Low Power Commutator for Pipelined FFT Processors

Wei Han[1], T. Arslan[1,2], A.T. Erdogan[1,2] and M. Hasan[1]

[1]School of Engineering and Electronics, University of Edinburgh, Edinburgh EH9 3JL, UK
[2]Institute for System Level Integration, Livingston EH54 7EG, UK

{W.Han,T.Arslan,Ahmet.Erdogan}@ee.ed.ac.uk

Abstract—**This paper proposes a low power commutator architecture for the implementation of radix-4 based pipelined Fast Fourier Transform processor. The architecture is based on dual port RAM blocks and exploits the interconnection topology among these blocks for low power implementation. The paper presents the commutator architecture, describes the design methodology and evaluation environment, and provides implementation results showing that the new commutator achieves up to 58% power saving for 256-point and 128-point FFTs as compared to previous commutator architectures.**

I. INTRODUCTION

When the Fast Fourier Transform (FFT) was popularized by Cooley and Tukey in the 1960s, it was only an algorithm based on software. The hardware implementation looked like problematic at that time. However, with the evolution in semiconductor technology, and the advent of SoC (Systems-on-Chip), FFT algorithms are being prototyped as parameterizable cores which could be embedded within a SoC platform [1]. Recently, under more and more demands for applications on portable systems such as chips for wireless local area networks (WLAN) and mobile terminals, the power consumption of FFT processors is being elevated to be one of the most critical design requirements.

An FFT processor can be classified into three main types: Pipelined FFT, Column FFT, and Fully parallel FFT [2]. Pipelined FFT is preferred especially for a high throughput demand or low power solution [3]. In real-time applications, input data is a sequential stream. Therefore it does not match the FFT algorithm since the FFT requires temporal re-ordering of data [4]. For this reason, the commutator is needed to reorder the input data. Among several pipelined FFT architectures, Radix-4 Single-path Delay Commutator (R4SDC) is widely used, owing to its high utilization of multipliers, butterfly elements and memory blocks [3][4]. The proposed architecture in this paper is an improvement in power efficiency compared to previous commutator architectures used in R4SDC pipelined FFT.

For the pipelined architectures, the commutator and the complex multiplier units contribute a dominating part of the overall power consumption. Moreover, the commutators will take up more proportion of the overall power consumption and act as a leading actor with the increase of FFT size. Therefore, reducing the power consumption of the commutator units is crucial for the low power implementation of pipelined FFT processors.

It is well known that the switching power is mainly responsible for power consumption in CMOS circuits. This power, P_{sw}, is given by [5]:

$$P_{sw} = \frac{1}{2} k C_{load} V_{dd}^2 f \qquad (1)$$

where *k* is the average number of times the gate makes an active transition during one clock cycle, *f* is the clock frequency, *Vdd* is the supply voltage, and *Cload* is the load capacitance of the gate. Hence, for achieving low power, one or more of the parameters *Cload*, *Vdd* and *k* need to be minimized. However, since *Cload* and *Vdd* are relative to the target technology, *k* becomes the main point of improvement. Therefore, this paper focuses on the reduction of the switching activity in the commutator units of FFT, at the same time, reducing the number of write operations to memory units, hence achieving a significant power saving as compared to previous commutator architectures.

II. ALGORITHM

The N-point DFT of a finite duration sequence x(n) is defined as follows [6]:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \qquad k = 0,1,2,\cdots,N-1; \qquad (2)$$

Where W_N is defined as $W_N = e^{-j(2\pi/N)}$

Let *N* be a composite number of *v* integers so that $N = r_1 r_2 \ldots r_v$ and define

$$N_t = N/(r_1 r_2 \cdots r_t) \qquad 1 \le t \le v-1 \qquad (3)$$

where *t* is the stage number of the decomposed DFT and r_t is its radix. The pipelined FFT processor is obtained by decomposing an N-point DFT into *v* stages. The final stage is defined in [7] as follows:

$$X(r_1 r_2 \cdots r_{v-1} m_v + \cdots + r_1 m_2 + m_1) = \sum_{q_{v-1}=0}^{r_v-1} x_{v-1}(q_{v-1}, m_{v-1}) W_{r_v}^{q_{v-1} m_v} \qquad (4)$$

