International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2017): 7.296

Design of FFT Based Multipath Delay Commutator for MIMO-OFDM Systems

Kavitha M V[1], S. Ranjitha[2], Dr. Suresh H N[3]

[1]Research Scholar, Department of E & IE, BIT, VTU

[2]M. Tech, Bangalore, Assistant professor, Gopalan college of Engineering, Studies & Research and Management, Bengaluru, Karnataka, Department of E&IE BIT,Bengaluru-560004

[3]Professor & Coordinator for PG

Abstract: ***This paper describes the design of This paper describes the design of MDC FFT for implementation of MIMO OFDM transceiver using FPGA targeted to future wireless LAN systems. The proposed system is pipeline Radix 2multipath delay commutation FFT has been designed for MIMO OFDM. The MIMO OFDM transceivers have been designed according to the proposed OFDM parameters. A low-power efficient and full-pipeline architecture enables the real-time operations of MIMO OFDM transceivers. . Moreover, due to the straightforward mechanism of the MDC, we have a tendency to propose simple memory planning ways for processor file and output bit/set reversingthat once more leads to a full utilization rate in memory usage. Since the memory needs sometimes dominate the die space of FFT/inverse fast Fourier transforms (IFFT) processors; the planned theme will cut back the memory size and so the die area additionally. Moreover, to use the proposed idea in practical applications, we have a tendency to let Ns = four and implement a 4-stream FFT/IFFT processor with unfolded pipelining as well as 2048, 1024, 512, and 128 for this systems. This processor may be employed in IEEE 802.16 WiMAX and 3GPP long run evolution applications. Finally, we have a tendency to analyze the complexness and performance of the enforced processor and compare it with alternative processors. The results show benefits of the planned theme concerning space and power consumption. For implementation of MIMO OFDM transceiver using FPGA targeted to future wireless LAN systems. The proposed system is pipeline Radix 2multipath delay commutation FFT has been designed for MIMO OFDM. The MIMO OFDM transceivers have been designed according to the proposed OFDM parameters. A low-power efficient and full-pipeline architecture enables the real-time operations of MIMO OFDM transceivers. . Moreover, due to the straightforward mechanism of the MDC, we have a tendency to propose simple memory planning ways for processor file and output bit/set reversingthat once more leads to a full utilization rate in memory usage. Since the memory needs sometimes dominate the die space of FFT/inverse fast Fourier transforms (IFFT) processors; the planned theme will cut back the memory size and so the die area additionally. Moreover, to use the proposed idea in practical applications, we have a tendency to let Ns = four and implement a 4-stream FFT/IFFT processor with unfolded pipelining as well as 2048, 1024, 512, and 128 for this systems. This processor may be employed in IEEE 802.16 WiMAX and 3GPP long run evolution applications. Finally, we have a tendency to analyze the complexness and performance of the enforced processor and compare it with alternative processors. The results show benefits of the planned theme concerning space and power consumption.***

1. Introduction

FAST Fourier transform (FFT) is a basic block in orthogonal frequency division multiplexing (OFDM) systems. OFDM has been adopted in a very broad selection of applications from wired communication modems, like digital subscriber lines (xDSL) to wireless communication modems, like IEEE802.11 WiFi, IEEE802.16 WiMAX or 3GPP long run evolution (LTE), to method baseband knowledge. Inverse fast Fourier remodels (IFFT) converts the modulated data from the frequency domain to time domain for transmission of radio signals, whereas FFT gathers samples from the time domain, restoring them to the frequency domain. With multiple input multiple output (MIMO) devices, knowledge turnout will be increased dramatically. Therefore, MIMO-OFDM systems give promising rate and dependableness in wireless communications To handle "multiple" education streams, intuitively the purposeful blocks ought to be duplicated for the process the synchronal inputs. While not a correct style, the complexness of FFT/IFFT processors in MIMO systems grows linearly with the quantity of information streams.

In-place-memory-updating and pipelines area unit the architectures most generally adopted for the implementation of FFT/IFFT. From the operation perspective, in-place memory change schemes perform the computation in 3 phases: writing within the inputs, change intermediate values, and reading out the results. In change step, the processor reuses the radix-*r* processor, specified one radix-*r* butterfly is ample to finish *N*-point FFT/IFFT computation. Since every part is non-overlapped, the outputs will be serial or as requested. However, it's the non-overlapping characteristic that produces the butterfly idle in memory write and skim phases, and therefore the overall method is extended. Continuous-flow mixed number (CFMR) FFT utilizes 2 *N*-sample recollections to come up with a constant output stream. one in all the recollections is employed to calculate current FFT/IFFT symbols, whereas the different stores the antecedently computed results and controls the output sequence.

