行政院國家科學委員會專題研究計畫 期中進度報告

使用 60GHz 之室內十億級位元傳輸率之無線基頻傳收機--子計畫三：針對通訊數位訊號處理器之電子系統層級驗証與合成環境(2/3)

期中進度報告(完整版)

計 畫 類 別 ： 整合型
計 畫 編 號 ： NSC 98-2220-E-009-028-
執 行 期 間 ： 98 年 08 月 01 日至 99 年 07 月 31 日
執 行 單 位 ： 國立交通大學電子工程學系及電子研究所

計 畫 主 持 人 ： 周景揚
共 同 主 持 人 ： 黃俊達

處 理 方 式 ： 本計畫可公開查詢

中 華 民 國 99 年 05 月 28 日

針對通訊數位訊號處理器之電子系統層級驗證與合成環境(2/3)

An ESL system verification and synthesis environment for communication DSP (2/3)

計畫編號: NSC98-2220-E-009-039

執行期間: 98 年 8 月 1 日 至 99 年 7 月 31 日

主持人：周景揚 交通大學電子工程系教授

共同主持人：黃俊達 交通大學電子工程系副教授

一、 中文摘要

快速傅利葉轉換處理器相當廣泛的應用在訊號處理系統及通訊系統中。雖然現存的文獻提供了許多快速傅利葉轉換處理器的架構，但要能夠在給定的條件下挑選出最適合的架構仍是一個相當重要的技術問題。一個快速傅利葉轉換處理器產生器，不但可以增加設計的生產力，同時也可以縮短整個系統設計開發的時程。在這篇論文中，我們針對管線化的快速傅利葉轉換架構提出了面積與通量折衷的方法，且能自動地產生對應的硬體設計。實驗結果顯示，我們在通量的限制之下，可以產生硬體面積較小的架構。

關鍵字

快速傅利葉轉換、管線化、參數化、產生器

英文摘要

The Fast Fourier Transform (FFT) processors are widely used in signal processing systems and communication systems. Many FFT architectures are proposed in literature to meet different applications. While designing an FFT processor, one of the most difficult issues is to choose the best architecture under the design constraints. An FFT generator can not only improve the productivity but also shorten time-to-market. In this thesis, we propose approaches which can make appropriate design trade-off between throughput and area of pipeline FFT architectures, and automatically generate the corresponding hardware design. The experimental results show that the proposed methodology can generate area-efficient architectures under throughput constraints.

Keywords

Fast Fourier Transform, pipeline, parameterization, generator

二、 計畫的緣由與自的

A. Introduction

Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) are widely used algorithms for calculating the Discrete Fourier Transform (DFT) and Inverse Discrete Fourier Transform (IDFT) because of the low computation complexity. FFT processor is an important block in communication system and signal processing system. For example, as shown in Figure 1, Orthogonal Frequency Division Multiplexing (OFDM) system is widely used in many communication applications such as xDSL modem, HDTV, and wide band mobile terminals. In those applications, FFT and IFFT are the most important processing blocks to meet the design constraints.

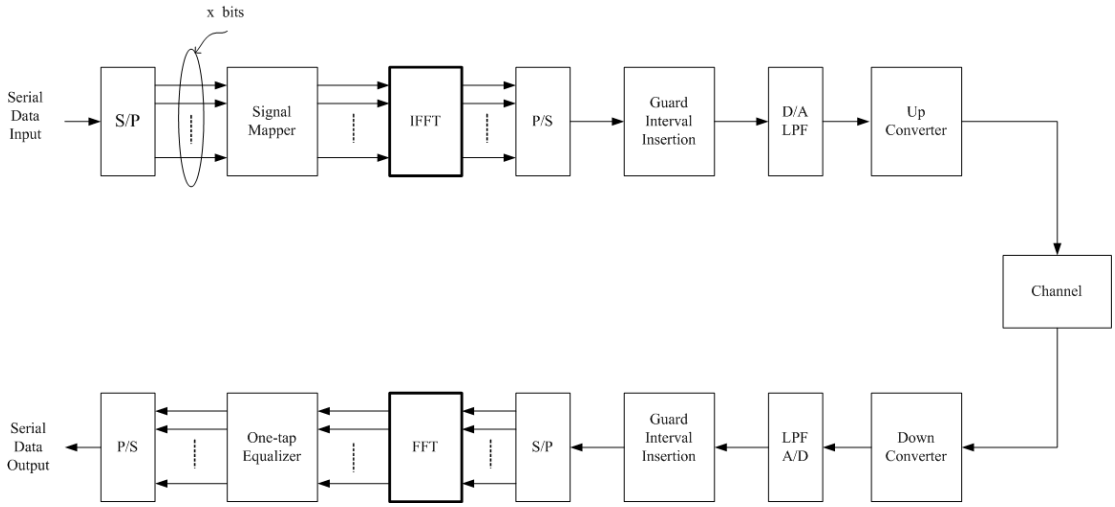

Figure 1 An example architecture of OFDM system

An automatic FFT generator can not only improve productivity but also shorten time-to-market. To support user customization, the automatic FFT generator provides some parameters to customize for the design constraints, such as the FFT transform sizes, I/O data ordering, data bitwidth, and the various architectures. In this thesis, we mainly focus on the trade-off between throughput and area of the FFT architectures.