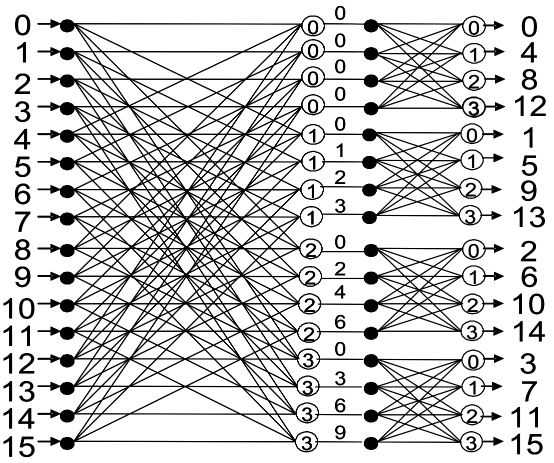


Fig. 1. Signal flow graph of a radix-4 16-point FFT [7].

while intermediate stages (t) are given by the following recursive equation [7]:

$$x_t(q_t, m_t) = W_{N_{t-1}}^{q_t m_t} \sum_{p=0}^{r_{t-1}} x_{t-1}(N_t p + q_t m_{t-1}) W_{r_t}^{p m_t} \quad (5)$$

where, for both (4) and (5)

$2 \le t \le v-1,\ 0 \le m_i \le r_i - 1, 0 \le q_i \le N_i - 1$ and $2 \le i \le v$

As an example, the signal flow graph of a radix-4 16-point FFT based on the above equations ($N=16$, $v=2$, $r_1=r_2=4$) can be seen in Figure 1, where each open circle denotes a summation while the closed circles define the stage borders. The number inside the open circle is the value of m_1 (for stage 1) or m_2 (for stage 2). The number outside the open circle is the FFT coefficient.

IMPLEMENTATION

A. Conventional Commutator Implementations

The R4SDC architecture was first proposed by Bi and Jones [7]. A pipelined N-point radix-4 FFT processor based on this architecture, shown in Figure 2, has $\log_4 N$ stages.

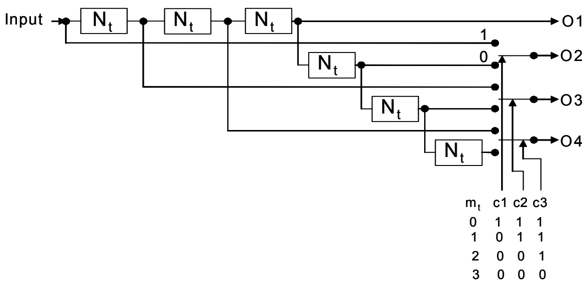


Fig. 3. General commutator architecture for R4SDC [1].

Each stage produces one output within each word cycle. Each stage contains a commutator, a butterfly element and a complex multiplier. The sequential outputs at each stage must be ordered in accordance with the value of m_t. For instance, from Figure 1 at stage1, the outputs associated with $m_1 = 0$ are produced in the first four word cycles, then those associated with $m_1 = 1$ in the next four cycles and so on. It is clear from equation 4 that input data for each summation at *stage t* are separated in time by N_t words. The required commutator comprises of six shift registers, each proving N_t words delay along with three multiplexers and is shown in Figure 3. The control signals C_1, C_2 and C_3 select the appropriate data using 2:1 multipliers according to the value of m_t. The four complex outputs from the commutator are connected to its associated butterfly. The commuator supplies the same set of data for N_t word cycles.

