

Implementation of low-power FFT processor cores using a novel order-based processing scheme

M. Hasan and T. Arslan

Abstract: The authors present a novel order-based coefficient processing scheme for the realisation of low-power FFT processors. The scheme is based on the minimisation of the Hamming distance between successive coefficients fed to the butterfly. A distinct feature of the scheme that distinguishes it from conventional order-based schemes lies in the fact that either the real part of the coefficient or its two's complemented value is used for the minimisation of the Hamming distance between successive coefficients and hence the switching activity. The paper describes the scheme and its implementation, and provides results using a number of fully synthesised FFT processor cores. The results demonstrate that the switching activity is reduced by up to 53% for different FFT lengths compared to only 27% when conventional order-based processing is employed. This significant reduction in switching activity leads to power savings in the range of 25% to 1% for different FFT processor cores.

1 Introduction

One of the fastest growing areas in the computing industry is the provision of high throughput low-power digital signal processing (DSP) and telecommunication systems in portable forms. With the advent of SoC (silicon on chip) technology, parameterisable low-power fast fourier transform (FFT) cores are being prototyped which could be embedded within the SoC platform.

It can be shown that the main source of power consumption in a typical CMOS logic gate is due to the switching power, P_{sw} , given by [1]:

$$P_{sw} = \frac{1}{2}kC_{load}V_{dd}^2f \quad (1)$$

where V_{dd} is the supply voltage, f is the clock frequency, C_{load} is the load capacitance of the gate and k is the switching activity factor, which is defined as the average number of times the gate makes an active transition in a single clock cycle. Therefore to achieve low-power in CMOS circuits one must aim to minimise one or more of the parameters C_{load} , V_{dd} and k . This paper primarily deals with low-power architectures obtained by reducing the switching activity.

The N -point discrete fourier transform (DFT) is defined as follows:

$$X_k = \sum_{m=0}^{N-1} x_m \cdot W_N^{mk}$$

where $W_N = \exp(-j2\pi/N)$ and $k=0, 1, \dots, N-1$. The radix-2 FFT [2] is an efficient way to compute an N -point DFT. The basic operations in an FFT are multiplication of the complex data inputs (x_m) by the FFT coefficients ($\exp(-j2\pi k/N)$) at each stage of the signal flowgraph, as

shown in Fig. 1 for a 16-point FFT. This is followed by their summation or subtraction and associated data and coefficient address generation.

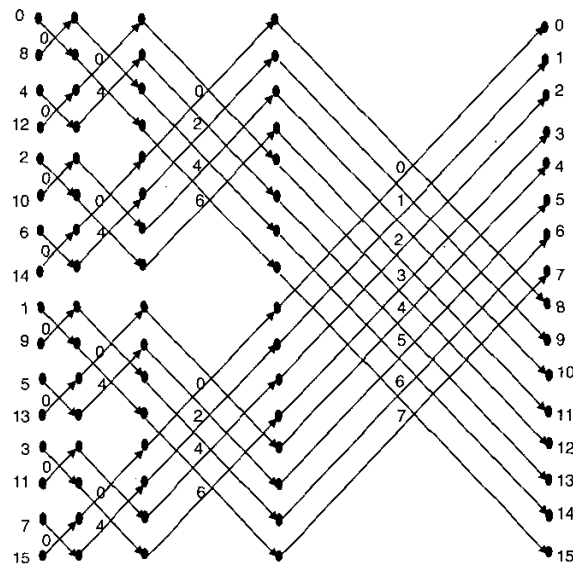


Fig. 1 Signal flowgraph of a 16-point FFT as an example

A number of researchers have investigated low-power implementation of FFT processors. In [3], the author implemented a low-power cache-memory architecture by using an algorithm that offers good data locality over large portions of the computation. This architecture is more suitable for longer length transforms. In [4, 5], the authors proposed a low-power architecture based on an algorithm that effectively minimises the number of complex multiplications. In [6, 7], the authors investigated the realisation of low-power FFT processors by using asynchronous processing elements. In [8], a multiplierless DCT is proposed by realising multiplication through shift-and-add for