Since the FFT algorithm was proposed by Cooley and Turkey in 1965 [1], many similar algorithms have been proposed to reduce the computation complexity of FFT. As the technology progress and algorithm improvement, FFT is widely used in Digital Signal Processing (DSP) applications. According to different algorithms, many kinds of FFT architectures have been proposed. Generally, there are two kinds of popular FFT architectures. One is memory-based architecture and the other is pipelined-based architecture. A single processing elements (PE) is used in memory-based architecture. It can be easily extended to other FFT transform sizes, so the memory-based architectures are usually used for low hardware cost and low throughput designs. Pipeline-based architectures have features such as regularity, simplicity, and high throughput rate. In this thesis, we only focus on pipelined-based architectures.

Several pipeline-based FFT architectures are proposed, such as the Radix-2 Multi-path Delay Commutator (R2MDC) [2], Radix-4 Multi-path Delay Commutator (R4MDC) [2], Radix-2 Single-path Delay Feedback (R2SDF) [3], Radix-4 Single-path Delay Feedback (R4SDF) [4], Radix-2^2 Single-path Delay Feedback (R2^2SDF) [5], Radix-2^2 Multi-path Delay Commutator (R2^2MDC) [6] and Radix-2^3 Single-path Delay Feedback (R2^3SDF) [7]. In these architectures, the R4SDF requires fewer multipliers than the R2SDF; however the R2SDF architecture is simpler and more regular than the R4SDF. The R2MDC requires fewer multipliers, adders and memory size than the R4MDC; however, the R4MDC can provide higher throughput. The R2^2SDF has the same multiplier complex as R4SDF, but retains the butterfly structure of radix-2 algorithm. The R2^2MDC uses the same algorithm as the R2^2SDF, and the R2^2MDC has higher throughput. As a result, in this work, our proposed FFT generator is based on the R2^2MDC and the R2MDC architectures.

B. Pipeline-based FFT architectures

Pipeline-based architectures can be further divided into two kinds of architectures depend on the design of register. One is Single-path Delay Feedback (SDF) architecture, and the other one is Multi-path Delay Commutator (MDC) architecture. SDF architecture has higher hardware usage and lower hardware cost; however, MDC architecture has higher throughput than SDF architecture. We introduce these architectures below.

We first introduce the SDF architecture. The Radix-2 SDF (R2SDF) architecture [3] is shown in Figure 2. By storing the butterfly output into the shift registers, R2SDF uses the registers efficiently. The butterfly passes the output to the next stage when doing addition operation and storing the output into the shift register when doing subtraction operation. In each cycle, only one output passes through the multiplier.

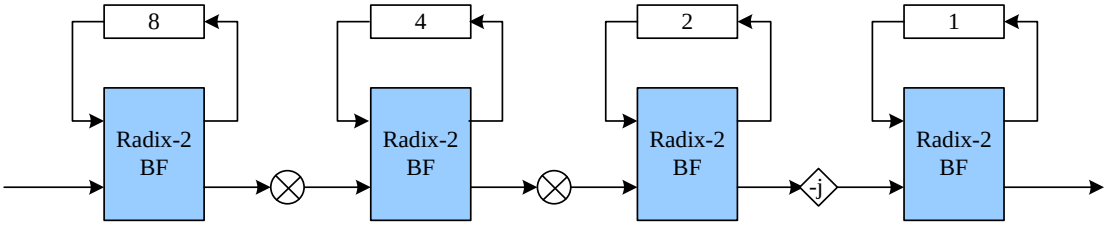

Figure 2 R2SDF architecture (N=16)

The Radix-4 SDF (R4SDF) architecture [4] is shown in Figure 3. Similar to the R2SDF architecture, radix-4 butterfly store three of outputs into shift registers, and only one output passes through the multiplier in each cycle.

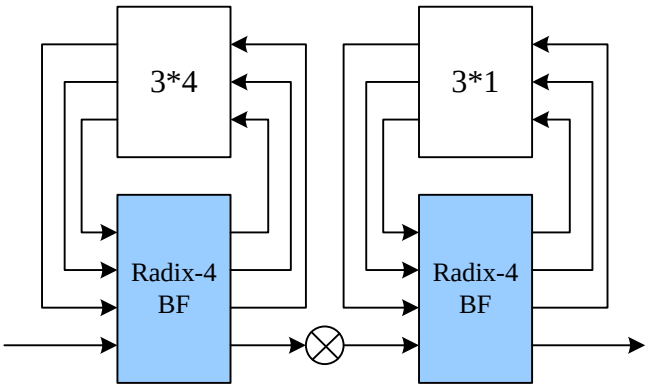

Figure 3 R4SDF architecture (N=16)

The Radix-2^2 SDF (R2^2SDF) [5] architecture is similar to the R2SDF architecture and reduces the number of multipliers. R2^2SDF uses two types of butterflies, one is the same as that

in R2SDF architecture and the other contains also some logic to implement the multiplication of twiddle factor of –j, as shown in Figure 4.