Thus, once CFMR is utilized in MIMO systems, the desired memory is increased in a very trend proportional to 2×*Ns*, wherever *Ns* is that the range of information streams. Such memory demand is also taboo if *Ns* is critical, as a result of the realm of mind doesn't shrink the maximum amount as that of logic gates once fabrication technology advances, thanks to the utilization of sense amplify electronic equipment. As for pipeline schemes, single-path delay feedback (wave pipelined technique) and multipath delay commutator (MDC) are the two most well-liked architectures projected a procedure to decompose a separate

Licensed Under Creative Commons Attribution CC BY

Fourier rework matrix so that the FFT processor will be enforced with pipeline consistently.

Wave pipelined technique schemes give feedback methods to manage part computed ends up in every pipe and to get seamless output immediately. The primary output sample will be created at once when the last input sample has been fed into the FFT/IFFT processor that is more, with the planning of data file, wave pipelined technique schemes are capable of process multiple input streams employing a single FFT/IFFT processor. On the opposite hand, MDC systems analyze feedback methods into feed forward streams mistreatment switch-boxes with a lot of memory. Meanwhile, the radix-*r* butterflies idle till the *r* the input is in position. Though the management of information flow in MDC is a lot of samples, the employment rate of the MDC FFT/IFFT computing core is 1*/r*, that is way but the 100 percent utilization rate in wave pipelined technique FFT/IFFT. Sansalone *et al.* instructed that MDC might save a lot of space than wave pipelined technique in FFT with multiple streams and Fu enforced a four-stream MDC FFT/IFFT processor during which the realm was seventy-fifth that of standard stylesto get similarity, radix-2 butterflies were duplicated at the primary stage. At the side of storage components, the central module occupied the most valuable space. To the most active of our information, for the FFT/IFFT processors employed in MIMO-OFDM systems, most of the researchers intuitively duplicated the butterflies and memory in keeping with some information streams so necessary ways that to maximize similarity whereas reducing the hardware quality. Also, few works have thought of output memory required for bit-reversed rearrangement for MIMO FFT/IFFT processors.

These inspire United States of America to explore Associate in Nursing FFT/IFFT design for MIMO systems, which might simply bring home the bacon a 100 percent utilization rate whereas the management mechanism remains secure. Meanwhile, we'd wish to cut back the memory demand for managing bit/set-reversed output order within the new design. During this paper, we have a tendency to take into account MIMO-OFDM systems with *Ns* knowledge streams Associate in Nursing proposes to use single radix-*Ns* butterfly at every folding stage to implement an MDC MIMO FFT/IFFT processor. In standard radix-*r* MDC FFT/IFFT processor with the only knowledge stream, the employment rate is 1*/r*. Hence, *(r*− 1*)/r* computing resource and memory ar wasted. However, for Associate in Nursing MIMO-OFDM system with *Ns* knowledge streams, if we have a tendency to let *r* = *Ns*, the vacancy will be stuffed, and therefore the processor can do a 100% utilization. it's worthy to emphasize that by doing it we tend to would like one butterfly at every pipeline stage solely. Since we tend to use one butterfly to method *Ns* knowledge streams at every pipeline stage, the input file has to be compelled to be regular before passing to the processor. due to the natural management mechanism of MDC, we tend to propose a straightforward mechanism for data planning, wherever the tool is climbable for *Ns* being the ability of two. Moreover, owing to the utilization of 1 butterfly at every stage, we tend to propose a straightforward output planning for bit/set-reversing, which may well scale back the desired output memory. If the output you want memory size is northwest for the one knowledge stream, the dimensions remain nearly *Nw* for multiple streams rather than Nw× *Ns* in conventional schemes, wherever several butterflies square measure required in every pipeline stage. what is more, to use the projected plans in practical applications, we tend to let *Ns* = 4 and perform a 4-stream FFT/IFFT processor with variable length together with 2048, 1024, 512, and 128. This method was enforced employing a UMC ninety nm process and may use in LTE or Wi-MAX applications.

The n point FFT and IFFT are calculated as follows

$$X[K] = FFT\{X[n]\} = \sum_{n=0}^{N-1} a[n] W_N^{nk}$$

And

$$X[n] = IFFT\{x[k]\} = \frac{1}{N}\sum_{K=0}^{N-1} X[K] = \frac{1}{N}\sum_{K=0}^{N-1} X[K] W_n^{-nk}$$

Where

$$W_N^{nk} = \cos\left(\frac{2\pi nk}{N}\right) - j\sin\left(\frac{2\pi nk}{N}\right)$$

