

OFDM/CDMA 통신용 부분 병렬 FWHT 프로세서

황지우, 황용택, 김민수, 유호영*

충남대학교 전자공학과

e-mail : *jwhwang.cas@gmail.com, ythwang.cas@gmail.com,*

mskim.cas@gmail.com, hyyoo@cnu.ac.kr

Partially-Parallel FWHT Processor for OFDM/CDMA Communication

Jiwoo Hwang, Yongtaek Hwang, Minsu Kim and Hoyoung Yoo*

Department of Electronics Engineering

Chungnam National University

Abstract

In this paper, we propose an efficient FWHT architecture for OFDM and CDMA communications, aiming to achieve low hardware complexity and intuitive structure compared to previous designs. Since FWHT is based on Cooley-Tukey algorithm, it has a similar data flow to FFT. Previous research has explored applying FFT structures to FWHT architectures but encountered the trade-off issues between hardware complexity and processing throughput in limited hardware resources. The proposed Partially parallel FWHT architecture utilizes an efficient data reordering process to provide improved performance in terms of hardware complexity and efficiency compared to previous architecture.

I. 서론

OFDM(Orthogonal Frequency Division Multiplex

ing)/CDMA(Code Division Multiple Access)와 같은 다중접속 방식은 무선 통신 시스템에서 한정적인 주파수 스펙트럼 문제를 해결하기 위해 사용된다[1]. 일반적으로 OFDM/CDMA 통신에 사용되는 범용 알고리즘으로는 FFT(Fast Fourier Transform)가 있다. 하지만, 하드웨어 상에서 FFT 연산을 수행하는 것은 상대적으로 많은 연산량과 높은 하드웨어 복잡도를 요구한다. FWHT(Fast Walsh Hadamard Transform)는 FFT 연산에 필요한 곱셈 연산을 포함하지 않으며 덧셈과 뺄셈만으로 연산을 수행할 수 있기 때문에 이러한 문제를 해결할 수 있다.

FWHT는 WHT(Walsh Hadamard Transform)의 고속 알고리즘이다. FWHT의 구조는 FFT의 복소수 곱셈이 포함된 butterfly unit을 덧셈기와 가산기로 구성된 butterfly unit으로 대체하여 구현할 수 있다 [3-4]. 추가적으로, 1과 -1로만 구성된 Hadamard 행렬을 기반으로 하기 때문에 FFT와 같은 정현파 형태의 변환보다 디지털 신호 처리에서 더 효율적이다. FWHT의 각 stage별 butterfly unit들은 분할 정복 알고리즘을 기반으로 더 이상 분할할 수 없을 때까지 주어진 입력 쌍의 덧셈과 뺄셈 연산을 각각 수행한다. 따라서, $N/2$ 개의 덧셈과 뺄셈 연산으로 구성된 각 stage 연산을 총 $\log_2 N$ 번 반복하여 수행할 수 있다.