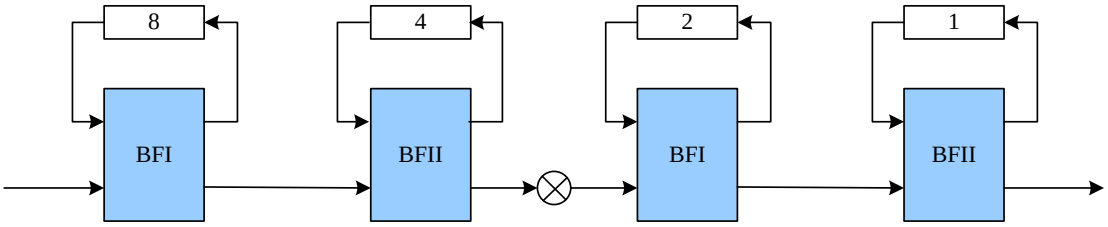

Figure 4 R2^{2}SDF architecture (N=16)

The Radix-2 MDC (R2MDC) architecture [2] is straightforward. The inputs are separated into two streams by the control of switches, and then go to butterflies in parallel, as shown in Figure 5.

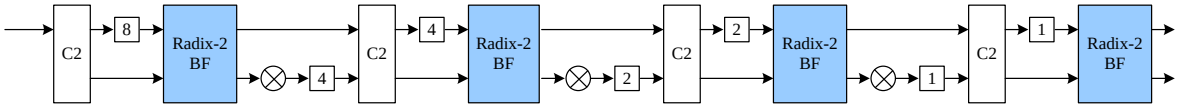

Figure 5 R2MDC architecture (N=16)

The Radix-4 MDC (R4MDC) architecture [2] is also similar to R2MDC architecture besides the raidx-4 butterfly and the number of registers, as shown in Figure 6.

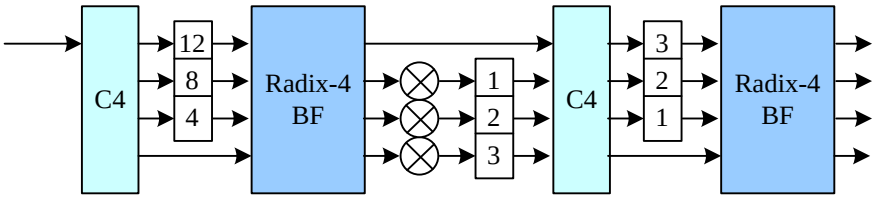

Figure 6 R4MDC Architecture (N=16)

The Radix-2^2 MDC (R2^{2}MDC) architecture [6] is the MDC type architecture of Radix-2^2 algorithm. In the flow graph of the complete decomposition of an N-point FFT computation with radix-2^2 algorithm, the even-numbered stages multiple twiddle factors not only the subtraction output but the addition output, so the R2^{2}MDC architecture needs two complex multipliers in even-numbered stages, as shown in Figure 7.

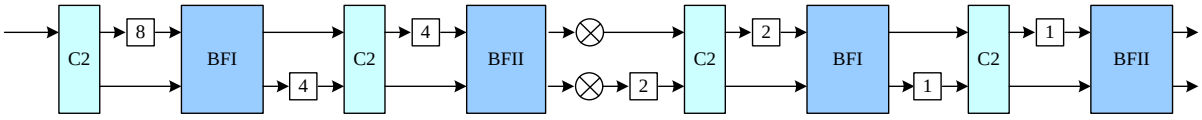

Figure 7 R2^{2}MDC architecture (N=16)

三、 研究方法及成果

A. Motivation

An exhaustive search approach is proposed to find all possible FFT architectures and then generate a set of acceptable FFT architecture according to the design constraints. However, from Table 1, we can find that all the possible solutions have the same number of multipliers, number of adders and number of registers usage under the throughput constraint.

Pease architecture bases on the radix-2 algorithm. Observing the raidx-2 flow graph, each butterfly is followed by a multiplication operation at the output of subtraction operation. Therefore, Pease architecture is a very regular architecture. However, the radix-2 algorithm contains many trivial multiplications which do not need multipliers to calculate. For example, multiplication of $-j$ involves only real-imaginary swapping and sign inversion, as shown in Figure 8.

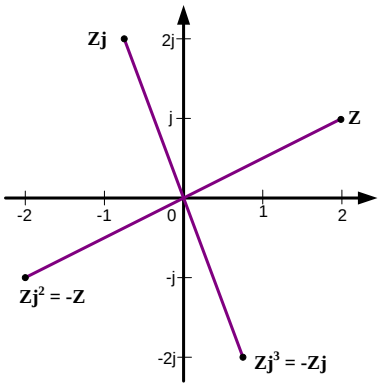

Figure 8 Illustration of $-j$ multiplication

The radix-2^2 algorithm considers the multiplication of $-j$ and merges the multiplication of $-j$ into odd-numbered columns. And the architecture of radix-2^2 algorithm contains two kinds of butterflies, BFI and BFII. From the view of architecture, the radix-2^2 algorithm is more irregular than the radix-2 algorithm.

