Design and Implementation of DSP algorithms for 100 Gbps Coherent Optical-OFDM (CO-OFDM) Systems

Pramod Udupa, Olivier Sentieys, Laurent Bramerie

HAL Id: hal-00931542
https://inria.hal.science/hal-00931542

Submitted on 15 Jan 2014

Design and Implementation of DSP algorithms for 100 Gbps Coherent Optical-OFDM (CO-OFDM) Systems

Pramod Udupa[1], Olivier Sentieys[1], Laurent Bramerie[2]

[1]INRIA/IRISA,Université de Rennes 1

[2]FOTON,ENSSAT,Lannion

{pudupa,sentieys}@irisa.fr[1],laurent.bramerie@enssat.fr[2]

Résumé − L'OFDM optique utilise la dérection cohérente et des traitements avancés des signaux numériques pour atteindre un débit de transmission des données de 10 Gbps dans une seule sous-bande. Cette exigence stricte en débit apporte une contrainte sur le type d'algorithmes de traitement du signal et les architectures utilisées pour la construction du système. Dans cet article, une architecture parallèle et évolutive utilisant une IFFT radix-2^2 est proposée. La seconde proposition consiste en un algorithme parallèle et évolutif de synchronisation temporelle qui peut supporter des débits d'entrée très élevés au niveau du récepteur. La complexité en nombre de MOPS, ainsi que les coûts en surface vs. débit de l'algorithme de synchronisation, sont donnés pour l'émetteur-récepteur OFDM afin de montrer et caractériser les améliorations dues à l'architecture proposée. L'exploration d'architectures est ralisée en utilisant un outil de synthèse de haut niveau.

Abstract − Multi-band Coherent Optical OFDM (MB CO-OFDM) is widely predicted to be one of the technologies which will empower 100 Gigabit Ethernet (100GbE) networks. CO-OFDM uses coherent technology and advanced digital signal processing (DSP) to achieve net data rate of 10 Gbps in a single band. This strict throughput requirement puts a constraint on the kind of signal processing algorithms and architectures used for building the system. In this paper, a scalable parallel architecture using radix-2^2 for IFFT is proposed. The second proposal consists of a scalable parallel timing synchronization algorithm which can support very high input rates at the receiver. MOPS count as well as area versus throughput for the synchronization algorithm are provided for the OFDM transceiver to show the improvements due to proposed architecture. Architecture exploration is done using a leading-edge high-level synthesis (HLS) tool.

1 Introduction

For next generation 100 Gigabit Ethernet (100GbE), Multi-band Coherent Optical-OFDM (MB CO-OFDM) is proposed to be a very good candidate for the upgrade of the core network [1]. It combines higher modulation formats (QPSK,16-QAM), coherent detection, and advanced digital signal processing (DSP) to reach higher data rates in the allocated bandwidth of 50GHz. Higher modulation formats along with Coherent Detection encode more bits per symbol and DSP algorithms help in combating non-idealities of the front-end and the optical channel.

MB CO-OFDM divides the whole system into multiple orthogonal sub-bands, where each sub-band carries different data and allows the usage of present day signal converters (DAC/ADC) to be used. DAC/ADC are still a strong limitation with respect to the bandwidth of the system since they have bandwidth of 5GHz at data precision of 6-8 bits (higher rates are available but with lower precision and a very high cost and power). Hence sub-banding of the total system relaxes constraints on the DAC/ADC and allows implementation using current DAC/ADC and FPGA/ASIC technologies. To attain the 100 Gbps data

rate in the allocated 50GHz ITU bandwidth, each sub-band has to contribute more than 10 Gbps. A possible solution using an FFT/IFFT size of 256 and a cyclic prefix of 8 consists of using 8 sub-bands. Each sub-band has a bandwidth of 5GHz, thus utilizing a total of 40GHz. The remaining 10GHz is used for guard band and space between sub-bands.

