

investigations show that the splitting of the output low level in Fig. 2b is due to the high sensitivity of the circuit to the intermediate input voltage. This splitting vanishes if the supply voltage is increased to $V_{CE} = V_{DD} = 6V$ which is possible because of the high breakdown voltage of the RTBT.

It should be emphasised that an XNOR function is also possible if the input low level is $V_{BE} = 0.8V$ and the input high level is $V_{BE} = 2.4V$ with the same supply voltage as for the NAND circuit. Additionally, the XOR function can be achieved if the input low is $V_{BE} = 1.6V$ and the input high is $V_{BE} = 3.6V$ with the same supply voltage as for the NOR gate.

By optimisation of the RTBT characteristics, common input and output levels can be achieved and cascading of the logic gates becomes possible. For this purpose the peak and valley voltage of the DB-RTD within the RTBT has to be reduced. Then complex logic functions with a drastically reduced number of active elements could be fabricated more easily.

In summary, we have presented ambivalent basic logic NAND and NOR gates and have also shown the potential for XNOR/XOR gate functionality using a single active device and fixed circuit topology. The measured low frequency transient behaviour of the circuit is in good agreement with the simulation results of the HP-MDS simulator which were obtained to facilitate designing the circuit. Future work will be devoted to achieving common input and output logic levels which can be realised by modifying the RTBT characteristics, enabling the development of more complex logic modules. By using a multiple base RTBT instead of the resistive voltage divider the coupling of the input terminals can be avoided.

© IEE 1998

8 October 1998

P. Velling, G. Janssen, U. Auer, W. Prost and F.J. Tegude (Gerhard-Mercator-University Duisburg, Solid-State Electronics Department, Lotharstrasse 65, D-47057 Duisburg, Germany)

E-mail: Guido.Janssen@hlt.uni-duisburg.de

References

- CAPASSO, F., SEN, S., BELTRAM, F., LUNARDI, L.M., VENGURLEKAR, A.S., SMITH, P.R., SHAH, N.J., MALIK, R.J., and CHO, A.Y.: 'Quantum functional devices: Resonant-tunneling transistors, circuits with reduced complexity, and multiple-valued logic', *IEEE Trans.*, 1989, **ED-36**, (10), pp. 2065-2082
- MAZUMDER, P., KULKARNI, S., BHATTACHARYA, M., SUN, J.P., and HADDAD, G.I.: 'Digital circuit applications of resonant tunneling devices', *Proc. IEEE*, 1998, **86**, (4), pp. 664-686
- SEABAUGH, A.C., TADDIKEN, A.H., BEAM, E.A. III, RANDALL, J.Y., KAO, Y.C., and NEWELL, B.: 'Room temperature resonant tunnelling bipolar transistor XNOR and XOR integrated circuits', *Electron. Lett.*, 1993, **29**, (20), pp. 1802-1803
- CHEN, K.J., WAHO, T., MAEZAWA, K., and YAMAMOTO, M.: 'Programmable logic gates based on controlled quenching of series-connected negative differential resistance devices'. Proc. IEEE Conf. DRC 54th, 24-26 June 1996, Santa Barbara, California
- VELLING, P., PROST, W., JANSEN, G., AUER, U., AGETHEN, M., REUTER, R., and TEGUDE, F.J.: 'On the design and modelling of novel 3-D integrated RTD/HBT'. Proc. IEEE Workshop on High Performance Devices for Microwave and Optoelectronic Applications (EDMO), Leeds, UK, 25-26 November 1996
- SEABAUGH, A.C., BEAM, E.A., TADDIKEN, A.H., RANDALL, J.N., and KAO, Y.-C.: 'Co-integration of resonant tunneling and double heterojunction bipolar transistors on InP', *IEEE Electron Device Lett.*, 1993, **14**, (10), pp. 472-474
- AUER, U., PROST, W., JANSEN, G., AGETHEN, M., REUTER, R., and TEGUDE, F.J.: 'A novel 3-D integrated RTD-HFET frequency multiplier', *IEEE J. Sel. Topics Quantum Electron.*, 1996, **2**, (3), pp. 650-654 (Special Issue on Ultra Fast Electronics, Optoelectronics, and Photonics)

Low power DCT implementation approach for CMOS-based DSP processors

S. Masupe and T. Arslan

An algorithm is presented for the low power implementation of the discrete cosine transform on single multiplier CMOS DSPs. The algorithm reduces power by reducing the amount of switched capacitance within the multiplier section of the DSP. This is achieved by a combination of using shift operations where possible, and manipulating bit-correlation between successive cosine coefficients applied to the input of the multiplier section. It is shown with a number of examples that up to 50% power saving can be achieved and that the algorithm provides a potential for more savings in power.