The IFFT can be obtained by slightly modifying the FFT. That is, the IFFT of $X[k]$ can be obtained by

$$x[n] = 1N\ (FFT\{X*[k]\}).$$

N is a power of 2, and also the implementation of 1*/N* solely involves right shift operation. Therefore, the IFFT will share a similar hardware with FFT. During this paper, we tend to take LTE and Wi-MAX systems as examples to implement the FFT/IFFT processor. For these two systems, there area unit four FFT/IFFT lengths, that is, *N* = 2048, 1024, 512, and 128. We tend to fold the four FFT/IFFT lengths victimization radix-4 butterflies as repeatedly as doable. Note that the last three stages of the four FFT/IFFT lengths will share a similar hardware. Based on the decomposition

$$X[K]=\sum_{n5=0}^{7}\{\sum_{n4=0}^{3}\{\sum_{n2=0}^{3}\{\sum_{n1=0}^{3} x[N] W_4^{n1k1} W_{2048}^{n1k1}\} \times W_4^{N2K2} w_{512}^{N2K2}\} W_{128}^{n3k3}\} W_4^{n4k4} W_{2048}^{n5k5}$$

Where $k_1+4k_2+16k_3+64k_4+256k_5, k_1 = 0\sim3$,
$K_2=0\sim3, k_3=0\sim3, k_4=0\sim3, k_5=0\sim7$, and
$N = 512n_1+128n_2+32n_3+8n_4+n_5, N_1=128n_2+32n_3+8n_5$,
$N2=32n_3+8n_4+n_5$, and $N3= 8n_4+n_5$

Each brace includes computations of a radix-4 butterfly and a twiddle factor multiplication. For non-power-of-4 FFT/IFFT,

Licensed Under Creative Commons Attribution CC BY

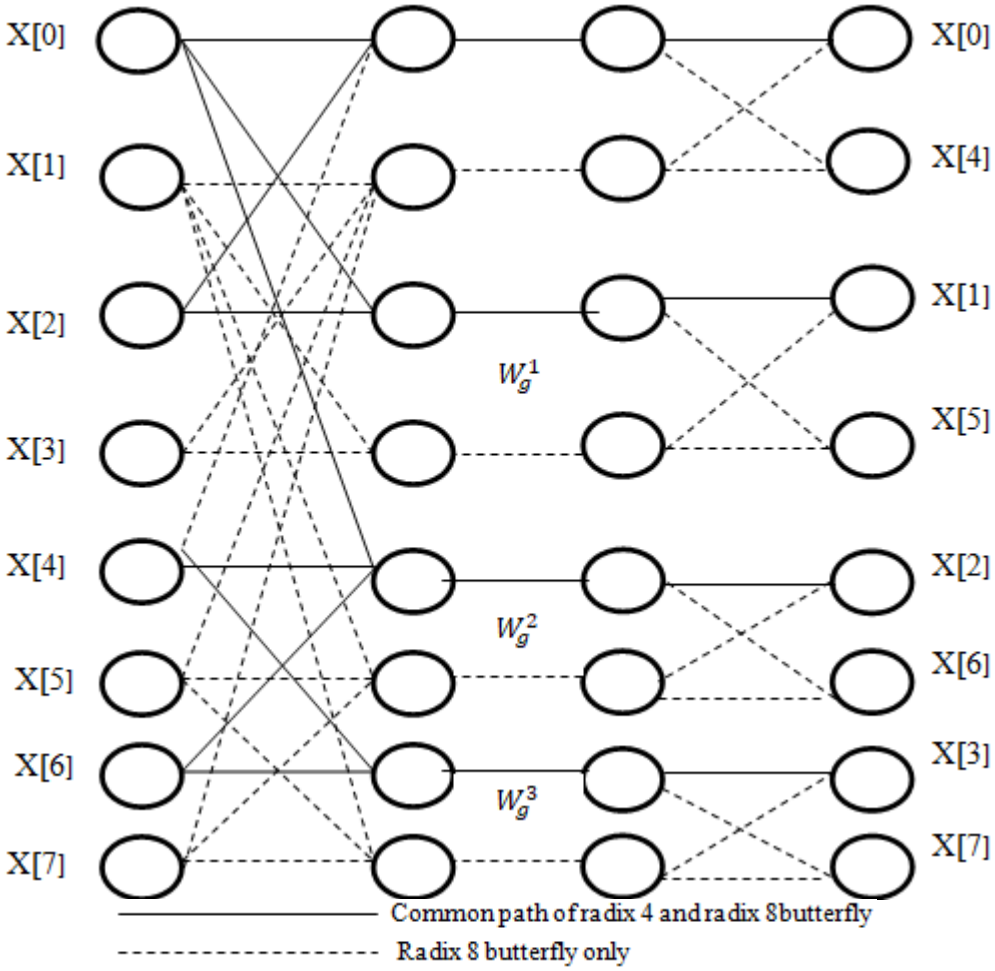


Figure 1: SFG of proposed radix-4, radix 8 butterfly

A radix-8 butterfly is located at the last stage. Hence, the final stage is configurable for each radix-4 and radix-8 computation. The planned radix-4/radix-8 butterfly for the last stage. Wherever a radix-8 butterfly has the information path indicated by each high, dotted lines, whereas a radix-4 butterfly has the information path indicated by solid lines solely. The regularity of the decomposition makes the processor climbable. This implies parameterized register-transfer-level ASCII text file is extremely reusable to increase the number of stages for an outsized N.