Using dual-polarization on each sub-band and using 10% loss on spectral efficiency, forward error correction (FEC), and zero sub-carriers, the data rate per polarization and per sub-band is

$$D_b = (1 - \epsilon_{fec})(1 - \epsilon_{tr})(1 - \epsilon_{null})(1 - \epsilon_{cp})log_2(M)f_c$$

with $\epsilon_{fec} = 0.0627$ corresponding to RS(255,239), $\epsilon_{null} = 0.1$, $\epsilon_{tr} = 0.1$, $M = 4$, $f_c = 5$GHz, which gives $D_b = 7.3$ Gbps. Using both horizontal and vertical polarizations, we get to 14.6 Gbps per sub-band. Using all the eight sub-bands, we finally obtain a total data rate of 118 Gbps.

The paper is arranged as follows. In Section 2, the computation complexity required for 100 Gbps CO-OFDM transceiver is evaluated. In Section 3, a parallel pipelined architecture using radix-2^2 for FFT is proposed and com-

pared with other radix FFT algorithms and feedback architectures. In Section 4, a parallel architecture for initial timing synchronization is also proposed. In Section 5, scalability and computational complexity of the proposed architectures are examined. Finally, Section 6 concludes the paper.

2   Complexity Evaluation of a CO-OFDM Transceiver

Figure 1 shows the general architecture of a single-band CO-OFDM transceiver, which consists of a set of blocks processing binary data, fixed-point data, and analog data. In the transmitter, binary data processing blocks mainly deal with FEC, scrambling and mapper. Then IFFT and pulse shaping are performed before the DAC. At the receiver side, the signals sampled from the ADC are first processed for synchronization and time-domain corrections before being sent to the FFT block. Finally, frequency-domain corrections, equalization, and channel decoding are performed. The required complexity for an MB CO-OFDM achieving a data rate of 100 Gbps for the transmitter is calculated. Computational complexity for 100 Gbps
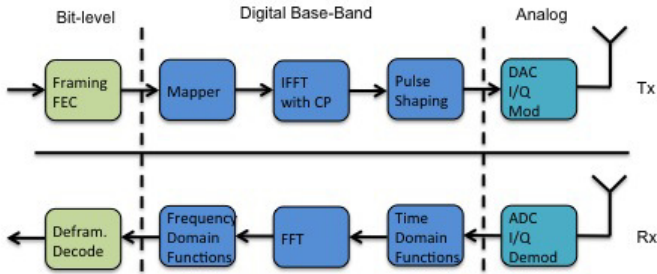

FIG. 1: Block diagram of a single-band CO-OFDM transceiver

transmitter is calculated using total operations and the contribution of the IFFT ($5Nlog_2N$) is pre-eminent. For $N = 256$, the total number of operations is $N_{ops} = 6144$. For 10 Gbps sub-band, the number of operations per second is given by $10^{10}N_{ops}/N$. This gives **400 GOPS** (giga operations per second) for the transmitter only and it doubles to more than **800 GOPS** for a full transceiver. For a 100 Gbps transceiver, the total number of operations for 10 sub-bands combined is $N_{tot} = 10 \times 2 \times 400$ GOPS $=$ **8 TOPS** (tera operations per second).

To achieve the data rate of the MB CO-OFDM with one sub-band in a single FPGA, it is therefore important to architect the algorithms in a parallel manner so as to support the input data rate from ADC at the receiver and output rate for the DAC.

3   Proposed Radix-2^2 Multipath Delay Commutator (MDC) IFFT Parallel Architecture

Since IFFT/FFT blocks contribute significantly to the computational complexity of the transceiver, it is important to have lower complexity architectures and efficient parallelization for supporting high data rates. The radix-2^2 IFFT algorithm [2], [3] combines the simplicity of radix-2 butterfly with the complexity of radix-4 algorithm for multiplications. A radix-2^2 2-Parallel IFFT architecture for size $N = 256$ is shown in Figure 2. It uses complex multipliers only at three stages like the radix-4 architecture. It supports continuous streaming input and due to its simple control mechanism, it can reach very high operating frequencies in an FPGA. A radix-2^2 4-Parallel architecture for $N = 256$ is shown in Figure 3. The 4-Parallel archi-
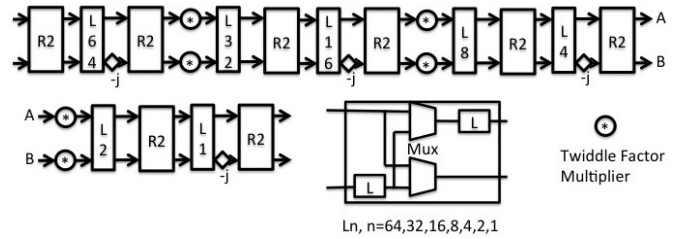