Introduction: Recently there has been a considerable interest in the low power implementation of the discrete cosine transform (DCT). This is mainly due to the DCT being the computational bottleneck of standards such as JPEG and MPEG [1]. Most research work considering the low power implementation of the DCT has targeted reducing the computational complexity of its design or modifying this for operation under a lower supply voltage [1, 2]. Both these techniques have a limited effect on power reduction. Another major contribution to power consumption is due to the effective switched capacitance [3]. Only a few researchers have targeted reducing the power of a DCT implementation through a reduction in the amount of switched capacitance. This reduction has been achieved through techniques such as the detection of zero-valued DCT coefficients [6], and lookup table partitioning [7]. This Letter presents a new technique for reducing the power dissipation of the DCT by targeting the multiplier section of a DCT processor. The pivot of this technique is a multiplication algorithm, which reduces power consumption by reducing the effective switched capacitance of the multiplier through effective manipulation of the multiplication process between the cosine and data matrices. Our results indicate that the effective capacitance can be reduced significantly by performing the multiplications of the DCT in an order dictated by the amount of correlation between subsequent cosine matrix elements applied to one of the inputs of the multiplier section. In addition we propose a modified processor architecture that accommodates this multiplication algorithm and is open to exploitation by other ordering algorithms. We demonstrate our algorithm through an example in which the cosine coefficients are ordered according to the minimum Hamming distance, revealing up to 50% power savings with a number of image examples.

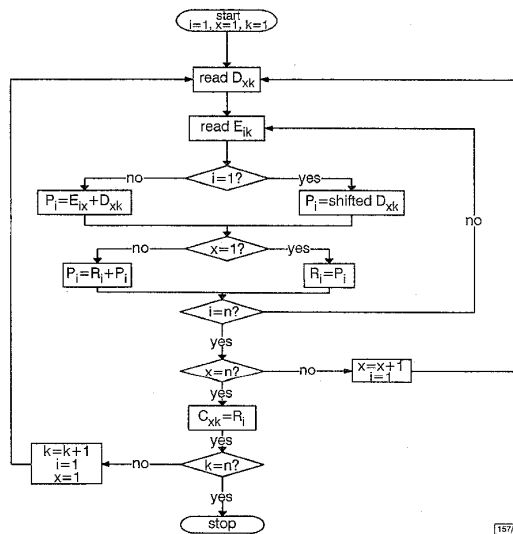


Fig. 1 Flowchart of algorithm

Implementation: The computational bottleneck for the implementation of the DCT is the multiplication of the cosine coefficients matrix [E], by the pixel matrix [D], in order to obtain the cosine coefficients [C], i.e.

$$[C] = [E] * [D] \quad (1)$$

where each element C_{ij} in a $[C]$ matrix of order n is given by:

$$C_{ij} = \sum_{k=1}^n E_{ik} D_{kj} \quad (2)$$

Traditionally, the multiplication process is performed in a row-by-column fashion [4], see eqn. 2. In some applications where the number of multipliers is limited, this implies new data at both multiplier inputs. This in turn implies a relatively large switching activity inside the multiplier circuit.

We carried out a number of experiments with various cosine and pixel matrices. Careful analysis of the multiplication procedure revealed two distinct categories of cosine elements: (i) elements that are powers of 2, and (ii) those which are non-powers of 2 numerics. For this reason, our proposed algorithm processes the elements in (i) by performing a simple shift operation. It is well known that the switched capacitance of a shift is significantly less than that of a multiplication [3].