© IEE, 2003

IEE Proceedings online no. 20030347

doi:10.1049/ip-cds:20030347

Paper first received 12th February 2002 and in revised form 8th October 2002

The authors are with the Department of Electronics and Electrical Engineering, The University of Edinburgh, The King's Buildings, Mayfield Road, Edinburgh EH9 3JL, UK

low power. The only reported work on the application of order-based processing to FFT coefficients involves the order-based processing of coefficients and data at the inputs of the FFT computational units so as to minimise the overall switching activity of successive coefficients and data samples [9, 10]. This results in a coefficient switching activity reduction of just 19%, whereas the data activity increases by 1% for a 9-point FFT [10]. The authors in [9, 10] have not shown the actual power reduction obtained by their scheme in the presence of hardware overheads in the full FFT architecture. We have demonstrated that the conventional order-based processing scheme does not lead to any power savings because of the small reduction in switching activity and overheads required to realise the scheme. To our knowledge, no work exists in the literature on the development of architectures which utilise an effective coefficient sequencing scheme in order to improve the performance of the FFT processor in terms of power. Our order-based processing scheme is based only on fixed FFT coefficients, as are most other order-based processing schemes in DSP blocks, such as FIR filters [11]. Our method of order-based coefficient processing in FFT processors is also based on Hamming distance minimisation between successive coefficients fed to the butterfly, with the difference that either the real part of the coefficient or its two's complemented value is used for the minimisation of switching activity. We show that this procedure results in a significant reduction in switching activity of the order of 53%, compared to only 27% with the conventional order-based processing approach in the case of FFT processors. This reduction in switching activity leads to power savings in the range of 25% to 1% for different FFT processor cores.

2 Order-based coefficient processing

The conventional order-based processing approach is based on arranging the coefficients in the FFT processor so as to minimise the Hamming distance (bit transitions or switching activity) between successive coefficients fed to the multipliers of the butterfly module. The basic idea is to reduce the switching activity at the coefficient input of the power-consuming multiplier blocks.

In an FFT processor, the positions of the real and imaginary parts of the coefficient set cannot be changed independently (every real part has a corresponding imaginary part and *vice versa*). The reduction in switching activity of the coefficient set by the conventional order-based method is insignificant because the switching activities of the real and imaginary parts of the coefficient set are almost complementary. For instance, if one changes the order of only the imaginary parts of the coefficient set for minimum switching activity, the new order of the real part will be such that the net activity of the overall coefficient set will not reduce substantially and *vice versa*.

The proposed scheme addresses the problem by either using the real part of the coefficient or its two's complemented value on the basis of a minimum Hamming distance with the preceding real part. The scheme can be best illustrated with the help of a flowchart shown in Fig. 2. Let X_R and X_I be the $N/2$ element real and imaginary part arrays respectively, for an N -point FFT. Functions used in the flowchart are defined as follows:

$Ham(i, j)$ – Hamming distance between array elements $X_I(i)$ and $X_I(j)$.

$HamN(i)$ – Hamming distance between elements $X_R(i)$ and $X_R(i-1)$.