The R2^2MDC [6] is a pipeline architecture that implements the radix-2^2 algorithm with throughput $\frac{2}{N}$, so R2^2MDC architecture is more irregular than Pease architecture. In the following subsections, we introduce how we make the trade-off between hardware and throughout based on R2^2MDC and R2MDC architecture.

B. R2²MDC Vertical Expansion Architecture

A general form of R2²MDC vertical expansion architecture is shown in Figure 9. Parameter ***N*** indicates the FFT transform size, where $N = 2^m, m = 1, 2, 3...$. Parameter ***t*** indicates the degree of parallelism, where $t = 1, 2, 4..., 2^{m-1}$. The number of registers of each original R2²MDC architecture decreases as the degree of parallelism increases, and the number of interconnection permutation matrix also increases. With the interconnection permutation matrix, data dependence would be kept. From Figure , we can derive the number of multipliers is $t(2\lceil \log_4 N \rceil - 2)$, the number of adders is $2t \log_2 N$, the number of registers is $N - 2t$ and the throughput is $\frac{2t}{N}$.

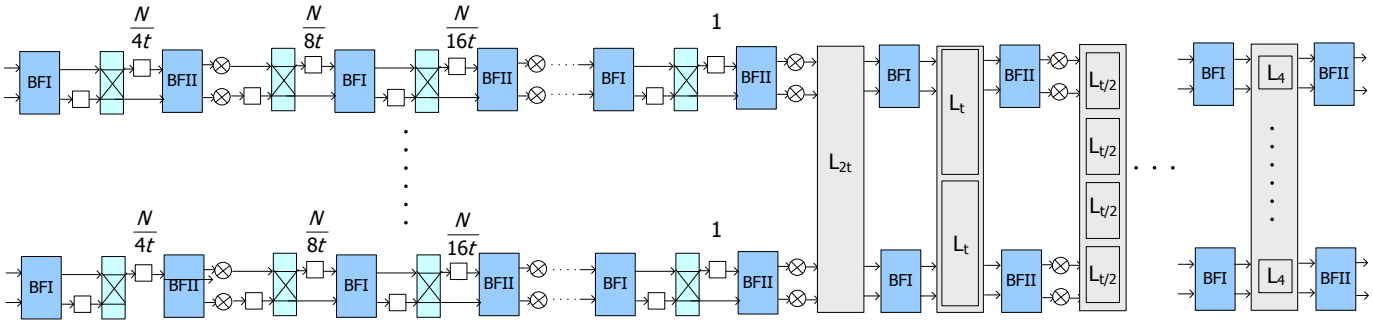

Figure 9 General form of R2²MDC vertical expansion architecture

Figure 10 shows the case when ***t*** =1, the original R2²MDC architecture, the number of multiplier is 2, the number of adders is 8, the number of registers of datapath is 14, and the throughput is $\frac{1}{8}$.

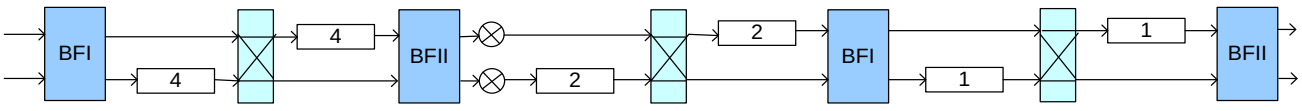

Figure 10 Example of R2²MDC vertical expansion architecture for t=1

Figure 11 shows the case when ***t*** =2, the number of multipliers is 4, the number of adders is 16, the number of registers of datapath is 12, and the throughput is $\frac{1}{4}$.

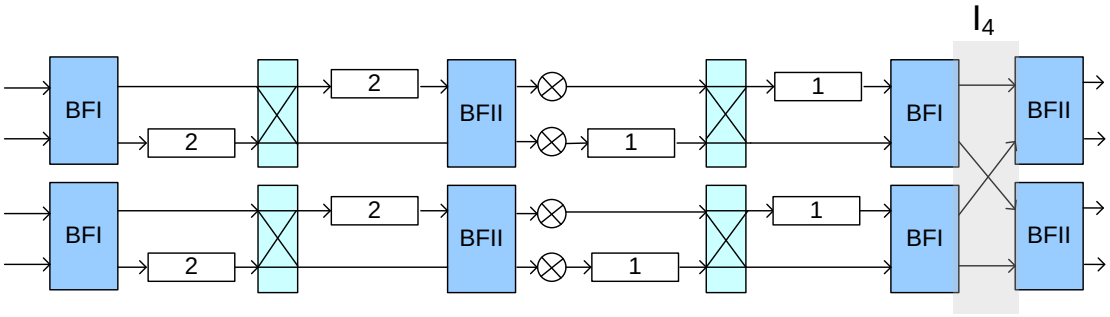

Figure 11 Example of R2^{2}MDC vertical expansion architecture for t=2

Figure 12 shows the case when ***t*** =4, the number of multipliers is 8, the number of adders is 32, the number of registers of datapath is 8, and the throughput is $\frac{1}{2}$.

