$$a_j^{\omega'} = \bigoplus_{k \subseteq_0 j' \oplus_d t} m_{k \oplus_d \omega} = \bigoplus_{k \subseteq_0 j'} m_{k \oplus_d \omega \oplus_d t} = a_{j'}^{\omega \oplus_d t} = a_{j \oplus_d t}^{\omega \oplus_d t} \tag{4}$$

For $j_p \neq j_q$, $k' \subseteq_0 j \in \{k' \in \mathbf{Z} | k'_p = 0 \text{ and } k'_q = - \text{ or } k'_q = 0 \text{ and } k'_p = -\}$. In either case, a_j^{ω} equal to a $GF(2)$ summation of an odd number of minterms $m_{k \oplus_d \omega}$ due to $k_p = k_q = 0$ and an odd number of minterms $m_{k \oplus_d \omega \oplus_d t}$ due to $k_p \neq k_q$. For those ω such that $\omega_p = \omega_q$, $m_{k \oplus_d \omega} \leftrightarrow m_{k \oplus_d \omega \oplus_d t}$, the $GF(2)$ summation in eqn. 4 remains unchanged. Therefore, $a_j^{\omega'} = a_j^{\omega}$ if $\omega_p = \omega_q$. In summary, when $\omega_p = \omega_q$, $A^{\omega'} = A^{\omega}$. When $\omega_p \neq \omega_q$,

$$a_j^{\omega'} = \begin{cases} a_{j \oplus_d t}^{\omega \oplus_d t} & \text{if } j_p \neq j_q \\ a_j^{\omega} & \text{if } j_p = j_q \end{cases}$$

Since each coefficient in polarity ω is mapped into a distinct coefficient in another polarity $\omega' = \omega \oplus_d t$, $W' = W$. □

Lemma 3: Complementing a function does not alter its w_l, but either increments or decrements its w_p by one, for all polarities of ω.

Proof of Lemma 3: This is trivial since $\overline{A^{\omega}} = 1 \oplus_d A^{\omega}$ for any ω. □

Conclusion: We have proven by using the subnumber operation that the Reed-Muller weight and literal vectors fully classify Boolean functions in an NP equivalent class and NPN equivalent class, respectively. The presented proof leads to applications in technology mapping and design with universal logic modules.

14 January 1999
Electronics Letters Online No: 19990549
DOI: 10.1049/el:19990549

Chip-Hong Chang and B.J. Falkowski (*Nanyang Technological University, School of Electrical and Electronic Engineering, South Spine, Block S1, Nanyang Avenue, Singapore 639798, Republic of Singapore*)

B.J. Falkowski: Corresponding author

E-mail: efalkowski@ntu.edu.sg

References

1 DAVIO, M., DESCHAMPS, J.P., and THAYSE, A.: 'Discrete and switching functions' (McGraw-Hill, New York, 1978)
2 DeMICHELI, G.: 'Synthesis and optimization of digital circuits' (McGraw-Hill, New York, 1994)
3 TSAI, C.C., and MAREK-SADOWSKA, M.: 'Boolean functions classification via fixed polarity Reed-Muller forms', *IEEE Trans.*, 1997, **C-46**, (2), pp. 173–186
4 CHANG, C.H., and FALKOWSKI, B.J.: 'Adaptive exact optimisation of minimally testable FPRM expansions', *IEE Proc. Comput. Digital Techniques*, 1998, **145**, (6), pp. 385–394
5 FISHER, L.T.: 'Unateness properties of AND-Exclusive-OR logic circuits', *IEEE Trans.*, 1974, **C-23**, (2), pp. 166–172

Relationship between arithmetic and Haar wavelet transforms in the form of layered Kronecker matrices

B.J. Falkowski

The relationship between the arithmetic and Haar wavelet transforms is investigated. An interesting connection in the form of layered vertical and horizontal Kronecker matrices between the two transforms is revealed. The new relations apply to an arbitrary dimension of the transform matrices and enable the arithmetic spectrum to be calculated directly from the Haar wavelet spectrum, and vice versa, without the necessity of re-obtaining the original function.