In [7], each FIFO is implemented through a set of shift registers. We term this implementation as SR. A memory based implementation is also widely used. The conventional approach uses dual port RAMs to realize FIFO elements of Figure 3. We term this implementation as DR. In [2], the authors used single port RAMs to replace dual port RAMs. This was due to smaller area of single port RAMs. The authors in [1] presented another architecture based on triple port RAMs (TR) in which three FIFO blocks of size $2N_t$ were used instead of six FIFO blocks of size N_t. The use of larger RAMs leads to less data movement as compared to other architectures and hence more power savings. However, as was discussed in [1], TR is not suitable for long FFTs, since the power consumption in the additional port increases considerably due to larger RAMs. Another disadvantage of

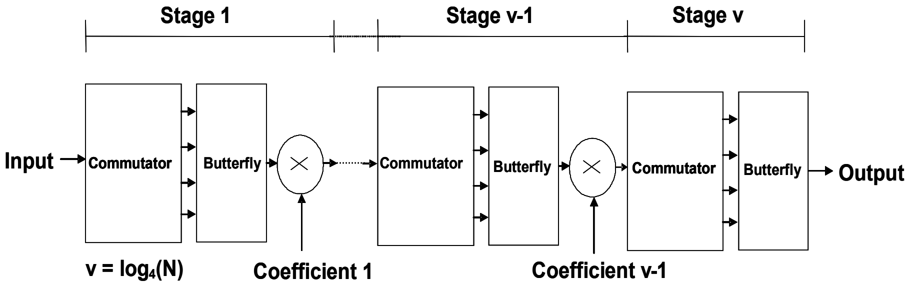


Fig. 2. R4SDC N-point radix-4 pipelined FFT processor architecture [7].

TR is the increase in area, since triple port RAMs are larger than dual port RAMs. Other methods, such as delay lines have also been used in some designs [8].

B. Proposed Commutator Architecture

We propose a new low power commutator architecture as shown in Figure 4. The architecture, which is termed as IDR, uses dual port RAMs as FIFO elements, however, the interconnection topology among the RAM blocks is different from that of the conventional approach. Table 1 illustrates which RAM blocks are enabled for write access during each period. It can be seen that there are at most three RAM blocks selected in a given period.

TABEL1
RAMS SELECTED IN DIFFERENT PERIODS

m_t	0	1	2	3
RAMs selected	DM1 DM3	DM0, DM2, DM4	DM1, DM3, DM5	DM0 DM2

For *stage t*, when m_t is equal to 1, new N_{t-1} input data is processed. The first N_t data will be written into DM0. The previous N_t data stored in DM0 will be read out and written into DM2 for vacating space for the new data. The same applies for DM2 and DM4. The other three RAM blocks (DM1, DM3 and DM5) will be disabled for write access during this period. For m_t = 0 and 3, the number of RAM blocks enabled is two, because the previous data stored in DM2 and DM3 are no longer needed for subsequent outputs. Therefore, during the four periods, each RAM is enabled 5/3 times on average. Whereas, for DR and TR architectures this corresponds to 4 and 10/3 times respectively. Hence our new commutator architecture is significantly more power efficient compared to other commutator architectures.

In Figure 4, the signal bus *m[6:0]* generated by the controller unit is used to control the four multiplexers. The control signals (*c4, c5,* and *c6*) are for the butterfly unit. The values on the RAMs' output ports (A, B, D, E and F) are obtained by controlling their read addresses (*radrra – radrrf*) which are generated by the controller unit. The RAM blocks are enabled only if their outputs are currently needed in order to prevent unnecessary switching activity. The timing diagram of the proposed commutator for stage1 of a 16-point FFT is shown in Figure 5, where '*t*' is the instant when the first input word arrives. Each input word occupies a word slot of duration *T* and is numbered according to its appearance in time. From Figure 5, it can be seen that among the six output ports (A – F), only C is needed full-time, others can be idle for 25% to 75% of the time.

III. SIMULATION RESULTS

The four commutator architectures (namely SR, DR, TR and IDR) have been implemented with Verilog HDL for the first three stages of 256-point and 128-point FFTs. The final stage of the 128-point FFT is based on radix-2, and N_t for the final stage of the 256-point FFT is only 1. Therefore, shift registers were used for the implementation of the final stages (stage 4) for both FFTs. Input data used were 32 bits complex data. Designs were synthesized using Synopsys Design Compiler targeting the UMC 0.18μ CMOS technology library at a clock frequency of 12.5MHz. Power evaluation was carried out using Synopsys Design Power at a clock frequency of 10MHz.