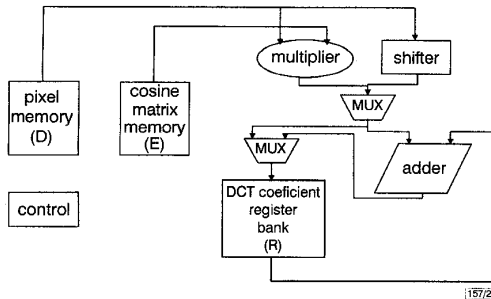


Fig. 2 Simplified architecture of processor

For each column being processed, the multiplication procedure outlined above has the product form ($m \times \text{constant}$), where m represents the cosine matrix elements being processed. The multiplication results are unchanged irrespective of the order in which the constant is multiplied by the sequence of cosine coefficients in the column. For this reason, our algorithm performs the multiplication of elements in eqn. 2 in a column-by-column fashion and orders the elements, in the column being processed, according to some criteria, in our case the minimum switching activity (Hamming distance) between consecutive coefficients multiplied by the constant. This guarantees that similar elements are subsequently applied to the inputs of the multiplier causing minimum switching activity within the internal circuit of the multiplier. Although our investigations were carried out using the example of ordering elements according to minimum Hamming distance, in practice, any ordering algorithm can be used and the amount of power saving is determined by the power of the algorithm used. The steps in Fig. 1 outline the algorithm, which commences with the following initialisation steps: (i) Process entries E_{ix} with shift operation; (ii) order remaining coefficients according to some ordering algorithm; (iii) save entries E_{ix} in (E) memory, see Fig. 2, with elements of the same column being adjacent to each other, followed by the next column, and so on.

To illustrate the above algorithm, consider an example where

$$E = \begin{bmatrix} 2.00 & 2.00 & 2.00 & 2.00 \\ 1.85 & 0.77 & -0.77 & -1.85 \\ 1.41 & -1.41 & -1.41 & 1.41 \\ 0.77 & -1.85 & 1.85 & -0.77 \end{bmatrix}$$

$$\text{and } D = \begin{bmatrix} 35 & 32 & 31 & 34 \\ 32 & 37 & 31 & 31 \\ 29 & 28 & 27 & 31 \\ 26 & 28 & 31 & 34 \end{bmatrix}$$

After the first iteration ($i = 1$, $x = 1$ and $k = 1$), the first column, E_{i1} is ordered according to minimum Hamming distance to produce the ordered sequence (2.00, 0.77, 1.41, 1.85). The original locations of these elements are stored. Next the above sequence is multiplied by the first entry in D_{x1} , 35. The entries in registers R_i will be (70, 64.75, 49.35, 26.95), where the first entry (70) is saved in R_1 and the second entry is saved in R_2 and so on. Similarly at

the end of iteration 2 ($i = 2$, $x = 1$, and $k = 1$), R_i will contain (134, 89.39, 4.23, -32.25). The last iteration ($i = 4$, $x = 1$, and $k = 1$) for this particular column results in R_i containing the first column of the DCT coefficient matrix $[C]$, i.e. $R_i = C_{x1} = (244, 18.96, 0, 1.35)^T$. This procedure is carried out until $x = k = n$, at which case R_i will contain C_{44} .

As it stands the algorithm can be implemented on traditional DSPs without any loss in throughput. However, for higher throughput applications, a modified processor architecture is required. The architecture, a simplified version of which is shown in Fig. 2, requires an internal register bank in order to store the partial products (R), which eventually result in the DCT coefficients $[C]$. In addition, a special memory unit is allocated for both the cosine and the pixel matrix elements, E_{ix} and D_{xk} , respectively. A shifter is included to cope with the additional shift operations.

Table 1: Typical power savings

Image	Ordering	Switched capacitance	Switched capacitance reduction
Horizontal stripes	Traditional	$\text{pF}(\times 10^3)$ 13.77	% -
	Column-based	8.45	38.64
	Hamming	7.06	48.73
Vertical stripes	Traditional	7.70	-
	Column-based	7.55	1.97
	Hamming	6.41	16.84
Blocks	Traditional	9.25	-
	Column-based	7.65	17.27
	Hamming	6.47	29.99
Checked	Traditional	13.81	-
	Column-based	8.39	39.25
	Hamming	6.83	50.50

Simulation and results: An 8×8 bit array multiplier was constructed using the Cadence VLSI suite with ES2 0.7 CMOS technology. The cosine coefficient matrix used was obtained using the MATLAB signal processing toolbox. This was scaled so that the entries could be represented by numbers between -128 and 127. The pixels were chosen to represent several types of scenarios including checked and striped images. A C-program was developed to generate input simulation files for the Verilog-XL™ digital simulator [5]. Three types of input simulation files are generated, each representing the use of one of the following: (i) traditional cosine DCT multiplication; (ii) Proposed column-based multiplication algorithm without ordering (column-based); (iii) proposed column-based multiplication algorithm with ordering according to minimum Hamming distance (Hamming). In each simulation, the number of signal transitions (switching activity) for each gate is monitored. Capacitive information for each gate is extracted from the layout of the multiplier circuit. Both of these are used to obtain a figure for the total switched capacitance of the multiplier.