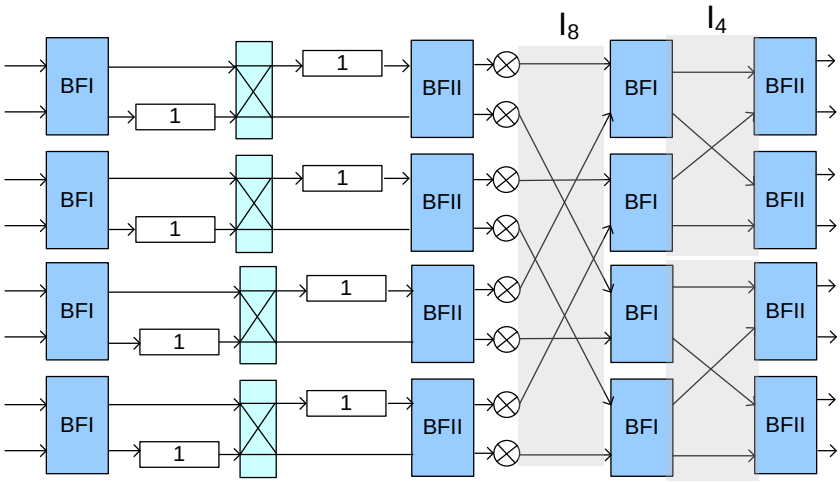

Figure 12 Example of R2^{2}MDC vertical expansion architecture for t=4

Figure 13 shows the case when ***t*** =8, namely, a fully parallelized R2^{2}MDC vertical expansion architecture, the number of multipliers is 16, the number of adders is 64, the number of registers of datapath is 0, and the throughput is $\frac{1}{2}$.

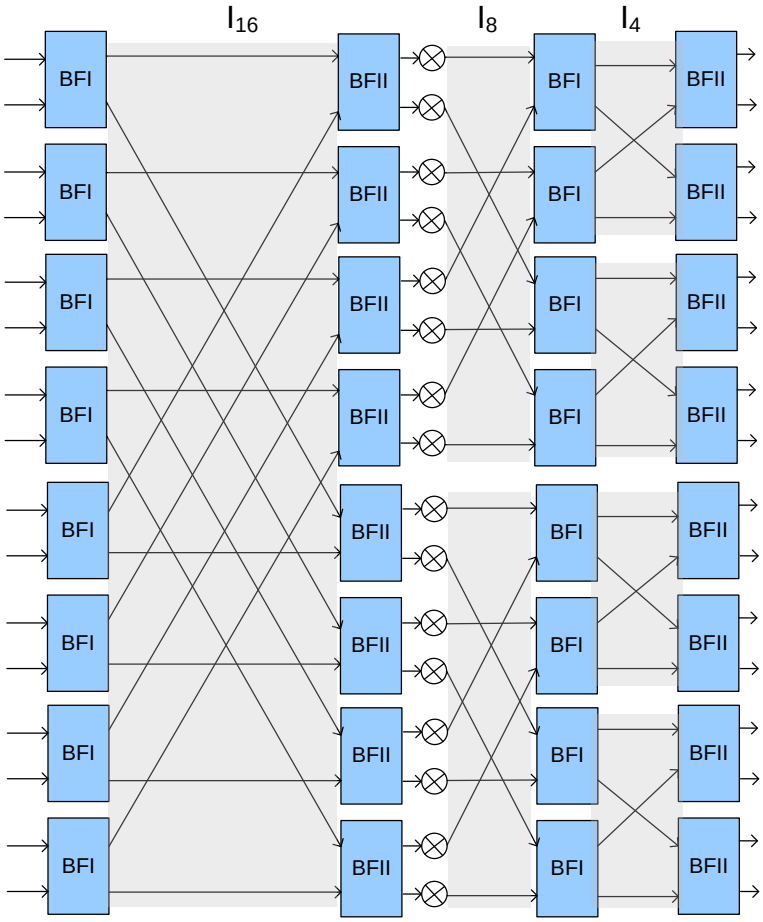

Figure 13 Example of R2^{2}MDC vertical expansion architecture for t=8

C. Summary

In this project, we proposed two directional trades-off approaches based on R2^{2}MDC architecture and R2MDC architecture. In vertical direction, we provide an expansion approach for R2^{2}MDC architecture to increase the throughput. Under the throughput constraint, our approach can provide only one exact solution. Table 1 lists the hardware and throughput comparison between our approach and previous work, where R2^{2}EMDC indicates the vertically expanded R22MDC architecture. Table 2 gives the normalized hardware and throughput comparison with the same throughput by replacing jk with $t \log_2 N$.

