

Discussion and conclusion: An explanation of the experimental results has to consider a superposition of the avalanche and drift mechanisms. This is obvious from a numerical simulation of a double drift diode at a current density of 200 kA/cm². An electric field distribution is obtained from solutions of the basic semiconductor equations which is much more like the static distribution in a *pin* diode than in a CW double drift diode. The maximum current density in *pin* diodes is limited by the breakdown of the electric field in the centre of the diode caused by the enormous number of injected carriers. This can be avoided by the introduction of a *p* and an *n* doped layer which simply leads to the flat profile double drift diode. In conclusion, the experimental results can be explained with a distributed avalanche and drift mechanism in the whole diode as originally suggested by Misawa for the *pin* diode.⁶

Acknowledgment: I would like to thank L. Gaul and M. Claassen from the Technical University of Munich for valuable discussions.

J.-F. LUY

21st September 1990

Daimler Benz Research
Wilhelm Runge Straße 11
D-7900 Ulm, Germany

References

- 1 BEHR, W., and LUY, J. F.: 'High power operation mode of pulsed IMPATT diodes', *IEEE Electron. Dev. Lett.*, 1990, **EDL-11**, pp. 206-208
- 2 WENGER, J., FREYER, J., and HARTH, W.: 'Pulsed and CW *n*-type silicon single-drift IMPATT diodes for 140 GHz'. 16th EuMC, Dublin, 1986, pp. 573-577; WENGER, J.: PhD Dissertation, Technical University of Munich, 1987
- 3 GLOVER, G. H., and TANTRAPORN, W.: 'Doping profile measurements from avalanche space-charge resistance: A new technique', *J. Appl. Phys.*, 1975, **46**, pp. 867-874
- 4 GRANT, W.: 'Electron and hole ionization rates in epitaxial silicon at high electric fields', *Solid State Electron.*, 1973, **16**, pp. 1189-1203
- 5 WIERICH, R. L.: PhD Dissertation, University College of London, 1971
- 6 MISAWA, T.: 'Negative resistance in *p-n* junctions under avalanche breakdown conditions: Part I and II', *IEEE Trans.*, 1966, **ED-13**, pp. 137-151

VLSI COMPUTING ARCHITECTURES FOR HAAR TRANSFORM

Indexing terms: Signal/image processing, VLSI architecture

The Haar transform is very useful in many signal and image processing applications where real-time implementation is essential. Three VLSI computing architectures are proposed for fast implementation of the Haar transform. Comparisons on the advantages and disadvantages of the proposed architectures are also presented.

Introduction: The Haar transform is based on the Haar functions which are periodic, orthogonal, and complete.¹ The Haar functions become increasingly localised as their number increases.^{1,2} The Haar transform provides a transform domain in which a type of differential energy is concentrated in localised regions.² This kind of property is very useful in image processing applications such as edge detection and contour extraction.²⁻⁵ The first order Haar matrix is defined as

$$H(1) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (1)$$

1962

and the higher order Haar transform can be generalised recursively as follows:

$$H(k+1) = \begin{bmatrix} H(k) \otimes (1 & 1) \\ 2^{k/2} I_{2^k} \otimes (1 & -1) \end{bmatrix} \quad (2)$$

where \otimes is the Kronecker product such that

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1N}B \\ a_{21}B & a_{22}B & \dots & a_{2N}B \\ \vdots & \vdots & \dots & \vdots \\ a_{N1}B & a_{N2}B & \dots & a_{NN}B \end{bmatrix}$$

where a_{ij} is the element of matrix A and I_{2^k} is an identity matrix of dimension 2^k . As an example, the second and third order Haar matrices are given below

$$H(2) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

$H(3) =$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$

A rationalised Haar transform can be deduced by deleting the irrational number and introducing integer powers of 2.¹ It is then generalised as follows:

$$H(k+1) = \begin{bmatrix} H(k) \otimes (1 & 1) \\ I_{2^k} \otimes (1 & -1) \end{bmatrix} \quad (3)$$

For both cases, the signal flow graphs for fast implementation share the same structure as shown in Fig. 1, in which an example of Haar transform of dimension eight is given.

