces the clock skew between datapath and control units, since they can share the same local clock driver. Local control units has simpler control function and, hence, are easier to design.

4. RESULTS

The FFT processor has been designed using a 0.35 μm standard CMOS technology with five metal layer from *Alcatel Mietec*™. The FFT processor has a core area of 3.1 mm × 3.4 mm and dissipates less than 200 mW at 1.5 V.

	Power Consumption @1.5 V, 25 MHz
Memories (RAMs, ROMs)	90 mW
Complex Multipliers	60 mW
Simplified Butterfly Elements	30 mW
Control unit & others	20 mW

Table 2: Estimated power consumption

As we can see from the table, the percentage of power consumption for complex multipliers is efficiently reduced in our design. The most power dissipating part in the FFT processor is memories, which dissipate 45% in the total power consumption. Low power design of memories should be important in the future works.

ACKNOWLEDGMENTS

The authors would like to thank Thomas Johansson and Dr. Mark Vesterbacka for fruitful discussions. This project is financed by SSF, the Foundation for Strategic Research in Sweden, under the program of INTELECT.

References

[1] S. He, *Concurrent VLSI Architectures for DFT Computing and Algorithms for Multi-Output Logic Decomposition*, Diss. No. 133, Lund University, Sweden, 1995.

[2] J. Melander, *Design of SIC FFT Architectures*, Linköping Studies in Science and Technology, Thesis No. 618, Linköping University, Sweden, 1997.

[3] T. Widhe, *Efficient Implementation of FFT Processing Elements*, Linköping Studies in Science and Technology, Thesis No. 619, Linköping University, Sweden, 1997.

[4] G. Bi and E. V. Jones, "A Pipelined FFT Processor for Word-Sequential Data," *IEEE Trans. on Acoustic, Speech, and Signal Process.*, vol. ASSP-37, No.12, pp. 1982-1985, Dec. 1989.

[5] E. Bidet, D. Castelain, C. Joanblanq, and P. Stenn, "A Fast Single-chip Implementation of 8192 Complex Point FFT," *IEEE JSSC.*, Vol. C-30, no. 3, pp. 300-305, Mar. 1995.

[6] W. Li, Y. Ma, and L. Wanhammar, "Word Length Estimation for Memory Efficient Pipeline FFT/IFFT Processors," to appear at the Inter. Conference on Signal Processing Applications & Technology (ICSPAT), Nov., 1999.

[7] A. P. Chandraksan, S. Sheng, and R. W. Brodersen, "Low-Power CMOS Digital Design," *IEEE JSSC.*, Vol. c27, pp. 473-483, 1992.

[8] J. Yuan, *High Speed CMOS Circuit Technique*, Linköping Studies in Science and Technology, Thesis No. 132, Linköping University, Sweden, 1988.

[9] Z. Mou and F. Jutand, "'Overturned-Stairs' Adder Trees and Multiplier Design," *IEEE Trans. on Computer*, vol. C-41, No. 8, pp. 940-948, Aug. 1992.

[10] W. Li and L. Wanhammmar, "A Complex Multiplier Using "Overturned-Stairs" Adder Tree," to appear at IEEE Inter. Conference on Electronics, Circuits and Systems (ICECS), Sept., 1999.

[11] P. P. Reusens, *High Performance VLSI Digital Signal Processing Architecture and Chip Design*, Cornell University, Thesis, Aug. 1983.

[12] L. Wanhammar, *DSP Integrated Circuits*, Academic Press, 1999.

[13] T. Johansson, *Design of a Complex Multiplier*, Linköping University, to be published.

[14] R. Brent and H.T. Kung, "A Regular Layout for Parallel Adders," *IEEE Trans. on Computer*, vol. C-31, No. 3, pp. 260-264, March, 1982.

[15] W. Li, *VHDL Modelling of a 1024-point Pipeline FFT Processor*, Master's Thesis, LiTH-ISY-EX-1839, Linköping University, Sweden, 1997.