Table 1 Hardware Requirement Comparison

FFT length (N)	multipliers	adders	registers	throughput
Pease	jk	2jk	N	$\frac{2jk}{N\log_2 N}$
R2^{2}EMDC	$t(2\lceil \log_4 N \rceil - 2)$	$2t\log_2 N$	N-2t	$\frac{2t}{N}$

Table 2 Hardware Requirement Comparison with the same throughput

FFT length (N)	multipliers	adders	registers	throughput
Pease	$t\log_2 N$	$2t\log_2 N$	N	$\frac{2t}{N}$
R2^{2}EMDC	$t(2\lceil \log_4 N \rceil - 2)$	$2t\log_2 N$	N-2t	$\frac{2t}{N}$

四、 結論與討論

The FFT processor is an important computing block in communication and signal processing systems. To improve productivity and shorten time-to-market, an automatic FFT generator can be used to design a specified FFT processor. In this thesis, we propose a parameterizable FFT generator with two approaches to make good design trade-off between throughput and area under the design constraints. First, the vertical expansion approach parallels the datapath to increase the throughput. Second, the horizontal compression approach folds the datapath to reduce the hardware usage. Besides, only the best FFT architecture is generated under the user-specified throughput constraint to reduce the computation time in our proposed FFT generator. Compared with the Pease architecture, for the length of 256 and 1024 cases, the generated FFT processor saves about 30.8% area under throughput constraints.

Various FFT architectures are proposed in literature. It can be implemented into our proposed FFT generator. In the future, more FFT algorithms such as the $R2^3$MDC FFT algorithm, mixed-radix FFT [17] algorithm will be considered to enlarge the search space. Besides, the bitwidth optimization techniques proposed in [18] will also be considered.

五、 參考文獻

[1] H. Keding, M. Willems, M. Coors, and H. Meyr, "FRIDGE: A fixed-point design and simulation environment," in *Proceedings of IEEE Design, Automation and Test in Europe (DATE '98)*, pp. 429–435, Paris, France, Feb. 1998.

[2] G. A. Constantinides, G. J. Woeginger, "The complexity of multiple wordlength assignment," *Applied Mathematics Letter,* pp. 137–140, 2002.

[3] Synopsys Inc. (2001). *CoCentric SystemC Compiler Behavioral Modeling Guide.*[online]. Avaible:www.synopsys.com

[4] Synopsys Inc. (2000). *Synopsys CoCentric Fixed-Point Designer Datasheet* [online]. Avaible:www.synopsys.com

[5] C.Fang, R. Rutenbar, T. Chen, "Fast, accurate static analysis for fixed-point finite-precision effects in DSP designs," *2003 International Conference on Computer-Aided Design, ICCAD-2003.*, vol., no., pp. 275-282, Nov. 2003

[6] D. Lee, A. Abdul Gaffar, R. Cheung, O. Mencer, W. Luk, and G. Constantinides, "Accuracy-guaranteed bit-width optimization," *IEEE Trans. Computer-Aided Design*

Integrated Circuits and Systems, vol. 25, no. 10, pp. 1990-2000, Oct. 2006.

[7] W. Osborne, R.C.C. Cheung, J. Coutinho, and W. Luk, "Automatic accuracy guaranteed bit-width optimization for fixed and floating-point systems," in *IEEE International Conference on Field-Programmable Logic and Applications (FPL)*, Netherlands, Aug. 2007.

[8] C.Fang, R. Rutenbar, Puschel M. and T. Chen, "Toward efficient static analysis of finite-precision effects in DSP applications via affine arithmetic modeling," *Design Automation Conference, 2003. Proceedings* , vol., no., pp. 496-501, Jun. 2003

[9] H. Choi and W. P. Burleson, "Search-based wordlength optimization for VLSI/DSP synthesis," in *Proceedings of IEEE Workshop on VLSI Signal Processing, VII*, pp. 198–207, La Jolla, Calif, USA, Oct. 1994.

[10] W. Sung and K. I. Kum, "Simulation-based word-length optimization method for fixed-point digital signal processing systems", *IEEE Transactions on Signal Processing*, vol. 43, no. 12, pp. 3087–3090, Dec. 1995.

[11] J. Babb, M. Rinard, C.A. Moritz, W. Lee, M. Frank, R. Barua, S. Amarasinghe, "Parallelizing applications into silicon", *Field-Programmable Custom Computing Machines, 1999. FCCM '99. Proceedings. Seventh Annual IEEE Symposium on* , vol., no., pp.70-80, 1999.

[12] S. Roy and P. Banerjee, "An algorithm for trading off quantization error with hardware resources for MATLAB based FPGA design," *IEEE Transactions on Computers*, Vol. 54, Issue 7, Jul. 2005.

[13] A. Mallik, D. Sinha, H. Zhou, and P. Banerjee, "Low power optimization by smart bit-width allocation in a SystemC based ASIC design environment,"*IEEE Transactions on Computer Aided Design of Integrated Circuits,* (to appear).

[14] M.L. Ku and C.C. Huang "A complementary codes pilot-based transmitter diversity technique for OFDM systems," *IEEE Transactions on Wireless Communication,* pp. 504-508 2006.

[15] K. Han and B.L. Evans, "Optimum wordlength search using sensitivity information," *EURASIP Journal on Applied Signal Processing,* pp. 1–14, 2006

[16] K. Han, I. Eo, K Kim and H. Cho, "Numerical word-length optimization for CDMA demodulator," *Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on* , vol.4, no., pp.290-293 vol. 4, May 2001.