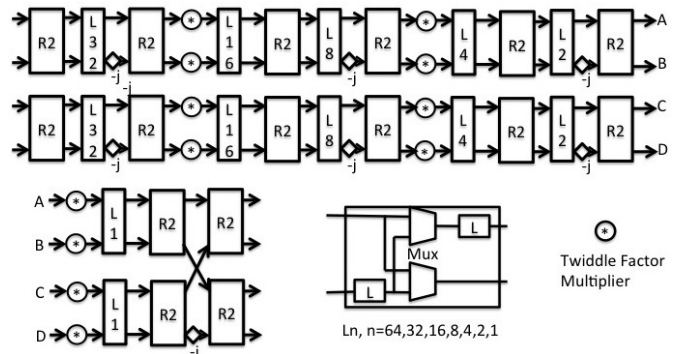FIG. 2: 2-Parallel radix-2^2 IFFT pipelined architecture

FIG. 3: 4-Parallel radix-2^2 IFFT pipelined architecture

tecture uses the same total amount of memory arrays as the 2-Parallel architecture and the same for an 8-Parallel architecture. Thus it is scalable in terms of memory array usage. Since the computation can be separated into even and odd streams, regularity of the architecture can be maintained till the last stage and, only in the last stage, exchange between odd and even streams reduce interconnection complexity and use regular structures to achieve very high speed in FPGA.

Using this architecture in the context of OFDM, the following choice is made for the IFFT architecture: the

input is in normal order and output is in bit-reverse order.

The inputs to IFFT require complex memory array of size $\frac{N}{2}$, while output size requires complex memory array of size $2N + N_{cyp}$ to support streaming input and output, where N_{cyp} is the length of the cyclic prefix. The total memory requirement for IFFT is $\frac{N}{2} + (\frac{N}{P} - 1)P + 2N + N_{cyp}$ to support P streaming outputs every cycle. This is the total size required for all 2- or 4- or 8-Parallel architectures. To support 4-Parallel or 8-Parallel architectures, the memory array is partitioned into smaller chunks for implementation, but the size remains the same. Thus, the proposed FFT/IFFT architecture can support very high output rates (which is an integer multiple of input clock rate) and can therefore be used for CO-OFDM. Comparison of the proposed feedforward (FF) architecture with other radix algorithms like radix-4 and also with feedback architectures is given in Tables 1 and 2.

TAB. 1: Comparison of proposed radix-2^2 MDC 2-Parallel architecture to previously proposed architectures

Algorithm	Radix	Complex "+"	Mem. Size
FF (MDC) [4]	Radix-2	$4(\log_4 N)$	N
FB (MDF) [3]	Radix-2^2	$8(\log_4 N)$	N
FB (MDF) [5]	Radix-2^4	$8(\log_4 N)$	$3N/2$
FF (MDC)	Proposed Radix-2^2	$4(\log_4 N)$	N

FF - feedforward, FB - feedback, MDC - Multipath Delay Commutator, MDF - Multipath Delay Feedback

TAB. 2: Comparison of proposed Radix-2^2 MDC 4-Parallel architecture to previously proposed architectures

Algorithm	Radix	Complex "+"	Mem. Size
FF (MDC) [3]	Radix-4	$8(\log_4 N)$	$8N/3$
FB (MDF) [6]	Radix-2^4	$16(\log_4 N)$	N
FF (MDC)	Proposed Radix-2^2	$8(\log_4 N)$	N

Compared to FB architectures, the proposed architecture uses lesser complex adders and and lesser memory than higher radix-2^4 FFT. It also uses lesser amount of memory size compared to radix-4.