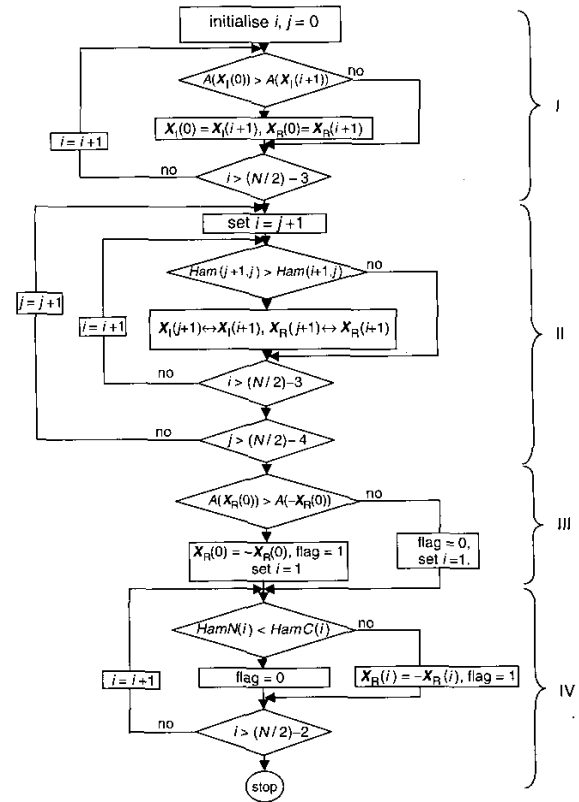


Fig. 2 Flowchart of the order-based processing scheme

$HamC(i)$ – Hamming distance between elements $-X_R(i)$ and $X_R(i-1)$.

$A(X_I(i))$ – number of 1s in array element $X_I(i)$.

The flow chart is divided into four sections. Section I selects the first imaginary part of the ordered coefficient set on the basis of a minimum number of 1s. Section II arranges the remaining imaginary parts of the coefficient set on the basis of the minimum Hamming distance between successive imaginary coefficients. After sections I and II, the imaginary parts of the coefficient set are ordered and are stored in array X_I . Section III deals with the selection of the first real part corresponding to the already ordered first imaginary part of the coefficient set. Either $X_R(0)$ or its two's complemented value $-X_R(0)$ is selected on the basis of a smaller number of 1s. A flag bit is asserted in case the complemented value is selected. Section IV chooses subsequent real parts or their two's complemented values corresponding to their imaginary parts on the basis of the minimum Hamming distance with the already ordered preceding real part. As an example, Table 1 depicts the coefficient sets obtained by application of our proposed order-based processing scheme, and the conventional order-based processing schemes, to a 32-point FFT processor. The others, ordered set is obtained by using the minimum Hamming distance approach on successive coefficients. The ordered set for our proposed scheme is obtained by following the procedure outlined in Fig. 2. A flag bit is stored along with the ordered coefficients to indicate the form of the real part of the coefficient.

The proposed scheme is most effective in the last stage of the signal flowgraph of the radix-2 decimation-in-time FFT algorithm, where the coefficient switching activity is most intense; the coefficients are different for each butterfly

Table 1: Description of the various order based processing schemes for a 32-point FFT processor as an example

Address	Coefficient set (real, imag)	16-bit quantised coefficient set (real, imag)	Conventional order-based processing schemes (ordered set) (real, imag)	Proposed order-based processing scheme (Ordered set) (flag, real, imag)
0000	1.0, 0.0	7fff,0000	0000,8000	1,8001,0000
0001	0.98, -0.19	7d89,e706	7641,cf04	0,0000,8000
0010	0.92, -0.38	7641,cf04	7d89,e706	0,7641,cf04
0011	0.83, -0.55	6a6d,b8e3	89be,cf04	1,7642,cf04
0100	0.71, -0.71	5a82,a57d	8276,e706	0,7d89,e706
0101	0.55, -0.83	471c,9592	e706,8276	1,7d8a,e706
0110	0.38, -0.92	30fb,89be	cf04,89be	1,e707,8276
0111	0.19, -0.98	18f9,8276	471c,9592	0,e706,8276
1000	0.0, -1.0	0000,8000	7ff,0000	1,cf05,89be
1001	-0.19, -0.98	e706,8276	18f9,8276	0,cf04,89be
1010	-0.38, -0.92	cf04,89be	30fb,89be	1,471d,9592
1011	-0.55, -0.83	b8e3,9592	b8e3,9592	0,471c,9592
1100	-0.71, -0.71	a57d,a57d	9592,b8e3	1,6a6e,b8e3
1101	-0.83, -0.55	9592,b8e3	6a6d,b8e3	0,6a6d,b8e3
1110	-0.92, -0.38	89be,cf04	a57d,a57d	0,a57d,a57d
1111	-0.98, -0.19	8276,e706	5a82,a57d	1,a57e,a57d

unlike other FFT stages as, shown in Fig. 1. Our investigations revealed a maximum power saving if the scheme is limited only to the last FFT stage. The hardware overheads to support ordering in the penultimate stages outweigh the power saving obtained.