[17] SystemC 2.0.1 Language Reference Manual, 2003 Available from the Open SystemC Initiative (OSCI) http://www.systemc.org.

[1] J. W. Cooley and J. W. Turkey, "An Algorithm for Machine Computation of Complex Fourier Series," *Math. Computation*, Vol. 19, pp. 297-301, April 1965.

[2] L. R. Rabiner and B. Gold. *Theory and Application of Digital Signal Processing*. Prentice-Hall, Inc., 1975.

[3] E. H. Wold and A. M. Despain, "Pipeline and Parallel-Pipeline FFT Processors for VLSI Implementation," *IEEE Trans. Computers*, vol. 33, no. 5, pp. 414-426, May 1984.

[4] A.M. Despain. "Fourier Transform Computer using CORDIC Iterations," *IEEE Trans. Comput.*, C-23(10):993-1001, Oct.1974.

[5] S. He and M. Torkelson, "A New Approach to Pipeline FFT Processor," in Proc. 10th Int'l Parallel Processing Symp. (IPPS '96), pp.766-770, 1996.

[6] R. Storn. "Radix-2 FFT-pipeline Architecture with Reduced Noise-to-signal Ratio," *IEE Proceedings- Vision, Image and Signal Processing*, 141:81-86, 1994.

[7] S. He and M. Torkelson, "Designing Pipeline FFT Processor for OFDM (de)Modulation", *International Symposium on Signals, Systems, and Electronics*, pp. 257- 262, Oct. 1998.

[8] P. Duhamel, H. Hollmann, "Split Radix FFT Algorithm," *Electronics Letters*, vol. 20, pp.14-16, January 1984.

[9] P. Duhamel, and H. Hollmann, "Split Radix FFT Algorithm," *Electronics Letters*, vol. 20, pp. 14-16, Jan. 5, 1984.

[10] D. Takahashi, "An Extended Split-Radix FFT Algorithm," *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 145-147, May 2001.

[11] G. Nordin, P. A. Milder, J. C. Hoe, and M. Püschel, "Automatic Generation of Customized Discrete Fourier Transform IPs," In Proc. of ACM/IEEE Design Automation Conf., pp. 471-474, 2005.

[12] P. A. Milder, M. Ahmad, J.C. Hoe, and M. Püschel, "Fast and Accurate Resource Estimation of Automatically Generated Custom DFT IP Cores," In *Proc. of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, pp. 211-220 2006.

[13] P. A. Milder, F. Franchetti, J. C. Hoe, and M. Püschel, "Formal Datapath Representation and Manipulation for Implementing DSP Transforms," *In Proc. of ACM/IEEE Design Automation Conf.*, pp. 385-390, 2008.

[14] J. Takala, T.Jarvinen, P. Salmela, and D. Akopial. Multi-port Interconnection Networks for Radix-r Algorithms. In *Proc. IEEE International Conference Acoustics, Speech, Signal Processing*, pp. 1177-1180, 2001.

[15] Synopsys DesignWare[Online], Available: http://www.synopsys.com .

[16] Matlab [Online], Available: http://www.mathworks.com .

[17] R.C. Singleton, "An Algorithm for Computing the Mixed Radix Fast Fourier Transform," *IEEE Trans. on AudioElectroacoust.*, vol. 1, no. 2, pp. 93-103, June 1969.

[18] C.Y. Wang, C.B. Kuo, and J.Y. Jou, "Hybrid Word-Length Optimization Methods of Pipelined FFT Processors", *IEEE Trans. Computers*, vol. 56, no. 8, pp. 1105- 1118, Aug. 2007.

[19] P.D. Welch, "A Fixed-Point Fast Fourier Transform Error Analysis," *IEEE Trans. Audio Electroacoustics*, vol. 17, pp. 151-157, June 1969.

[18] A. Pomerleau, H.L. Buijs, and M. Fournier, "A Two-Pass Fixed Point Fast Fourier Transform Error Analysis," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 25, pp. 582-585, Dec. 1977.

六、 計畫成果自評

This year, we have developed:

1) An expandable multipath delay commutator based FFT architecture
2) An FFT generator to produce a synthesizable FFT core under a given throughput constraint

在計畫的研究成果豐碩，我們共發表一篇待審之長篇期刊，一篇待審之國際研討會論文，一篇國際研討會論文，並申請了中國民國與美國專利。未來將能技術移轉運用於產業界，提升研究的實用價值。雖然，期刊論文的發表進度稍有落後，但是，嚴謹的審查過程，更能確保發表論文的品質。另外，本計畫亦培育多位人才，共有一位博士班研究生、二位碩士班研究生，目前均有很好的發展。

Submitted journal articles

[1] Bu-Ching Lin, Juinn-Dar Huang, and Jing-Yang Jou "Bitwidth-Aware Multiple Constant Multiplication (MCM) Synthesis for FIR Filters," submitted to IEICE Transactions Fundamentals.
[2] Bu-Ching Lin, Jhih-Hong Lu, Juinn-Dar Huang, and Jing-Yang Jou "Delay Optimal Compressor Tree Synthesis for LUT-Based FPGAs," submitted to ACM Transactions on Reconfigurable Technology and Systems.