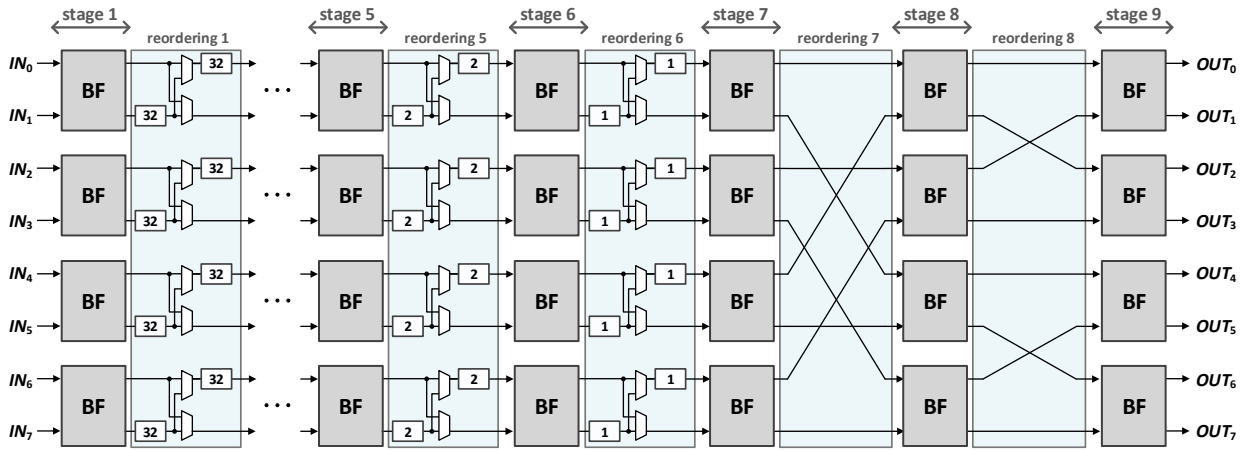


그림 1. 부분 병렬 구조의 FWHT 프로세서 ($N = 512, P = 8$)

II. 일반적인 FWHT 프로세서

일반적으로 FWHT 프로세서는 순차적으로 들어오는 입력에 대해서 하나씩 연산을 수행한다. 본 논문에서는 이러한 직렬 구조로 SDF(Single-path Delay Feedback) 구조를 예로 한다.

SDF 구조는 butterfly unit 연산을 위해 입력 데이터 간 떨어진 거리만큼 지연 요소를 두어 기다린 후 해당 데이터가 입력될 때 butterfly unit 연산을 수행하는 구조이다. 이때, 지연 요소는 feedback loop를 활용한 단일 경로를 사용하기 때문에 직렬의 데이터 흐름을 갖는다. $\log_2 N$ 개의 stage들로 구성된 SDF 구조는 butterfly unit, feedback을 위한 FIFO (First In First Out) 레지스터, 데이터 치환을 위한 multiplexer로 구성된 기본적인 회로를 활용한다. FIFO 레지스터는 각 stage i 별로 ($N/2^i$) 길이를 가지며 feedback loop를 통해 데이터를 지연시킨다. 따라서, $N = 16$ 의 SDF 구조는 각각 8, 4, 2, 1 길이의 FIFO 레지스터를 포함하는 기본적인 회로 4개로 구성된다. SDF 구조는 적은 하드웨어 비용만으로도 구현이 가능하지만 낮은 처리량을 가진다. 따라서, OFDM/CDMA 통신의 이동국을 포함한 실시간에 가까운 빠른 계산을 요구하는 산업 분야에서는 사용하기 어렵다.

III. 제안하는 FWHT 프로세서

본 논문에서는 Folding Transformation[3] 기법을 활용한 데이터 재배열 과정을 통해 처리 속도와 하드웨어 복잡도 사이 trade-off 문제를 해결할 수 있는 부분 병렬 FWHT 프로세서 구조를 제안한다.

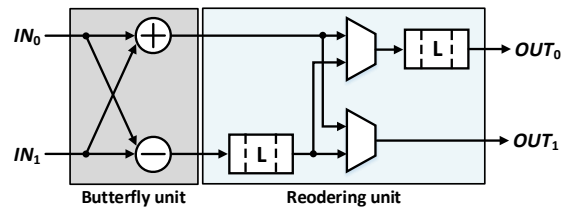


그림 2. 부분 병렬 구조의 기본적인 회로

Folding Transformation은 하드웨어 자원을 절약하기 위해 널리 사용되는 기법 중 하나이며 시간 다중화(Time-Multiplexing)를 기반으로 한다. Folding Transformation 기법에서 folding(ordering) set을 결정하는 것은 중요한 설계 요소 중 하나이다. 제안하는 부분 병렬 구조의 FWHT에서 folding set은 각 stage 별 butterfly unit들의 입력 쌍을 의미한다. 본 논문에서 제안하는 FWHT 구조의 경우 입력 순서가 순차적이기 때문에 다음 stage 입력은 연속된 순서로 입력되어야 한다. 따라서, butterfly unit의 출력을 재배열하는 과정이 필요하다. butterfly unit들의 출력 쌍 순서는 입력 개수 $N = 2^n$ 일 때, stage s 에서 $(n-s)$ 비트만 반전되어 나타나는 특성이 존재한다. 따라서, 반전되어 나타나는 순서를 연속된 순서로 재배열하기 위해 반전되는 비트의 크기와 병렬 인자의 크기를 고려한 지연 소자가 필요하다. 효율적인 butterfly unit들의 입력 쌍, 즉 folding set을 구성하기 위해 그림 2와 같은 기본적인 회로를 활용하여 각 stage 사이에 데이터 재배열 회로를 구현하였다. 제안하는 부분 병렬 구조의 기본적인 회로는 병렬 차원 사이 데이터 교환을 위해 요구되는 지연 시간을 반영한 길이의 FIFO 레지스터 한 쌍과 데이터 치환을 위한 multiplexer 한 쌍으로 구성된다.