2. Related Works

The FFT computation depends on different radix algorithms and architectures. To meet the communication requirement, different designs have been developed to implement corresponding communication condition. Discrete Fourier Transform (DFT) is usually adapted to process digital signals, but it has high computational complexity and takes a long time to implement. Therefore, Cooley and Tukey proposed in 1965 [3] a method to re- duce the computational complexity of DFT. FFT computation rules follow Radix- *n*, where a higher *n* indicates more complex computational standards and circuits. A Radix- *n* algorithm can compute an FFT that has a point size of a power of any multiple of *n*.

Therefore, if the required computation is 512-point FFT, the function needs to be implemented with both Radix-2 and Radix-4. The Radix-2 algorithm reduces DFT computational complexity from $O(N\ 2)$ to $O(N \log N)$ [4,5]. Many FFT architectures have been designed, including memory- based design, cached memory architecture and pipelined architecture [7]. The pipelined architecture has a high throughput, and its hardware area and memory can be reduced by using different FFT algorithms. Therefore, the pipeline is suitable for OFDM. The pipeline architecture can be classified as Multipath Delay Commututor (MDC) and Single-path Delay Feedback (wave pipelined technique) [8]. Wave pipelined technique has the single input and a single output, and so requires FFT computation to be performed in a specific order. Therefore, the system needs to store the earlier input data in a register element; wait for the calculation of the input data to complete, and then transmit the output to the next stage.

A primary N -point wave pipelined technique has several campus- national stages in N -point FFT computation, and each stage con- trains a register element, a butterfly (BF) circuit, and a twiddle- factor computational circuit. The entry part of each stage decreases at each computational stage. After BF computation, the output data are multiplied by the corresponding twiddle factors, then transmitted to the next level. The butterfly circuits and register elements are each changed by a different radix. The complex multiplier number of N -point wave pipelined technique is $(\log 2\ N) -1$. The butter- fly circuits have a complexity of $\log 2\ N$, and require $(N -1)$ register elements. However, the throughput and speed of FFT computation are insufficient to meet the requirement. Therefore, the MDC architecture has been

Licensed Under Creative Commons Attribution CC BY

proposed to obtain high efficiency. The wave pipelined technique architecture is less efficient than the MDC architecture. The MDC architecture is multi-path, and therefore can input multiple FFT data simultaneously. The complexity of N -point MDC multiplier is (log 2 N) −1; that of the butterfly circuits is log 2 N , and the number of register elements is (3 N /2) −1. The comparison of property with wave pipelined technique and MDC. The implementation tools for FFT with FPGA device.

3. Proposed Methodology

3.1 Data Memory Programming

There area unit twelve memory banks at the input stage for changing the parallel data streams into consecutive blocks, such one butterfly at every stage will reason the four knowledge flows while not the idle amount. The twelve memory banks area unit sorted into four memory sets, memory establishes *a*, *b*, *c*, and *d*, that area unit accustomed store the input streams *A*, *B*, *C*, and *D*, severally. There area unit 2 sorts of grouping strategies, explicitly grouping for even indexed symbols and gathering for strange indexed symbols. Let the index of OFDM image begin from zero. For even-indexed OFDM symbols, Four illustrates the memory programming for even-indexed OFDM symbols.

3.2 Unfolded Pipelining technique

The schedule for odd-indexed OFDM symbols can become apparent once the illustration for even-indexed OFDM symbols. Allow us to take N = 2048 as Associate in Nursing example and justify the input schedule as follows. At first, the twelve memory banks area unit logically sorted into four sets. Every set is accountable of 1 input stream. From the primary to the 3N/4th cycle, the memory banks keep the main to 3N/4th samples of every data stream. For the sample of N = 2048, the memory banks, and store the samples 1th–512th, 513th–1024th, 1025th–1536th} of the primary, the second, the third, and also the fourth input streams, severally. From the (3N/4+1)the to the ordinal cycle.

The routing rule updates each N/4s+1 clock cycles. (a) Initial input order. (b) Sorted input order at the output of input buffer. (c) Computed output order while not sorting. (d) Order once output was sorting. Memory set then this memory set square measure updated with the incoming samples from stream *B*, *C*, and *D*. That is, at the side of the antecedently keep 1st to 3N/4th specimens, currently the radix-4 butterfly will method the samples of stream A, as a result of the (3N/4 + 1)the to the ordinal samples square measure prepared at this moment, also, since only 1 butterfly is employed at every stage, the (3N/4 + 1)the ordinal samples for input streams *B*, *C*, and *D* square measure keep within the vacated recollections *a*1, *a*2, and *a*3, severally. Continued with the instance of N = 2048, at the top of the 2048th clock cycle, the radix-4 butterfly has computed the 2048 samples of stream *A*, and also the memory set is updated with the 1537th to the 2048th samples of stream *B*, *C*, and *D*, severally. Similarly, within the next N/4 cycles, the contents of memory set *b* square measure updated. The processor reads out the 2048 samples of stream B from the memory banks *a*1 and sends it to the radix-4 butterfly. Then, the clear recollections *a*1 and square measure updated by the primary to the *N*/4th samples of streams *A*, *B*, *C*, and *D*, severally, of the second OFDM symbols.