Introduction: In the preceding decade there has been increasing interest in applications of different discrete transforms to the digital signal processing of integer-valued and complex signals. Depending on the application, some transforms are better suited than others. It is often advantageous to apply more than one transform in a given application based on the local properties of a data signal. In the latter case it is of interest to investigate mutual relations between various local discrete transforms such as, for example, the Haar wavelet and arithmetic transforms. Both the Haar wavelet transform (non-normalised version of the transform where only signs are entered into the transform matrix) and the arithmetic transform have been used in many logic design applications [2, 3, 5, 6, 9]. Therefore it is not only interesting theoretically but also practical to state their mutual relations. Initial work in this area has recently been carried out where such mutual relations were presented for the transformation matrices of order 3 [4]. In this Letter, for the first time, such relations are presented for an arbitrary transform matrix order.

Arithmetic and Haar transforms: The matrix of order $N = 2^n$ for the arithmetic transform is defined as [2, 5, 6]

$$A_N = \begin{bmatrix} A_{\frac{N}{2}} & 0 \\ -A_{\frac{N}{2}} & A_{\frac{N}{2}} \end{bmatrix} \quad A_1 = 1 \quad N = 2, 3, \ldots$$
$$\text{Also } A_N = A_2 \otimes A_{\frac{N}{2}} \text{ for } N = 2, 3, \ldots$$

The non-normalised Haar transform H_N of order N = 2^n can be defined recursively as [1, 3, 8, 10]

$$H_N = \begin{bmatrix} H_{\frac{N}{2}} \otimes [1 \quad 1] \\ I_{\frac{N}{2}} \otimes [1 \quad -1] \end{bmatrix} \text{ and } H_1 = 1$$

where $I_{N/2}$ is an identity matrix of order $N/2$. In the above equations, the symbol '$\otimes$' denotes the Kronecker direct product [10].

For an n-variable Boolean function $F(x_1, x_2, \ldots, x_n)$, the Haar and arithmetic spectrum (a column vector of dimension $2^n \times 1$) is given by $H = [H_N]F$ and $A = [A_N]F$ where H is the Haar spectrum and A the Arithmetic spectrum, accordingly. From the above definitions it is obvious that the first two rows of $[H_N]$ are global basis functions $H_0(x)$ and $H_1(x)$, respectively. All subsequent rows comprise local basis functions $H_l^{(k)}(x)$ in ascending order of l and k. l = 1, 2, ... is the degree of the Haar function describing the number of zero crossings, and k = 1, ..., 2^l is the order of the Haar function describing the position of the subset l within a function. In the arithmetic transformation matrix $[A_N]$, all rows but the last comprise local basis functions.

Relations between Haar and arithmetic functions in form of layered Kronecker matrices: The following relations will be given only for arithmetic functions and non-normalised Haar wavelet functions. From earlier results [4] it is trivial to modify the presented equations for normalised Haar wavelet functions by adding normalising factors. Also, to make this Letter consistent with other published papers, Hadamard ordering is used for both transforms. A_i denotes an ith arithmetic function and the Haar functions follow the definition from the previous paragraph. Kaczmarz gave the definition of Walsh functions by Haar wavelet functions for the first eight functions (n = 3) [7] and similar developments for arithmetic and Haar wavelet functions were presented in [4]. Here, the definitions for higher n are shown for expressing arithmetic functions by non-normalised Haar functions and vice versa. To make the work applicable to the transformation matrix of any dimension, the presented relations are given in the form of layered Kronecker products.

Mutual relations between the Haar and arithmetic transforms for the general case of arbitrary n are as follows:

$$\begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_{12} \\ A_3 \\ A_{13} \\ A_{23} \\ A_{123} \\ \vdots \end{bmatrix} = \frac{1}{2^n} \begin{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & -2 \end{bmatrix} & \otimes & \left(\overset{n-1}{\otimes} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) & & & \vdots \\ \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} & \otimes & \left(2 \begin{bmatrix} 1 \\ -2 \end{bmatrix} \right) & \otimes & \left(\overset{n-2}{\otimes} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) & \vdots \\ \left(\overset{2}{\otimes} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \right) & \otimes & \left(4 \begin{bmatrix} 1 \\ -2 \end{bmatrix} \right) & \otimes & \left(\overset{n-3}{\otimes} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) & \vdots \\ \left(\overset{3}{\otimes} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \right) & \otimes & \left(8 \begin{bmatrix} 1 \\ -2 \end{bmatrix} \right) & \otimes & \left(\overset{n-4}{\otimes} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) & \vdots \\ \vdots & & \vdots & & \vdots & \end{bmatrix} \begin{bmatrix} H_0 \\ H_1 \\ H_2^{(1)} \\ H_2^{(2)} \\ H_3^{(1)} \\ H_3^{(2)} \\ H_3^{(3)} \\ H_3^{(4)} \\ \vdots \end{bmatrix}$$