3 Hardware architecture and its implementation

An ordered radix-2 FFT processor core, shown in Fig. 3, comprises the following components.

- (i) a butterfly module;
- (ii) two dual port RAMs for holding data at each FFT stage;
- (iii) a ROM for holding the fixed coefficients;

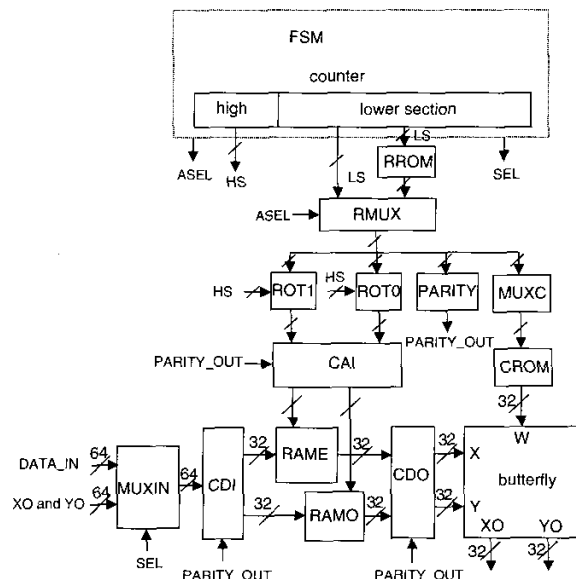


Fig. 3 Block diagram of an ordered FFT core

- (iv) coefficient and data address generation logic;
- (v) control logic in the form of a finite state machine (FSM); and
- (vi) a look up table (LUT) and a multiplexer to support order-based processing.

A butterfly module accepts complex data at every clock cycle in order to produce complex outputs during the same cycle. The in-place strategy (inputs and intermediate outputs are stored in the same memory location after every stage) is employed to reduce the memory size for an N -point transform to only N locations. Each memory location is 32-bits wide for storing both the 16-bit real and imaginary parts of the data. The memory is organised into two banks, based on parity of the address bits, in order to generate the addresses of the data required at successive FFT stages. The inputs are read from the RAMs (Both RAME and RAMO) into the butterfly in the same cycle as the outputs are written back to the RAMs for minimising the number of clock cycles [12]. The ROM block (CROM) is used to hold the fixed 16-bit FFT coefficients. The coefficient and data-address generation logic is based on a single counter realised as a part of the FSM block. This counter is partitioned into two sections. The more significant counter section (HS) keeps track of the different stages of the FFT whereas the lower significant section (LS) takes care of the butterfly operations within every stage. The coefficient address generation logic is based on a single multiplexer (MUXC) [13]. The data-address generation logic comprises rotation (ROT0 and ROT1) and parity generator (PARITY) blocks, along with interchange blocks for address (CAI) and data (CDI and CDO) to locate the proper address and data for the dual port RAMs [12]. The address and data interchange blocks direct the addresses and data to the respective dual port RAMs on the basis of address-bits parity. The order-based processing block consists of an LUT (RRROM) and a multiplexer (RMUX). RRROM stores the addresses of the ordered coefficient set. These addresses will be active only in the last stage of the FFT when order-based processing has to be incorporated. RMUX is used to select between the

ordered addresses for the coefficient and data and the conventional addresses. The ordered addresses are used only in the last stage, whereas the conventional addresses are used in all the previous FFT stages. The select signal (ASEL) for this multiplexer can be generated by using the bits of the more significant counter section. A MUXIN block is used for loading data (DATA_IN) into the RAM as well as for loading processed data (XO and YO) at successive FFT stages. The select signal (SEL) for this block is also generated by the FSM. The FSM block controls the loading, processing and outputting of the FFT processor core.