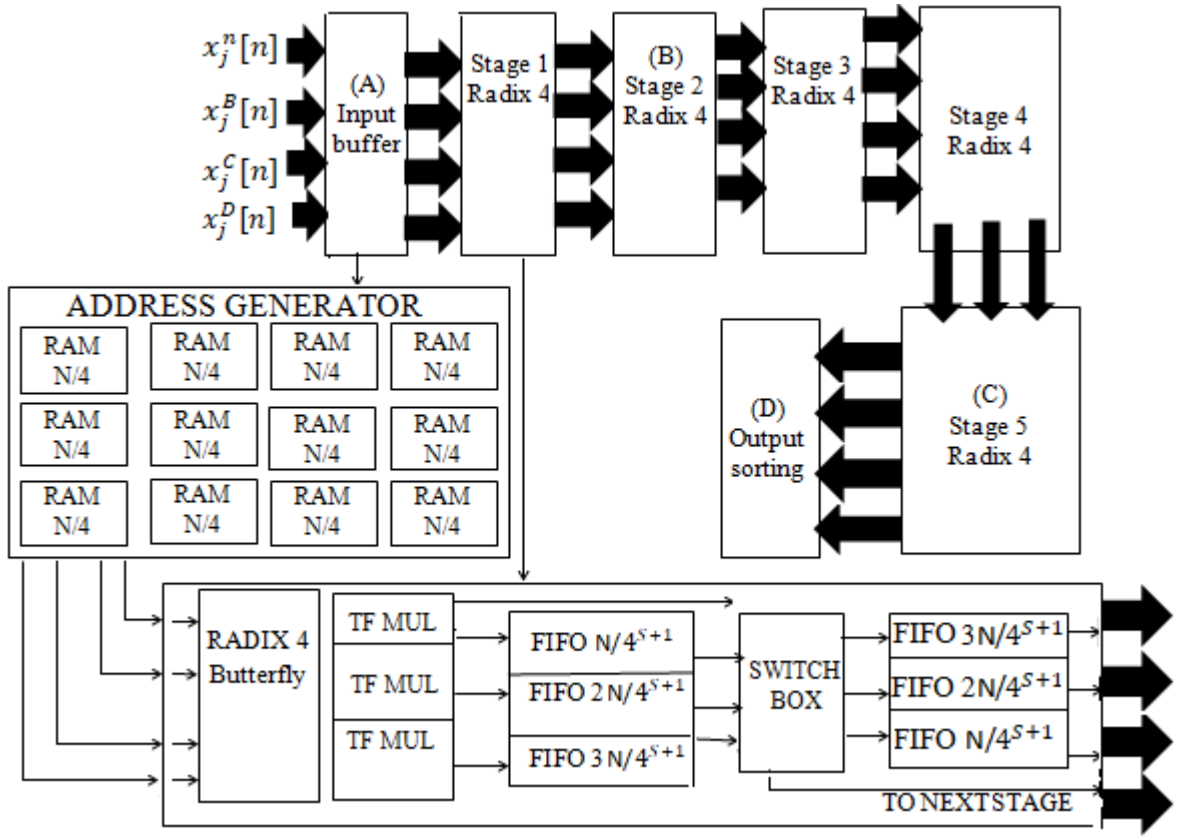


Figure 2: Block diagram of the unfolded pipelining based MDC MDC FFT/IFFT processor for MIMO -OFDM

Licensed Under Creative Commons Attribution CC BY

Continued with the instance of N = 2048, at the top of the 2560th clock cycle, the radix-4 butterfly has computed the 2048 samples of stream *B*, and also the recollections *a*1 and square measure updated with the primary to the 512th samples of stream *A*, *B*, *C*, and *D*, severally, of the second OFDM symbols. Similar procedure is dead for stream *C* and *D*. 3*N*/4th samples of the second OFDM image. Continued with the instance of *N* = 2048, at the top of the3072nd and also the 3584th clock cycles, the radix-4 butterfly has handled streams *C* and *D*, severally. Moreover, at the head of the 3584th clock cycle, all the recollections square measure updated with the primary to the 1536th samples of the second OFDM image. Next, similar procedures mentioned on top of the plaza measure want to handle the second OFDM image. For a practical implementation, the management mechanism of the projected input planning is summarized. Wherever the switch-box at stages updates the routing rule each *N*/4*s*+1 OFDM image time. Adders, Multipliers, and memories are 100%. The computational complexity for each stage is thus one radix-4 butterfly, three twiddle-factor multipliers, and a switch-box with first in first outs (FIFOs). Since ste*ps* needs 3*N*/4*s* words of FIFOs, together with the input scheduling memory that is of 3*N* words, the overall required memory size of the proposed radix-4 MDC FFT/IFFT processor with four parallel data streams is