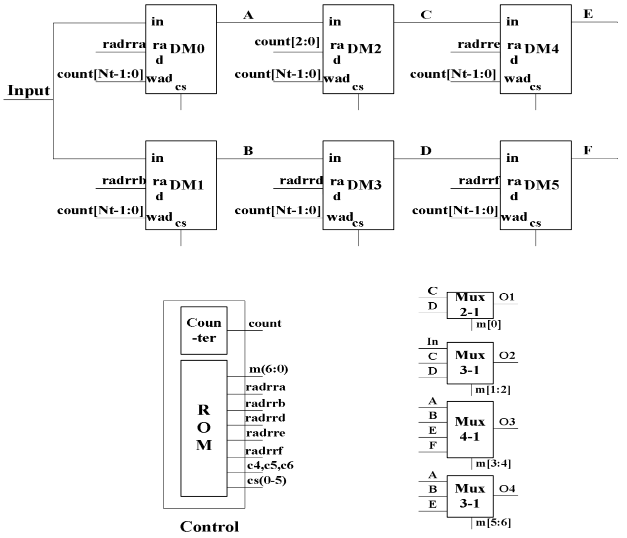


Fig.4. Proposed commutator architecture.

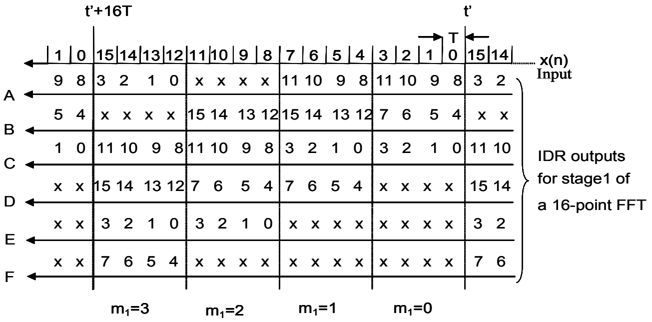
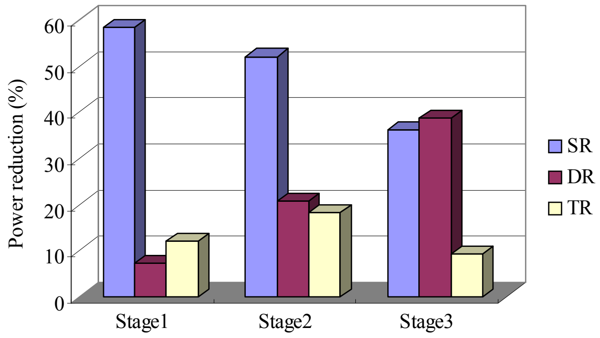
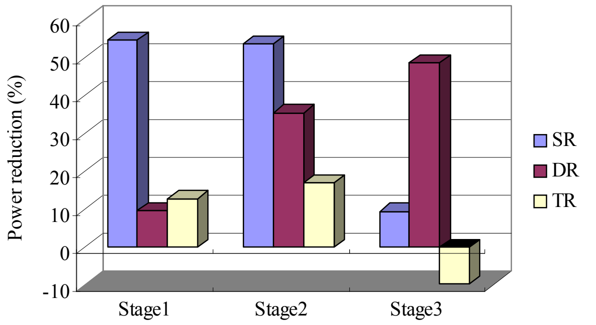


Fig. 5. Timing diagram of IDR outputs for the first stage of a 16-point FFT processor.

The comparative results in terms of power reduction for the first three stages of the 256-point FFT and those of the 128-point FFT are given in Figure 6. It is clear that the IDR commutator architecture achieves significant power savings compared to previous commutator architectures. For example, compared to SR, power reduction is maintained above 30% except for the third stage of the 128-point FFT where power reduction drops to 10%. This is because the size of the FIFO for the third stage is only 2, which is too small to be implemented efficiently as a memory module. In the case of DR, power saving increases from 5% to 36%