The hardware overhead required to support the ordering scheme is in the form of an additional ROM (RROM) having $N/2$ locations with word-width equal to $\log_2(N/2)$, a 2:1 channel multiplexer (RMUX) and an array of 16 Ex-OR gates in the modified multiplier module for the real coefficients only. It is clear that only the ROM size and the word-width, to a lesser extent, increase with FFT size. The ROM block consumes much less power and hence only introduces a small overhead as compared to the conventional approach. The ROM and multiplexer blocks are also required in the conventional ordering approaches and hence the only extra hardware overhead of the proposed approach compared to conventional approaches is the insignificant Ex-OR gates within the multiplier module. The butterfly module will be discussed in the next Section.

3.1 Butterfly module

A butterfly of a radix-2 decimation-in-time FFT has two complex inputs namely 'x' ($x_r + jx_i$) and 'y' ($y_r + jy_i$) and two complex outputs 'x_o' ($x_{ro} + jx_{io}$) and 'y_o' ($y_{ro} + jy_{io}$). The complex FFT coefficients are represented by 'w' ($w_r + jw_i$). The butterfly outputs and inputs are related by the following equations:

$$x_{ro} = x_r + (y_r w_r - y_i w_i) \quad (2)$$

$$x_{io} = x_i + (y_i w_r + y_r w_i) \quad (3)$$

$$y_{ro} = x_r - (y_r w_r - y_i w_i) \quad (4)$$

$$y_{io} = x_i - (y_i w_r + y_r w_i) \quad (5)$$

The block diagram of the butterfly hardware is shown in Fig. 4. A dedicated multiplication module has to be used for the real part of the coefficient (w_r) instead of a two's complement multiplier to support our order-based processing scheme. This will be discussed in the next Section.

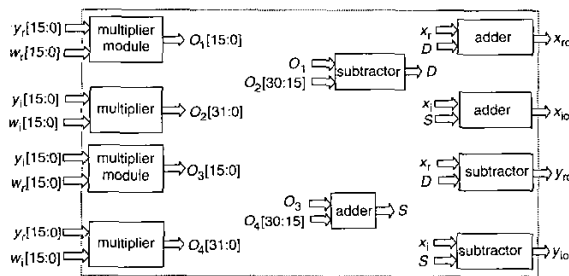


Fig. 4 Hardware block diagram of a butterfly module

3.2 Multiplication module

A simple two's complement multiplier cannot be used for the real coefficients because the real coefficients are used in two's complement form as well. A flag bit is also stored along with each real coefficient to indicate the form of the real coefficient.

The multiplication module corresponding to the real coefficients is obtained by adding 16 control inverters (Ex-Ors) to the output of a two's complement multiplier shown in Fig. 5. The flag bit controls the Ex-Ors and hence the final output (FO) of the multiplication module. The FO is the complement of the 16-bit multiplier output when the flag bit is '1', otherwise it will be the same as the multiplier output. The multiplier output is limited to only 16 bits instead of 32 bits. There is a difference of unity at the LSB position between the FO and the actual multiplier's 16-bit output in the conventional approach. This difference arises only in rare cases, where all the lower 15 bits of the actual multiplier's 32-bit output are zero and when the real coefficient is represented in its two's complement form. This is because only the upper 16-bit output of the multiplier is complemented instead of two's complementing the whole 32-bit output. This is performed to avoid the use of a 32-bit adder to generate the two's complement of the multiplier output, thereby saving power. Moreover, it does not lead to any error because the 16-bit output of the butterfly at each FFT stage is halved to avoid overflow. This approach has been verified extensively by comparing the outputs of different length FFTs with random data. The hardware overhead in the form of 16 Ex-OR gates in the multiplication module is insignificant when compared to the power saving obtained, due to the reduction in switching activity at the coefficient input of the module.