$$3N + \sum_{S=1}^{[\log_4 N]-1} \frac{3N}{4^S}$$

3.3 Butterfly Operations

The planned FFT/IFFT processor uses radix-4 butterflies as basic computing components. Every stage adopts identical radix-4 butterfly, whereas the last step uses a radix-8 butterfly which may even be designed as a radix-4 butterfly. As for the storage demand of the twiddle factors, sculptures prompt keeping solely the twiddle factors whose part indices area unit inside *N*/8 the remainder of the twiddle factors are often derived from quadrant conversion. As for the advanced multiplications, every radix-4 butterfly wants three multipliers and five real adders. We have a tendency to adopted the routing rule for switch-box planned by Swartzlander. We provide a configurable radix-8/radix-4 butterfly for the last stage, wherever the multiplications of twiddle issue are often accomplished by constant multipliers. This butterfly consists by one radix-4 and four radix-2 butterflies. once a radix-4 rather than a radix-8 computation is required, this butterfly permits the inner radix-4 computations solely and disables the different radix-2 computations.

4. Performance Analysis and Comparison

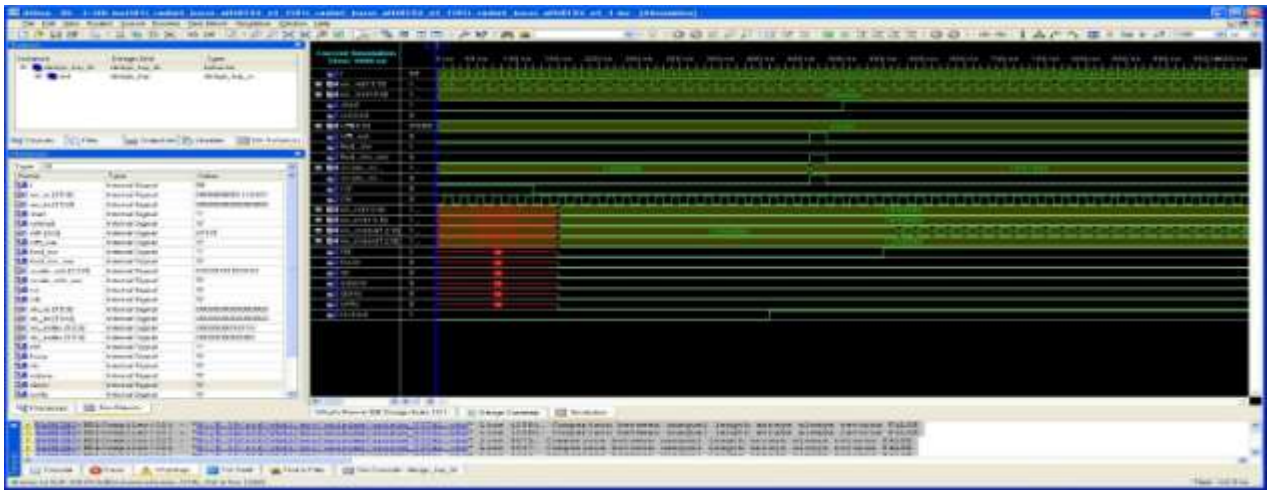


Figure 3: Simulation result

$$A_{bass} = \frac{Area}{(Tech/0.5um)^2}$$

$$P_{bass} = \frac{Tech\ X\left[\frac{2Width}{3\ \ 20} + \frac{1}{3}\left(\frac{Width}{20}\right)^2\right]}{Power\ X\ Exec\ Time\ X\ 10^{-6}}$$

Different FFT length *N*, system frequency, and applied CMOS processes fundamentally affect the area and power consumption. Thus Peng in considered different FFT/IFFT length *N* and proposed the normalized area Open and standardized power *P*eng as

$$A_{Peng} = \frac{\text{Area}}{\text{NX (Tech /0.18)}^2}$$

$$A_{Peng} = \frac{\text{power X Exsec time}}{\text{N X (V}_{DD}\text{/1.8)}^2}$$