$$
\begin{bmatrix} H_0 \\ H_1 \\ H_2^{(1)} \\ H_2^{(2)} \\ H_3^{(1)} \\ H_3^{(2)} \\ H_3^{(3)} \\ H_3^{(4)} \\ \vdots \end{bmatrix}
=
\left[\begin{array}{ccccc}
\begin{bmatrix} 2 & 1 \\ 0 & -1 \end{bmatrix} & \otimes & \left(\overset{n-1}{\otimes}[2\ 1]\right) & & \\
\hline
\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} & \otimes & [0\ -1] & \otimes & \left(\overset{n-2}{\otimes}[2\ 1]\right) \\
\hline
\left(\overset{2}{\otimes}\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) & \otimes & [0\ -1] & \otimes & \left(\overset{n-3}{\otimes}[2\ 1]\right) \\
\hline
\left(\overset{3}{\otimes}\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) & \otimes & [0\ -1] & \otimes & \left(\overset{n-4}{\otimes}[2\ 1]\right) \\
\vdots & \vdots & \vdots & \vdots & \vdots
\end{array}\right]
\begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_{12} \\ A_3 \\ A_{13} \\ A_{23} \\ A_{123} \\ \vdots \end{bmatrix}
$$

In the above equations, the symbols '$\overset{n}{\otimes}$' and '$\otimes$' represent the Kronecker direct product of n and two matrices, respectively. The vertical dotted lines denote the layered vertical Kronecker matrices, and the horizontal dashed lines denote the layered horizontal Kronecker matrices, respectively. A layered horizontal Kronecker matrix is defined as the horizontal sum of Kronecker matrices [10]. In a similar manner, in this Letter the notion of a layered vertical Kronecker matrix is introduced that can be defined as the vertical sum of Kronecker matrices. When the Kronecker direct product of i matrices is carried out for the above equations for $i \leq 0$, then the term $\overset{i}{\otimes}$ disappears from the above equations.

Example: For $n = 3$, the above relations become

$$
\begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_{12} \\ A_3 \\ A_{13} \\ A_{23} \\ A_{123} \end{bmatrix}
= \frac{1}{8}
\left[\begin{array}{ccccc}
\begin{bmatrix} 1 & 1 \\ 0 & -2 \end{bmatrix} & \otimes & \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \otimes & \begin{bmatrix} 1 \\ 0 \end{bmatrix} \vdots \\
\begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} & \otimes & \left(2\begin{bmatrix} 1 \\ -2 \end{bmatrix}\right) & \otimes & \begin{bmatrix} 1 \\ 0 \end{bmatrix} \vdots \\
\left(\overset{2}{\otimes}\begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}\right) & \otimes & \left(4\begin{bmatrix} 1 \\ -2 \end{bmatrix}\right) & &
\end{array}\right]
\begin{bmatrix} H_0 \\ H_1 \\ H_2^{(1)} \\ H_2^{(2)} \\ H_3^{(1)} \\ H_3^{(2)} \\ H_3^{(3)} \\ H_3^{(4)} \end{bmatrix}
$$

$$
= \frac{1}{8}
\begin{bmatrix}
1 & 1 & 2 & 0 & 4 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -8 & 0 & 0 & 0 \\
0 & 0 & -4 & 0 & -4 & 4 & 0 & 0 \\
0 & 0 & 0 & 0 & 8 & -8 & 0 & 0 \\
0 & -2 & -2 & 2 & -4 & 0 & 4 & 0 \\
0 & 0 & 0 & 0 & 8 & 0 & -8 & 0 \\
0 & 0 & 4 & -4 & 4 & -4 & -4 & 4 \\
0 & 0 & 0 & 0 & -8 & 8 & 8 & -8
\end{bmatrix}
\times
\begin{bmatrix} H_0 \\ H_1 \\ H_2^{(1)} \\ H_2^{(2)} \\ H_3^{(1)} \\ H_3^{(2)} \\ H_3^{(3)} \\ H^{(4)} \end{bmatrix}
$$