International conference proceedings

[1] Bu-Ching Lin, Yu-Hsiang Wang, Juinn-Dar Huang, and Jing-Yang Jou "Delay Optimal Compressor Tree Synthesis for LUT-Based FPGAs," submitted to IEEE International SOC Conference.
[2] Yu-Hsiang Kao and Juinn-Dar Huang, "High-Performance NAND Flash Controller Exploiting Parallel Out-of-Order Command Execution," *Proc. of IEEE International Symposium on VLSI Design, Automation, and Test*, pp.160-163, Apr. 2010.

Patent

[1] 黃俊達、呂智宏、林步青、周景揚,"應用於查找表式 FPGA 的壓縮樹延遲最佳化合成演算法," 中華民國專利申請案號 098144372, 98 年 12 月 23 日
[2] 黃俊達、王毓翔、林步青、周景揚,"可參數化管線式快速傳利葉轉換硬體產生器," 中華民國專利申請案號 099100407, 99 年 1 月 8 日
[3] Juinn-Dar Huang, Jhih-Hong Lu, Bu-Ching Lin, and Jing-Yang Jou, "Delay optimal compressor tree synthesis for LUT-based FPGAs," US12/717,520, May 4, 2010

表一簡列近年本研究群的相關研究成果。95 年發表會議論文 3 篇，期刊論文 4 篇，並有 4 篇論文於 IEEE 期刊發表。96 年發表會議論文 3 篇，期刊論文 3 篇，並有 3 篇論於 IEEE 期刊會議論文發表。97 年發表會議論文 1 篇，期刊論文 3 篇，並有 3 篇論於 IEEE 期刊會議論文發表，98 年發表會議論文 1 篇，期刊論文 2 篇，並有 2 篇論於 IEEE 期刊會議論文發表，99 年發表會議論文 2 篇，期刊論文 3 篇，並有 2 篇論於 IEEE 期刊會議論文發表。

Year	Number of Papers				
	Domestic		International		SCI
	Conference	Journal	Conference	Journal	
2006	0	0	3	4 (IEEE: 2)	4
2007	1	0	2	3 (IEEE: 2)	3
2008	0	0	1	3 (IEEE: 3)	3
2009	0	0	2	2 (IEEE: 2)	2
2010	1	0	1	3 (IEEE: 1)	2

表一、本研究群近年相關研究成果

可供推廣之研發成果資料表 一

■ 可申請專利　■ 可技術移轉　日期：99 年 5 月 30 日

國科會補助計畫	計畫名稱：針對通訊數位訊號處理器之電子系統層級驗證與合成環境(2/3) 計畫主持人：周景揚 計畫共同主持人：黃俊達 計畫編號：NSC98-2220-E-009-039　學門領域：EW
技術/創作名稱	高效能且低成本之可參數化快速傅利葉轉換硬體產生器 A Parameterizable Generator for High-Performance and Low-Cost FFT Cores
發明人/創作人	周景揚、黃俊達
技術說明	中文： 快速傅利葉轉換處理器相當廣泛的應用在訊號處理系統及通訊系統中。雖然現存的文獻提供了許多快速傅利葉轉換處理器的架構，但要能夠在給定的條件下挑選出最適合的架構仍是一個相當重要的技術問題。一個快速傅利葉轉換處理器產生器，不但可以增加設計的生產力，同時也可以縮短整個系統設計開發的時程。在這篇論文中，我們針對管線化的快速傅利葉轉換架構提出了面積與通量折衷的方法，且能自動地產生對應的硬體設計。實驗結果顯示，我們在通量的限制之下，可以產生硬體面積較小的架構。
	英文： The Fast Fourier Transform (FFT) processors are widely used in signal processing systems and communication systems. Many FFT architectures are proposed in literature to meet different applications. While designing an FFT processor, one of the most difficult issues is to choose the best architecture under the design constraints. An FFT generator can not only improve the productivity but also shorten time-to-market. In this thesis, we propose approaches which can make appropriate design trade-off between throughput and area of pipeline FFT architectures, and automatically generate the corresponding hardware design. The experimental results show that the proposed methodology can generate area-efficient architectures under throughput constraints.
可利用之產業及可開發之產品	產業: IC、通訊系統晶片設計公司 產品: 無線網路存取設備、手機
技術特點	o 可降低硬體複雜度，減少系統設計時間 o 可加速快速傅利葉轉換處理器之開發
推廣及運用的價值	o 加速通訊、多媒體系統晶片產品開發速度 o 降低系統晶片的設計成本

※ 1.每項研發成果請填寫一式二份，一份隨成果報告送繳本會，一份送　貴單位研發成果推廣單位（如技術移轉中心）。

※ **2.本項研發成果若尚未申請專利，請勿揭露可申請專利之主要內容。**

※ 3.本表若不敷使用，請自行影印使用。