Now allow us to cross-check an additional general analysis as follows. The process complexness for associate degree *N*-point FFT/IFFT in radix-*r* is *N* log *N*. Within the practical implementation, the addition and multiplication operations are well scheduled and dead by a little variety of radix-*r* butterflies and complex multipliers. In such a case, the complexness of various *N*-FFT/IFFT ought to grow in ordered series rather than a linear scale. As of the different radix-*r* butterflies, the implementations area unit still supported simple radix-2 structure. Once *N* increases, the quantity of pipeline stages is proportional to log *N*. Therefore; the comparison ought to be normalized to the central radix-2 structure. The facility consumption is proportional to load capacitance, provide voltage, and operational frequency, thatis, $P \propto CV2F$. For comprehensive

Licensed Under Creative Commons Attribution CC BY

and comparable analysis of assorted *N*, design, technology and variety *M* of information streams, we tend to might revise the realm metric as

$$A_{propose} = \frac{\text{Area X } 10^3}{(\frac{\text{Tech}}{0.09\text{u m}})^2 \text{ XM X } \log_2 \text{N}}$$

And the metric for power consumption as

$$P_{propose} = \frac{\text{Power X Esec Time X } 10^3}{\text{M X } V_{DD}^2 \text{ X Nlog}_2\text{N}}$$

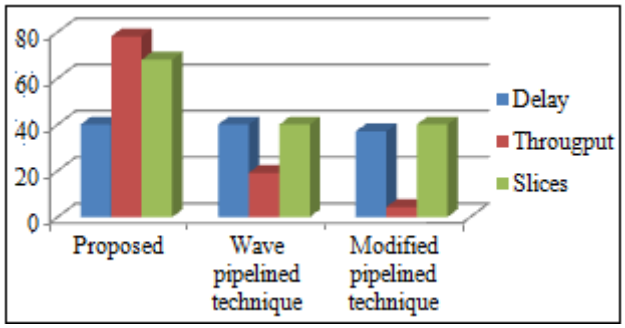
Note that the *VDD* in *0*.09 μm method is one V. There area unit still alternative factors that have an effect on the scrutiny criterion, like the kind of applied RAM macros, the overall load capacitance, or completely different synthesis constraints. Meanwhile, the issue of system frequency isn't enclosed within the revised metrics. operational frequencies in similar style result in the different synthesis and Apr results.

As antecedently declared, completely different fabrication technology and synthesis constraint have an effect on the idea as compared. Therefore, the FFT/IFFT processors with a similar *N* area unit sorted for discussion. For FFT/IFFT processors with *N* = 128, the normalized area for the MDC theme is 67% of that for wave pipelined technique. Note that with higher clock rate, the normalized energy is reduced at the price of the larger normalized space. The trend is carried on to 512- and 2048-FFT/IFFT processors. Now consider the FFT/IFFT processors with a large size of *N* = 2048, where memory macros and storage elements dominate.

The die space. Though the normalized space within the memorybase processor is smaller, the normalized energy isn't reduced proportionately. This can be as a result of the memory-based theme uses second quantity of memory than those in pipeline systems with continuous output. As long as computing parts access the reminiscences, the macros consume power. Consequently, to cut back the realm and electricity consumption to a good *N*-FFT/IFFT processor, decreasing memory usage could also be a key resolution. Moreover, the output latency of the memory-based FFT/IFFT processors is often as long in concert OFDM image. So it's ineffectual to handle consecutive OFDM symbols unless the extraordinarily high clock is employed to method comparatively slow information. Attempting to hunt a decent trade-off between these standard schemes, the projected FFT/IFFT processor adopts accessible memory programming ways for each input and output information; this allows the processor to use a comparatively bit of memory to handle continuous and multiple data streams.

Table 1: Comparison among Different FFT/IFFT Processors

Architecture	Proposed				Wave-pipelined technique	Modified pipelined technique
	MDC				128~2048	128~2048 / 1536
FFT Size	2048	1024	512	128		
Clock rate(MHZ)	40				40	35
Stream no.	4				1	1
Process(um)	0.09				0.18	0.18
Voltage (v)	1				18	1.8
Area (mm^2)	3.1				453	1.932
Output sorting	yes				NO	NO
Power(MW)	63.72	62.72	11.29	51.69	55.64	11.29
Execution time(us)	51.20	25.60	234.00	3.20	55.12	234.00
Normalized energy	36.20	39.33	36.19	46.15	39.07	36.19
Normalized area	70.45				102.92	43.91



Graph 1: Throughput comparison among various 2048-FFT/IFFT processors

Discover's the memory half, which incorporates input reminiscences, intermediate FIFOs, and output sorting takes 85.95% of the general space and 61.72% of the general power consumption. As a result, the programming ways not solely cut back the realm however conjointly contribute to energy saving. Comparing, the projected FFT/IFFT processor uses fewer computing parts, and therefore the execution time is merely a quarter of that for main distribution frame (modified pipelined technique). Moreover, because of the employment of output memory programming, the projected FFT/IFFT processor will handle four information streams and manufacture bit/set-reversed output information at the same time, from integration perspective, the adjacent purposeful blocks like frequency domain equalizer will directly apply the bit/set reversed results from FFT/IFFT processor while not extra effort for rearrangement. The similar clock rate, normalized power, and space are marked to indicate the look trade-off. Though the FFT/IFFT processor is that the highest output with the borderline normalized energy, it's at the price of the larger normalized space and therefore the peak operational rate at three hundred megahertz. Note that the FFT was specifically for 16-quadrature modulation application and therefore the word-size for real half and therefore the imaginary part of a complex number is simply 4-bit. Among the 2048-point FFT/IFFT processors with clock rate below fifty megahertz, our projected style is that the highest output.