Table 1 shows the results of the different scenarios used to test the algorithm. As the Table shows, power savings are achieved with all examples, with a maximum of 50% power saving for an 8bit multiplier.

Conclusions: An algorithm for the low power implementation of the discrete cosine transform on CMOS based DSPs has been presented. The algorithm reduces power by manipulating data at the multiplier inputs such that switching activity is reduced. This in turn has the effect of reducing the total switched capacitance. It has been shown that the algorithm can be used to obtain up to 50% power reduction when used in conjunction with an ordering algorithm which manipulates bit correlation at multiplier inputs.

References

- 1 PAO, I., and SUN, M.: 'Computation reduction for discrete cosine transform'. Int. Symp. Circuits and Systems, Monterey, CA, USA, 31 May - 3 June 1998
- 2 CHEN, L., *et al.*: 'Low power 2D DCT chip design for wireless multimedia terminals'. Int. Symp. Circuits and Systems, Monterey, CA, USA, 31 May - 3 June 1998
- 3 ERDOGAN, A.T., and ARSLAN, T.: 'Data block processing for low power implementation on single multiplier CMOS DSP processors'. Int. Symp. Circuits and Systems, Monterey, CA, USA, 31 May - 3 June 1998
- 4 BHASKARAN, V., and KONSTANTINIDES, K.: 'Image and video compression standards: Algorithms and architectures' (Kluwer Academic Publishers, 1997)
- 5 THOMAS, D.E., and MORBY, P.R.: 'The Verilog hardware description language' (Kluwer Academic Publishers, 1995)
- 6 XANTHOPOULOS, T., and CHANDRAKASAN, A.: 'Low-power IDCT macrocell for MPEG MP@ML exploiting data distribution properties for minimal activity'. Proc. 1998 Symp. VLSI Circuits, Honolulu, HI, USA, 11-13 June 1998
- 7 CHO, S., *et al.*: 'Ultra low power variable length decoder for MPEG-2 exploiting codeword distribution'. Proc. 1998 IEEE Custom Integrated Circuits Conf., Santa Clara, CA, USA, 11-14 May 1998,

MRF model based image segmentation using hierarchical distributed genetic algorithm

Hang Joon Kim, Eun Yi Kim, Jin Wook Kim and Se Hyun Park

An unsupervised method for segmenting noisy and blurred images is proposed. A Markov random field (MRF) model is used which is robust to degradation. Since this is computationally intensive, a hierarchical distributed genetic algorithm (HDGA) is used which is unsupervised and parallel. Experimental results show that the proposed method is effective at segmenting real images.

Introduction: Image segmentation is the process by which an image is segmented into a group of homogeneous regions according to characteristics such as colour and texture. It is the front-end processing stage in image processing systems. We use a Markov random field (MRF) model which is robust to degradation, but is computationally intensive [1, 2, 5]. The parameter estimation problem is a crucial issue for methods based on MRF, and their performance depends on the availability of correct parameter estimates. A more complex and powerful segmentation algorithm is therefore required. In this Letter, we use a hierarchical distributed genetic algorithm (HDGA) which is unsupervised and parallel. The HDGA uses a multilevel search strategy. The HDGA provides support for complicated problems with large numbers of parameters. To analyse segmentation results, we use the uniform evaluation criterion proposed in [6].