그림 1은 $N = 512, P = 8$ 인 부분 병렬 FWHT 프로세서 구조이다. 기본적으로 병렬 인자 P 에 해당하는 부분 병렬 FWHT 프로세서는 한 클럭당

P 개의 데이터가 병렬로 처리되기 때문에 각 stage별로 $(P-2)$ 개의 butterfly unit들과 데이터 재배열을 위한 총 $(N-P)$ 길이의 FIFO 레지스터들로 구성된다. 부분 병렬 FWHT 구조는 각 stage 사이에 존재하는 데이터 재배열 회로에 지연 요소 낭비를 최소화하기 위해 feed back 구조가 아닌 feed forward 구조를 활용하였다. 또한, 데이터 지연이 더 이상 필요 없는 부분은 직접 연결하였다. 제안하는 부분 병렬 구조는 입력 데이터의 개수 N 과 병렬 인자 P 에 따라 해당 설계 방법을 적용하여 대칭적으로 구현할 수 있다. 제안하는 부분 병렬 구조의 설계 방법은 포괄적인 구조로서 모든 병렬 차원에 적용할 수 있다. 따라서, 다양한 설계 환경에 따라 병렬 인자를 선택하여 부분 병렬 구조의 FWHT 프로세서를 설계한다면 가장 효율적인 FWHT 프로세서를 설계할 수 있다.

IV. 구현 결과

표 1은 일반적으로 사용되는 SDF 구조와 제안하는 부분 병렬 FWHT 구조 사이의 하드웨어 리소스, 처리 속도, 처리율, 하드웨어 효율성 측면에서 정량적인 비교를 보여준다. 표 1을 통해 제안하는 부분 병렬 구조의 장점을 명확하게 확인할 수 있다. 제안하는 부분 병렬 구조는 SDF 구조와 비교하여 현실적인 처리량을 제공하면서 유연한 하드웨어 복잡도를 제공할 수 있다. 실질적인 비교를 위해 SDF 구조와 제안하는 부분 병렬 구조의 FWHT 프로세서를 CMOS 180nm 공정을 활용하여 동작 주파수 각각 400MHz, 200MHz로 합성을 진행하였다. 표 2의 합성 결과로부터 제안하는 구조가 SDF 구조 대비 하드웨어 복잡도는 6% 높지만 처리량이 405% 높은 것을 확인할 수 있다. 추가적으로, 처리량과 하드웨어 복잡도 사이 효율성 측면에서 대략 4배 정도 더 효율적인 것을 확인할 수 있다.

V. 결론

본 논문에서는 OFDM/CDMA 통신에서 활용 가능한 부분 병렬 FWHT 구조를 제안하였다. 제안하는 구조를 도출하기 위해 각 stage별 데이터 재배열 과정이 갖는 규칙성을 파악하고 일반화된 데이터 재배열 회로를 제시하였다. 이를 통해 일반적으로 활용되는 SDF 구조의 낮은 처리량 문제를 해결하였다. 제안하는 구조 검증을 위해 진행된 CMOS 180nm 공정 합성 결과 SDF 구조 대비 6% 높은 하드웨어 복잡도를 가지

표1. 하드웨어 복잡도 비교

	SDF	Partially parallel
# Butterflies	$\log_2 N$	$(P/2)\log_2 N$
# Registers	$N-1$	$((N/P)-1)P$
Delays	$N-1$	$N-P$
Latency	$N-1$	$(N/P)-1$
Throughput	1	P
Utilization	50%	100%

표2. 합성 결과 ($N=512, P=8$)