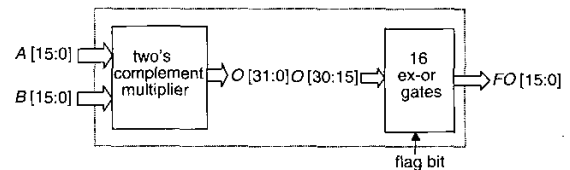


Fig. 5 Multiplier block for the real part of the coefficient

4 Implementation and results

A number of FFT cores of varying sizes have been designed at the register transfer level (RTL) using Verilog hardware description language. The cores were synthesised using Synopsys *DesignCompiler* with the 0.35μ Alcatel MTC45000 CMOS technology library. In order to evaluate the performance of the synthesised cores, netlist simulations were performed using Cadence's Verilog-XL simulator for 1000 FFT blocks of data samples. The switching activity information obtained from the netlist simulations was then fed into Synopsys *DesignPower* for power analysis. The power analysis was performed at a clock frequency of 10 MHz and a supply voltage of 3.3 V.

Table 2 lists the switching-activity reductions obtained by following the conventional and the proposed order-based processing schemes. It is clear from the Table that the switching activity reduction is close to 50% for the proposed scheme compared to only around 20% for the other schemes. Table 3 depicts the power consumption comparison for the different FFT cores, obtained by following the conventional and proposed approaches for the non-Booth-coded Wallace tree-type multiplier. It is evident from the Table that the power saving of our proposed scheme ranges from 1% to 25% for 512-point to 16-point FFT processors, respectively, over the conventional approaches. The percentage power savings continues to reduce for longer length FFTs because our scheme is directed at reducing power by lowering the switching activity at the coefficient inputs of the multipliers in the

Table 2: Switching activity comparison of different schemes

FFT size (points)	Total coefficient switching activity	Conventional order-based processing scheme		Proposed order-based processing scheme	
		Switching activity	Reduction (%)	Switching activity	Reduction (%)
16	126	120	05	68	46
32	240	204	15	126	48
64	476	368	23	222	53
128	828	672	19	424	49
256	1520	1196	21	780	49
512	2874	2106	27	1492	48
1K	5250	3934	25	2786	47

Table 3: Power consumption comparison of FFT cores

FFT size in (points)	Conventional FFT processor (mW)	Conventional order-based processing approach based FFT processor (mW)	Proposed order-based processing approach based FFT processor (mW)	Power reduction of proposed scheme over conventional order-based scheme (%)
16	67.6	68.44	51.49	25
32	84.55	86.82	75.68	13
64	104.93	105.88	95.72	10
128	213.28	210.34	198.89	7
256	355.27	354.74	343.80	3
512	864.35	863.80	855.21	1

Table 4: Power consumption of the different cells of the 32-point FFT cores along with net switching power

FFT Core Cells	Conventional FFT processor in (mW)	Conventional order-based processing approaches based FFT processor (mW)	Proposed order-based processing approach based FFT processor (mW)
Butterfly	22.4283	23.0468	17.9938
RAME	4.2294	4.2741	4.2573
RAMO	4.2213	4.2625	4.2478
MUXIN	1.2402	1.2656	1.2140
CDI	0.6959	0.7056	0.7080
CDO	0.3591	0.3717	0.3694
FSM	0.0472	0.0458	0.0447
RROM	—	0.0256	0.0336
CROM	0.0265	0.0301	0.0193
ROT1	0.0159	0.0134	0.0128
ROT0	0.0156	0.0166	0.0156
CAI	0.0145	0.0157	0.0158
MUXC	0.001862	0.002307	0.002313
RMUX	—	0.006518	0.006317
PARITY	0.005502	0.006419	0.005869
Internal cell power (IP)	33.301	34.089	28.947
Net switching power (NP)	51.2460	52.7295	46.7331
Total FFT core power TP = IP+NP	84.5472	86.8185	75.6801

butterfly structure. The butterfly complexity in an FFT remains fixed with the FFT length, whereas the RAM size goes on increasing. This means that the power consumed in the butterfly increases slightly with FFT length as compared to the power consumed in the RAM blocks. This results in a lowering of the percentage savings in power for longer FFTs. The conventional schemes lead to no power savings in most cases because the switching activity reduction is much less compared to the hardware overhead required to

support the scheme. This is not true for our proposed scheme because of the significant reduction in the switching activity of the ordered coefficient set. Table 4 lists the power consumed by the various blocks of the FFT processor core. It can be concluded from the Table that the butterfly and the RAM contribute most to the power consumption. The ROM and other modules consume a lot less power. The proposed scheme leads to power savings both in the internal cells as well as on the nets.