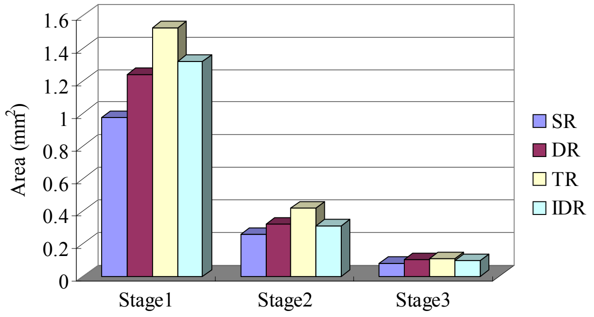


(a) 256-point FFT

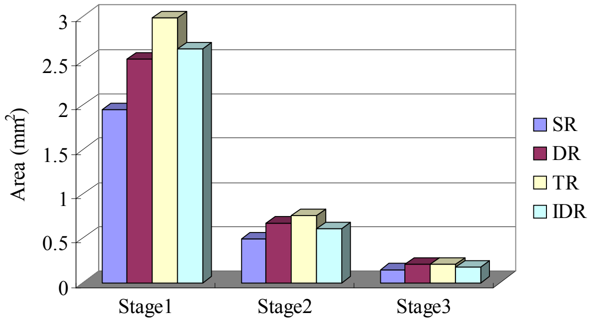


(b) 128-point FFT

Fig. 6. Power comparison of IDR relative to SR, DR and TR.



(a) 256-point FFT



(b) 128-point FFT

Fig. 7. Area comparison.

with the decrease in memory size. Compared with TR, the power saving is between 7% to 15% for 256-point FFT, and 13% to 17% for 128-point FFT, except for stage3 where power increases by 10%. This increase in power consumption is derived from the fact that the size of each memory block (DM0 – DM5) in stage3 is only 2 words for 128-point FFT. Hence, power savings achieved by IDR through the reduction of memory write accesses cannot compensate for power consumption due to more data movement as compared to TR.

A comparison of area usage for the four architectures is given in Fig. 7. Clearly, SR architecture has the smallest area in all stages, due to its simple structure. Our new commutator architecture (IDR) excels TR in all stages, and is commensurate with DR. Clearly, for the first three stages of 256-point FFT and the first two stages of 128-point FFT, the proposed commutator architecture has the best tradeoff between power consumption and area usage. However, for stage3 of 128-point FFT, SR architecture is a better choice.

IV. CONCLUSION

This paper has presented a new low power commutator architecture based on dual port RAMs for a pipelined radix-4 based FFT processor. The proposed commutator architecture was compared to three alterative architectures, using 128-point and 256-point FFT examples. It was shown that the proposed commutator architecture achieves up to 58%, 36% and 17% power savings compared to shift register, conventional dual port RAM and triple port RAM based implementations respectively.

REFERENCES

[1] Mohd. Hasan and Tughrul Arslan, “A triple port RAM based low power commutator architecture for a pipelined FFT processor” Circuits and Systems, 2003. ISCAS '03. May 2003, pp. 353 -356

[2] W. Li and L. Wanhammar, “A pipeline FFT processor” IEEE Signal Processing Systems Worksho 1999. Oct. 1999, pp. 654 - 662

[3] S. He and M. Torkelson, “Design and implementation of a 1024-point pipeline FFT processor” Custom Integrated Circuits Conference, 1998. Processing of the IEEE 1998, 11-14 May 1998. pp. 131 - 134

[4] W. Li, Y. Ma and L. Wanhammar “Word length estimation for memory efficient pipeline FFT/IFFT processor” Inter. Conference on Signal Processing Applications & Technology (ICSPAT), Nov., 1999.

[5] J. M. Rabaey and M. Pedram, Low Power Design Methodologies, Kluwer Academic Publishers, 1997, pp. 6-8

[6] W. W. Smith and J. M. Smith, Handbook of Real-time Fast Fourier Transforms Wiley-IEEE press, 1995, pp. 10-11

[7] G. Bi and E. V. Jones, “A pipelined FFT processor for word-sequential data” IEEE Transactions on acoustic, speech, and signal processing, Vol. 37 , no: 12 , Dec. 1989, pp.1982 - 1985

[8] E. Bidet, D. Castelain, C. Joanblanq and P. Senn, “A fast single-chip implementation of 8192 complex point FFT” IEEE JSSC., vol. c-30, no. 3, pp. 300-305, Mar. 1995.

[9] M. Hasan Low Power Techniques and Architectures for Multicarrier Wireless Receivers PhD thesis, University of Edinburgh, 2003.