4 Proposed Timing Synchronization Algorithm and Architecture

The timing synchronization [7] algorithm chosen here for optical OFDM is a hierarchical procedure. The first stage *(coarse sync)* [8] is a low complexity auto-correlation step and the second stage *(fine sync)* [9] is a computationally demanding cross-correlation step. The training symbol used is $[C \, C \, C \, -C], C = [A \, B], B = A^*[-n]$. The sign pattern $[+1 + 1 + 1 - 1]$ chosen for steep timing metric roll-off. The equations for coarse synchronization in iterative form are given by

$$M_{init}[n] = \frac{|P[n]|^2}{R^2[n]} \tag{1}$$

$$P[n+1] = P[n] - x^*[n] \cdot x[n+M] + 2x^*[n+2M] \cdot x[n+3M] \tag{2}$$
$$- x^*[n+3M] \cdot x[n+4M]$$

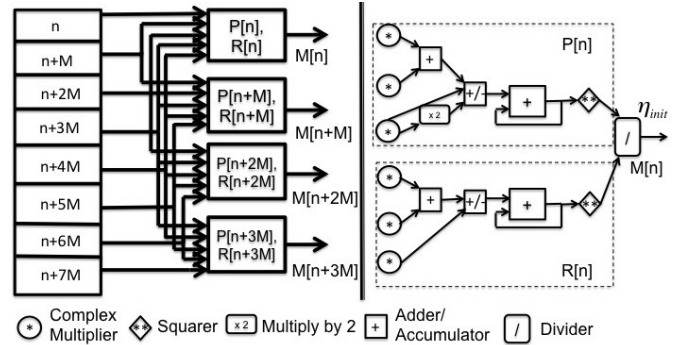$$R[n+1] = R[n] + |x[n+4M]|^2 - |x[n]|^2 \tag{3}$$

FIG. 4: Block diagram of 4-Parallel coarse synchronization

The time index corresponding to the maximum value of $M_{init}[n]$ gives the initial estimate $\widehat{\eta}_{init} = arg \max_n M_{init}[n]$. The proposed architecture performs block-parallel computation. The block size is chosen to be the length of C. A 4-Parallel architecture is shown in Figure 4. The increase in memory for 4-Parallel and 8-Parallel is linear and corresponds to the increase in parallelism achieved: $M = N/4$.

The fine synchronization algorithm operates over $2N_{cyp} + 1$ samples around $\widehat{\eta}_{init}$, where N_{cyp} is the length of cyclic prefix. The fine synchronization algorithm is given by

$$M_{fine}[n] = \frac{|P_{fine}[n]|^2}{R^2_{fine}[n]} \tag{4}$$

$$P_{fine}[n] = \sum_{k=0}^{\frac{N}{2}-1} r[n-k-1] \cdot r[n+k] \tag{5a}$$

$$R_{fine}[n] = \sum_{k=0}^{\frac{N}{2}-1} |r[n+k]|^2 \tag{5b}$$

As can be observed, R_{fine} can be written in an iterative form. P_{fine} can also be computed in a parallel manner by using the locality of memory accesses. Since P_{fine} uses multiplication between samples separated by fixed distance, for the second iteration, the complex sample fetched from memory for the first iteration can be reused and new index point computation can be spawned every

new cycle. Thus the same memory and logic setup used for coarse synchronization stage can be completely reused for the fine synchronization stage. Thus, a scalable parallel architecture for timing synchronization is proposed which can support the high input rates of optical OFDM and quickly provide initial synchronization even in presence of large carrier frequency offset (CFO) of laser.

5   Results

The implementation of both 4-Parallel and 2-Parallel IFFT architectures was synthesized using Xilinx ISE on a Virtex-6 FPGA development board. Frequencies above 400 MHz were obtained for both 2-Parallel and 4-Parallel architectures and for the realization of 5 GHz sub-band,three 4-Parallel IFFT blocks can be used in parallel to attain an output rate of $400 \times 4 \times 3 = 4800$ Msamples per second.