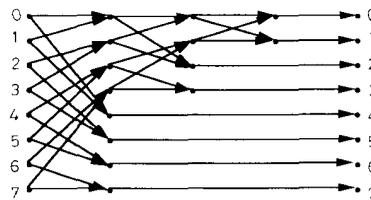


Fig. 1 Signal flow graph of Haar transform

VLSI architectures: In image and signal processing, real-time implementation is a crucial issue in many applications. The Haar transform has been applied to various image applications. Up to now, no VLSI architecture for practical implementation of the Haar transform has been considered. Three VLSI architectures are proposed for hardware implementation of the Haar transform.

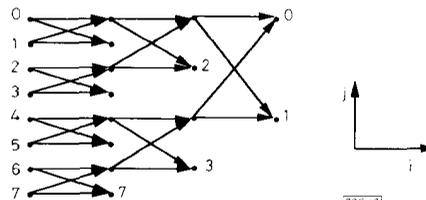


Fig. 2 Modified locally connected signal flow graph

Systolic tree architecture: In VLSI system design, local communication of processing cells is one of the basic requirements for implementation of a large scale system. The signal flow graph shown in Fig. 1 can be rearranged in a way so that only local communications are needed. The idea is to shuffle the output sequence so that no global communication is needed. The rearranged signal flow graph is shown in Fig. 2. It is a tree-like butterfly structure. Suppose each butterfly is replaced by the processing cell shown in Fig. 3, a systolic tree architecture for the Haar transform can be obtained as shown in Fig.

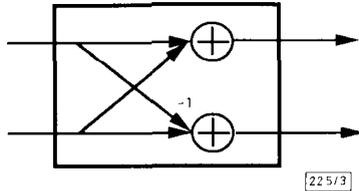


Fig. 3 Processing cell of systolic tree architecture

4. A total of $N - 1$ processing cells and $\log_2 N$ pipelined stages are required. This architecture offers excellent pipelina-bility. At the first stage, the output sequence $\{X_{N/2}, X_{N/2+1}, \dots, X_{N-1}\}$ are obtained; at the second stage, the sequences $\{X_{N/4}, X_{N/4+1}, \dots, X_{N/2-1}\}$ are obtained. In general, at the k th stage, the sequences $\{X_{N/2^k}, X_{N/2^k+1}, \dots, X_{N/2^{k-1}-1}\}$ are obtained. Since there are $\log_2 N$ pipelined stages, the time required to complete the computation is $\log_2 N$.

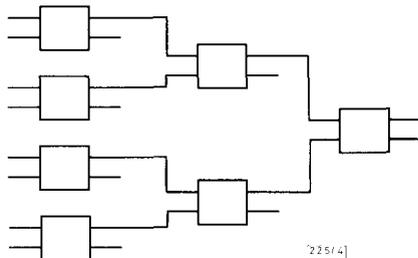


Fig. 4 Systolic tree architecture for Haar transform

Linear data flow array: A major disadvantage of the systolic tree architecture approach is that a total of $N - 1$ processing cells is required. Algorithm mapping is an effective way of obtaining various different architectures in VLSI implementation.^{6,7} Consider a projection of the signal flow graph in Fig. 2 onto the j direction, we obtain a linear data flow array as shown in Fig. 5. In this linear array, only $\log_2 N$ processing

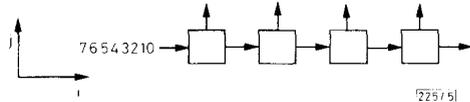


Fig. 5 Linear data flow array for Haar transform

cells are required. The processing cell is shown in Fig. 6. Each processing cell has two local registers and is data driven in the sense that when the two local registers are filled with new data, it fires (starts operations). The first cell fires 50% of the time; the second 25% and so on. It is an asynchronous system without any global clocking circuitry because of the data driven property. The first cell generates the output sequences $\{X_{N/2}, X_{N/2+1}, \dots, X_{N-1}\}$. In general, the k th cell generates sequences $\{X_{N/2^k}, X_{N/2^k+1}, \dots, X_{N/2^{k-1}-1}\}$. The time required to complete the computation is $N + \log_2 N$.