5 Conclusion

The paper has presented a novel order-based processing scheme capable of reducing the switching activity of the coefficient set by 53%. The significant reduction in switching activity leads to considerable power savings in various successfully implemented FFT cores of different lengths. The power savings range from 25% to 1% for 16-point to 512-point FFT processor cores, respectively, over the conventional approaches. Our scheme is more suitable for shorter length FFTs where the contribution of butterfly power to the overall power of the FFT core is much more than the power consumed by the RAMs.

6 References

- 1 Abu-Khater, I.S., Bellaouar, A., and Elmasry, M.I.: 'Circuit techniques for CMOS low-power high-performance multipliers', *IEEE J. Solid-State Circuits*, 1996, **31**, (10), pp. 1535-1546
- 2 Rabiner, L.R., and Gold, B.: 'Theory and application of digital signal processing' (Prentice-Hall, Englewood Cliffs, NJ, 1975)
- 3 Baas, B.M.: 'A low-power, high-performance, 1024-point FFT Processor', *IEEE J. Solid-State Circuits*, 1999, **34**, (3), pp. 380-387
- 4 Jia, L., Gao, Y., Isoaho, J., and Tenhunen, H.: 'A new VLSI oriented FFT algorithm and implementation'. Proceedings of the 11th IEEE International ASIC conference, 1998, pp. 337-341
- 5 Jia, L., Gao, Y., Isoaho, J., and Tenhunen, H.: 'Implementation of a low-power 128-point FFT processor'. Proceedings of the 5th International conference on Solid-state and integrated circuit technology, 1998, pp. 369-372
- 6 Stevens, K.S., and Suter, B.W.: 'A mathematical approach to a low-power FFT Architecture'. Proceedings of IEEE Int. Symp. Circuits Syst., 1998, pp. II-21-II-24
- 7 Hunt, B.W., Stevens, K.S., Suter, B.W., and Gelosh, D.S.: 'A single chip low-power asynchronous implementation of an FFT algorithm for space applications'. Proceedings of the 4th International symposium on Advanced research in asynchronous circuits and systems, 1998, pp. 216-223
- 8 Liang, J., and Tran, T.: 'Fast multiplierless approximations of the DCT with the lifting scheme', *IEEE Trans. Signal Process.*, 2001
- 9 Masselos, K., Merakos, P.K., Stouraitis, T., and Goutis, C.E.: 'Novel techniques for bus power consumption reduction in realizations of sum-of-product computations', *IEEE Trans. Very large Scale Integr. (VLSI) Syst.*, 1999, **7**, (4)
- 10 Masselos, K., Theoharis, S., Merakos, P.K., Stouraitis, T., and Goutis, C.E.: 'A novel methodology for power consumption reduction in a class of DSP algorithms'. Proceedings of IEEE Int. Symp. Circuits Syst., 1998, pp. VI-199-VI-202
- 11 Erdogan, A.T., Hasan, M., and Arslan, T.: 'A low-power FIR filtering core'. Proceedings of the 14th IEEE International ASIC/SOC Conference, 2001, pp. 271-275
- 12 Cohen, D.: 'Simplified control of FFT hardware', *IEEE Trans. Acoust. Speech Signal Process.*, 1976, pp. 577-579
- 13 Hasan, M., and Arslan, T.: 'A coefficient addressing scheme for FFT processors', *Electron. Lett.*, 2001, **37**, (22), pp. 1322-1324