The implementation of parallel timing synchronization algorithm was also performed on a Virtex-6 FPGA development board. Figure 5 provides values of area in number of LUTs vs. throughput in number of clock cycles for different levels of parallelism, for a clock frequency of 200MHz. The implementation is synthesized using CatapultC HLS tool with the algorithm specified in C.
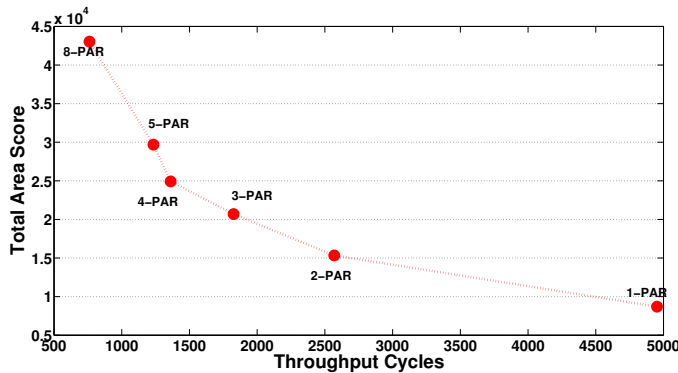

FIG. 5: Area (#LUTs) vs. Throughput (clock cycles) for different levels of parallelism for the synchronization block

6   Conclusion

A pipelined 2/4-Parallel radix-2^2 IFFT architecture is proposed for attaining high throughput rates required for single band CO-OFDM. The gains provided by radix-2^2 in terms of reduction of computations is shown in Table 3. A reduction of 1.2 TOPS is obtained, while still using a simple butterfly compared to radix-4 IFFT. Also, scalable parallel IFFT and timing synchronization architectures are proposed. They can support the very high speed throughput requirement of optical OFDM. Figure 5 gives the area vs throughput trade-off curve which helps in

TAB. 3: MOPS calculation for $N = 256$ size IFFT and 100Gbps O-OFDM Transceiver in Tera OPS (TOPS)

Algorithm	Real "×"	Real "+"	TOPS(GMACS) 100G O-OFDM
Radix-2^2	3072	5632	6.8(240)
Radix-2	4096	6144	8(320)

choosing the optimal parallel version of the architecture.

References

[1] J. D'Ambrosia, *100 gigabit Ethernet and beyond*, in IEEE Communications Magazine, Vol. 48, 2010.

[2] S. He, M .Torkelson, *A new approach to pipeline FFT processor*, in 10th International Parallel Processing Symposium, 1996.

[3] M. Garrido, J. Grajal, M.A. Sánchez and O. Gustafsson, *Pipelined Radix-2^k Feedforward FFT Architectures*, in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 21, January 2013.

[4] S. He, M .Torkelson, *Design and Implementation of a 1024-point pipeline FFT processor*, in Proc. IEEE Custom Integr. Circuits Conf., 1998.

[5] J. Lee, H. Lee, S.I. Cho, and S.-S. Choi, *A high-speed, low-complexity radix-2^4 FFT processor for MB-OFDM UWB systems*, in Proc. IEEE Int. Symp. Circuits Syst., 2006.

[6] H. Liu, and H. Lee, *A high performance four-parallel 128/64-point radix-2^4 FFT/IFFT processor for MIMO-OFDM systems*, in Proceedings of IEEE Asia Pacific Conf. Circuits Syst.,2008.

[7] M. Morelli, C. Kuo and M.-O. Pun, *Synchronization Techniques for OFDMA:A Tutorial Review*, in Proceedings of the IEEE, July, Vol. 95, 2007.

[8] H. Minn, V.K. Bhargava and K.B. Letaief, *A Robust Timing and Frequency Synchronization for OFDM Systems*, in IEEE Transactions on Wireless Communications, Vol. 2, July 2003.

[9] S.D, Choi, J.M.J. Choi and J.H. Lee, *An Initial Timing Offset Estimation Method for OFDM Systems in Rayleigh Fading Channel*, in VTC-2006 Fall,2006.