Problem statement: We regard the original image as a pair $X = (F, L)$, where F is the matrix of observable pixel colours and L denotes a matrix of unobservable label processes. We consider an input image to be degraded by blurring matrix H and Gaussian white noise N . Let $S = \{(i, j) : 1 \leq i \leq m, 1 \leq j \leq n\}$ denote the $m \times n$ lattice, such that the elements in S index the image pixels. We can define a neighbourhood on S as $\Gamma = \{\eta_{i,j}\}$ where $\eta_{i,j}$ is a set of sites neighbouring (i, j) . Let $\Lambda = (\lambda_1, \dots, \lambda_n)$ denote the label set and $X = \{X_{i,j}\}$ be a family of random variables defined on the set S , in which each random variable $X_{i,j}$ takes a value $X_{i,j} \in \Lambda$. X is an MRF on S with respect to a neighbourhood system because the two conditions of [5] are satisfied. Our goal is to find the ω which maximises the posterior distribution for a fixed input image g . That is, we want to determine

$$\arg \max P(X = \omega | g) = \arg \max P(g | \omega) P(\omega) / P(g) \quad (1)$$

Eqn. 1 is divided into two components of the following:

$$\max P(g | \omega) = \max (N = g - H(x) | \omega)$$

$$= \max \left\{ \exp(-\|\mu - (g - H(f))\|^2 / 2\sigma^2) \left(\sqrt{2\pi\sigma^2} \right)^{-mn} \right\} \quad (2)$$

$$\max P(\omega) =$$

$$\max \left\{ \exp(-U(f, l)) = \max \{ \exp(-U(f|l) + U(l)) \} \right.$$

$$\left. = \max \exp \left(- \left(\sum_C V_c(f|l) + \sum_C E_c(l) \right) \right) \right\} \quad (3)$$

In eqn. 3, C is a possible first-order clique set in the neighbourhood, and V and E are potential functions defined as follows:

$$V_c(f|l) = \begin{cases} -1 & L(ch_i) = L(ch_j) \text{ and } F(ch_i) = F(ch_j) \\ 1 & L(ch_i) = L(ch_j) \text{ and } F(ch_i) \neq F(ch_j) \\ 0 & L(ch_i) \neq L(ch_j) \end{cases}$$

$$E_c(l) = \begin{cases} -1 & \text{if labels of pixels in a clique are same} \\ 1 & \text{otherwise} \end{cases}$$

where L represents the label number of a chromosome ch and F represents the colour feature of a chromosome. Eqn. 1 is represented by the following equation, which is defined as an energy function:

$$\min \left\{ \sum_C V_c(f|l) + \sum_C E_c(l) + \frac{\|\mu - (g - H(f))\|^2}{2\sigma^2} + \frac{1}{2} \log(2\pi\sigma^2) \right\}$$

where μ and σ are the average and variance of the noise, respectively. Instead of maximising the posterior distribution, we minimise the energy function.

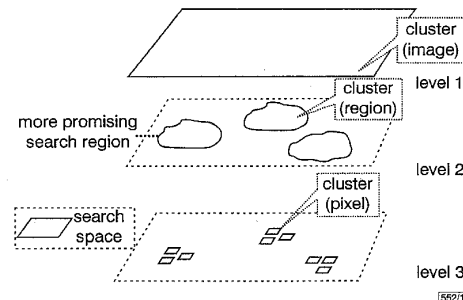


Fig. 1 Structure of HDGA

Structure of HDGA: We use the HDGA to minimise the energy function. The HDGA uses a multilevel search strategy illustrated in Fig. 1, in which a higher level cluster investigates a wide search space [3]. The HDGA facilitates an expanding search parameter approach. The first level only searches the colour features of each pixel then these features are sent to the second level where optimal blurring matrices of each region are searched. Note that in this level only the blurring matrices are manipulated. Finally, these parameters are sent to the higher level where optimal noise parameter values are searched.

A chromosome consists of four parts: label, colour features, blurring matrix and noise. The first and second layers of the HDGA optimise the energy function using a distributed genetic algorithm [4]. We assume that the blurring effect on all pixels in a region is equal. The third layer carries out optimisation with the noise parameter values using a simple genetic algorithm. For the stopping criterion we define the equilibrium. At each time step, the equilibrium is given by the percentage of pixels in the segmented image that are assigned the label they had at the previous time step. The stopping criterion is reached when the equilibrium is above the equilibrium threshold, or when the maximal number of generations has been reached.

Experimental results: To evaluate the performance of the proposed method, experiments were performed using 50 images. The maximum number of labels is 16 and the size of the blurring matrix is 3×3 . The neighbourhood of a point is made of the eight contiguous