Sequential queue architecture: We can further map the linear array in Fig. 5 onto the i direction. The sequential queue architecture shown in Fig. 7 is obtained. In this architecture, only a processing cell and a first in/first out queue are used. The required maximum length of the queue is N . All the data are stored in the queue before the computation. The processing cell then fetches data from the queue. When the pro-

cessing cell has fetched two data into its local registers, it is driven to fire in a similar manner to the operations in the processing cell of the linear data flow array. An output is

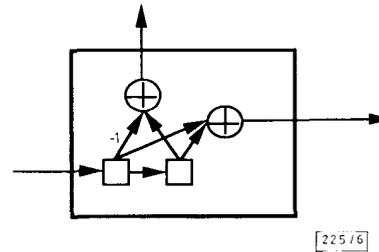


Fig. 6 Processing cell of linear data flow array

obtained and an intermediate result is fedback to the queue for further processing. The input sequences for both the sequential queue architecture and the linear data flow array are serial. The output sequences for the sequential queue architecture are $\{X_{N/2}, X_{N/2+1}, \dots, X_{N-1}, X_{N/4}, X_{N/4+1}, \dots, X_{N/2-1}, X_{N/8}, \dots, X_1, X_0\}$. The time required to complete the job is $N \log_2 N$.

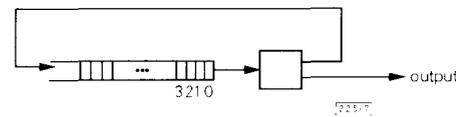


Fig. 7 Sequential queue architecture for Haar transform

Conclusions: Three VLSI architectures for fast implementation of the Haar transform were proposed. Each architectures may have certain advantages and disadvantages compared with the others. A summary of the properties of these three architectures is given in Table 1. The choice of one over

Table 1 COMPARISONS OF VLSI ARCHITECTURES FOR HAAR TRANSFORM

	Systolic tree archit.	Linear data flow array	Sequential queue architecture
Computation time	$\log_2 N$	$N + \log_2 N$	$N \log_2 N$
Number of cells	$N - 1$	$\log_2 N$	1
Input/output	Parallel	Serial/parallel	Serial
Pipelina-bility	Very good	Good	Bad

another is primarily determined by the application and the resources available in the design.

K. J. RAY LIU

18th September 1990

Electrical Engineering Department
Systems Research Center
University of Maryland
College Park, MD 20742, USA

References

- ELLIOTT, D. F., and RAO, K. R.: 'Fast transforms: algorithms, analyses and applications' (Academic Press, 1982)
- PRATT, W. K.: 'Digital image processing' (Wiley, 1978)
- LYNCH, R. T., and REIS, J. J.: 'Haar transform image coding'. Proc. National Telecommunication Conf., Dallas, 1976, pp. 44.3-1-44.3-5
- DIXIT, V. V.: 'Edge extraction through Haar transform'. Proc. 14th Asilomar Conf. Circuits and Syst. and Comput., Pacific Grove, 1980
- WENDLING, S., GAGNEUX, G., and STAMON, G.: 'Use of the Haar transform and some of its properties in character recognition'. Proc. Int. Conf. Pattern Recognition, Coronado, 1976, pp. 844-848
- MEAD, C., and CONWAY, L.: 'Introduction to VLSI systems' (Addison-Wesley, 1980)
- KUNG, S. Y.: 'VLSI array processors' (Prentice-Hall, 1988)