Licensed Under Creative Commons Attribution CC BY

5. Conclusion

In this paper, we tend to expected a radix-*r* primarily based MDC MIMO FFT/IFFT processor for process *Ns* streams of parallel inputs, wherever *r* = *Ns* for achieving a 100 percent utilization rate. The projected approach is appropriate for MIMO-OFDM baseband processors like WiMAX or LTE applications, wherever *Ns* = four and *N* are often designed as 2048, 512, 256, and 128. Moreover, we tend to project an economic memory planning to utilize memory totally. This significantly decreases the chip space as a result of the memory demand typically dominates the chip area in an FFT/IFFT processor. It values action that the projected style relies on an MDC design, that isn't most popular, thanks to its low utilization rate in memory and procedure parts like adders and multipliers. However, by mistreatment the projected memory planning, MDC design is verified appropriate for FFT/IFFT processors in MIMO-OFDM systems, as a result of the butterflies and multipliers ar capable of achieving a 100 percent utilization rate. Meanwhile, the characteristics of easy management provided by MDC is maintained within the projected style. The reduction in memory usage conjointly ends up in economic power saving, that is critical for mobile devices. For applications applying sizable amount *Ns* of knowledge streams like gigabit passive optical network, *Ns* are often as high as sixty-four. During this case, the projected radix-*Ns* MDC theme and memory planning can also be applied to attain a 100 percent utilization rate with the straightforward management mechanism. Therefore, we tend to conclude that the projected styles found a decent balance among complexness, energy consumption, and chip space, for the MIMO-OFDM systems.

References

[1] S.N. Tang , C.H. Liao , T.Y. Chang , An area- and energy-efficient multimode FFT processor for WPAN/WLAN/WMAN systems, IEEE J. Solid State Circuits 47 (6) (2012) 1419–1435 .
[2] Kang, B. Kim, J., Low complexity multi-point 4-channel FFT processor for IEEE 802.11n MIMO-OFDM WLAN system, in: Proceedings of the IEEE International Symposium on Green and Ubiquitous Technology (GUT), Jul. 2012, pp. 94–97.
[3] J.W. Cooley , J.W. Tukey , An algorithm for the machine calculation of complex Fourier series, in: Proc. Math. Comput., 19, 1965, pp. 297–301 .
[4] E.J. Kim , H.S. Myung , High speed eight-parallel mixed-radix FFT processor for OFDM systems, in: Proceedings of the IEEE International Symposium on Cir- cuits and Systems (ISCAS), 2011, pp. 1684–1687 .
[5] S. Yoshizawa , A. Orikasa , Y. Miyanaga , An area and power efficient pipeline FFT processor for 8 ×8 MIMO-OFDM systems, in: Proceedings of the IEEE Interna-tional Symposium on Circuits and Systems (ISCAS), 2011, pp. 2705–2708.
[6] T. Cho , H. Lee , A high-speed low-complexity modified Radix-25 FFT processor for high rate WPAN applications, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 21 (1) (2013) 187–191 .
[7] C.C. Hung , P.L. Chiu , Y.H. Huang , A variable-length FFT processor for 4 ×4 MIMO-OFDM systems, in: Proceedings of the IEEE International Symposium on VLSI Design Automation and Test (VLSI-DAT), 2010, pp. 156–159 .
[8] B.C. Lin , Y.H. Wang , J.D. Huang , J.Y. Jou , Expandable MDC-based FFT architec- ture and its generator for high-performance applications, in: Proceedings of the IEEE International SOC Conference (SOCC), 2010, pp. 188–192 .
[9] http://www.xilinx.com , Partial Reconfiguration User Guide, UG702 (v14.5), 2013 (accessed 15.07.14).
[10] http://www.xilinx.com , Partial Reconfiguration of Xilinx FPGAs Using ISE De- sign Suite, WP374 (v1.2), 2012 (accessed 18.08.14).
[11] Trong-Yen Lee ∗, Chi-Han Huang , Wei-Cheng Chen , Min-Jea Liu" A low-area dynamic reconfigurable MDC FFT processor design" Microprocessors and Microsystems 42 (2016) 227–234.
[12] Deepika and Nidhi Goel "*Design of FIR Filter Using Reconfigurable MAC Unit"* 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN).2016

Licensed Under Creative Commons Attribution CC BY