	SDF	Partially parallel
Operating Frequency [MHz]	400	200
Critical path delay [ns]	1.76	3.91
Gate count [#NAND]	74K	78.4K
Latency [ns]	1277.5	315
Throughput [Gbps]	0.4	1.62
Operating Frequency [MHz]	400	200
Efficiency [Kbps/#NAND]	5.41	20.66

는 반면 처리량과 복잡도 사이의 효율성 측면에서 더 효율적인 것을 확인할 수 있다. 결과적으로 제안하는 구조는 직관적인 설계 기법을 통해 FWHT 연산의 입력 개수 N 및 병렬 인자 P 의 변화에도 유연하게 최적화가 가능한 FWHT 프로세서 설계가 가능하여 다양한 설계 환경에서 순차적인 FWHT 출력을 필요로 하는 OFDM 통신 또는 CDMA 통신용 System on Chip (SoC) 및 Application Specific Integrated Circuit (ASIC)에 활용할 수 있을 것이라 기대한다.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2022R1A5A8026986), and supported by National R&D Program through the National Research Foundation of Korea(NRF) funded by Ministry of Science and ICT(2020M3H2A1078119)

참고문헌

- [1] S. Choi, Y. Shin, K. Lim and H. Yoo, "Efficient Partially-parallel NTT Processor for Lattice-based Post-quantum Cryptography," in JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE, vol. 22, no. 6, pp. 459-474, Dec. 2022.
- [2] M. Ayinala, M. J. Brown, and K. K. Parhi, "Pipelined parallel FFT architectures via folding Transformation," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no.6, pp.1068-1081, Jun.2012.
- [3] H. Yoo and I. -C. Park, "Partially Parallel Encoder Architecture for Long Polar Codes," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 62, no. 3, pp. 306-310, March 2015.
- [4] M. Dali, R. M. Gibson, A. Amira, A. Guessoum and N. Ramzan, "An efficient MIMO-OFDM radix-2 Single-Path Delay Feedback FFT implementation on FPGA," 2015 NASA/ESA Conference on Adaptive Hardware and Systems (AHS), Montreal, QC, Canada, 2015, pp. 1-7.
- [5] M. Noshad and M. Brandt-Pearce, "Hadamard coded modulation: An alternative to OFDM for wireless optical communications," 2014 IEEE Global Communications Conference, Austin, TX, USA, 2014, pp. 2102-2107.
- [6] A. Amira, et al. "An FPGA implementation of Walsh-Hadamard transforms for signal processing." 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. Vol. 2, 2001.
- [7] AT Shaheen, "Design and Simulation of parallel CDMA System Based on 3D-Hadamard Transform," in Diyala Journal of Engineering Sciences, vol. 6, no. 4, pp. 54-69, Dec. 2013.
- [8] P. Mazumder, S. Chandra, S. Rana, M. Mukhopadhyay and M. K. Naskar, "Parallel Hardware Implementation of Walsh Hadamard Transform," in Journal of Scientific & Industrial Research, vol. 81, no. 7, pp. 748-753, July 2022.
- [9] S. Hafizullah, M. S. S. V. Srikrishna Manideep, V. Sharma, P. Nath, A. Naugarhiya and S. Verma, "An Efficient Hardware Implementation of Walsh Hadamard Transform for JPEG XR," 2018 15th IEEE India Council International Conference (INDICON), Coimbatore, India, 2018, pp. 1-4.
- [10] A. Garcia, Manjarres, et al. "Parallel-Pipeline Fast Walsh-Hadamard Transform Implementation Using HLS." 2021 International Conference on Field-Programmable Technology (ICFPT), 2021.
- [11] S. S. U. Qadri et al., "Hardware implementation of fast-sequence ordered complex hadamard transform," 2014 IEEE 10th International Colloquium on Signal Processing and its Applications, Kuala Lumpur, Malaysia, 2014, pp. 106-110.
- [12] G. Bi, A. Aung and B. P. Ng, "Pipelined Hardware Structure for Sequence-Ordered Complex Hadamard Transform," in IEEE Signal Processing Letters, vol. 15, pp. 401-404, 2008.
- [13] P. K. Meher and J. C. Patra, "Fully-pipelined efficient architectures for FPGA realization of discrete Hadamard transform," 2008 International Conference on Application-Specific Systems, Architectures and Processors, Leuven, Belgium, 2008, pp. 43-48.