$$
\begin{bmatrix} H_0 \\ H_1 \\ H_2^{(1)} \\ H_2^{(2)} \\ H_3^{(1)} \\ H_3^{(2)} \\ H_3^{(3)} \\ H^{(4)} \end{bmatrix}
=
\left[\left[\begin{array}{ccccc}
\begin{bmatrix} 2 & 1 \\ 0 & -1 \end{bmatrix} & \otimes & [2\ 1] & \otimes & [2\ 1] \\
\hline
\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} & \otimes & [0\ -1] & \otimes & [2\ 1] \\
\hline
\left(\overset{2}{\otimes}\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) & \otimes & [0\ -1] & &
\end{array}\right]\right]
\begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_{12} \\ A_3 \\ A_{13} \\ A_{23} \\ A_{123} \end{bmatrix}
$$

$$
=
\begin{bmatrix}
8 & 4 & 4 & 2 & 4 & 2 & 2 & 1 \\
0 & 0 & 0 & 0 & -4 & -2 & -2 & -1 \\
0 & 0 & -2 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & -2 & -1 & 0 & 0 & -2 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 0 & -1 & 0 & -1
\end{bmatrix}
\times
\begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_{12} \\ A_3 \\ A_{13} \\ A_{23} \\ A_{123} \end{bmatrix}
$$

As can be easily verified, the results for $n = 3$ are exactly the same as those presented in [4].

Conclusion: In this Letter new mutual relations between the Haar wavelet and arithmetic transforms in the form of layered vertical and horizontal Kronecker products are presented. These relations generalise earlier results presented in [4] and allow known results of spectral logic design in the arithmetic domain [2, 5, 6] to be transferred to the Haar domain and the efficiency of both approaches to be compared for different applications of large Boolean functions. Many other applications of the arithmetic transform in logic design and other areas were described in [6], while some applications of the Haar Wavelet transform are available in [1, 8, 10].

26 February 1999
Electronics Letters Online No: 19990526
DOI: 10.1049/el:19990526

B.J. Falkowski (*School of Electrical and Electronic Engineering, Nanyang Technological University, Block S1, Nanyang Avenue, Singapore 639798, Republic of Singapore*)

References

1 BURRUS, C.S., GOPINATH, R.A., and GUO, H.: 'Introduction to wavelets and wavelet transforms' (Prentice-Hall, Upper Saddle River, 1998)
2 CHANG, C.H., and FALKOWSKI, B.J.: 'Logical manipulations and design of tributary networks in the arithmetic spectral domain', *IEE Proc. Comput. Digital Tech.*, 1998, **145**, (5), pp. 347–356
3 FALKOWSKI, B.J., and CHANG, C.H.: 'Forward and inverse transformations between Haar spectra and ordered binary decision diagrams of Boolean functions', *IEEE Trans. Comput.*, 1997, **46**, (11), pp. 1272-1279
4 FALKOWSKI, B.J.: 'Forward and inverse transformations between Haar wavelet and arithmetic functions', *Electron. Lett.*, 1998, **34**, (11), pp. 1084–1085
5 CHEN, T.H.: 'Fault diagnosis and fault tolerance' (Springer-Verlag, Berlin, 1992)
6 FALKOWSKI, B.J., SHMERKO, V.P., and YANUSHKEVICH, S.N.: 'Arithmetical logic - its status and achievements'. 4th Int. Conf. Applications of Computer Systems, ACS'97, Szczecin, Poland, November 1997, Invited Paper, pp. 208–223
7 KACZMARZ, S., and STEINHAUS, H.: 'Theory of orthogonal series' (Chelsea, New York, 1951) (in German)
8 KARPOVSKY, M.G.: 'Finite orthogonal series in the design of digital devices' (New York, Wiley, 1976)
9 RUIZ, G., MICHELL, J.A., and BURON, A.: 'Switch-level fault detection and diagnosis environment for MOS digital circuits using spectral techniques', *IEE Proc. Comput. Digital Tech.*, 1992, **139**, (4), pp. 293-307
10 YAROSLAVSKY, L.P.: 'An introduction to digital picture processing' (Berlin, Springer-Verlag, 1985)

Tuning logic simulators for timing analysis

D.M. Maksimović and V.B. Litovski

An original method for digital circuit delay estimation within a logic simulator framework and HDL modelling mechanism needed for its implementation are proposed. The method is implemented using the simulator Alecsis and its efficiency is demonstrated on a set of ISCAS'85 benchmark circuits.

The frequency of operation of a circuit is one of the most important user requirements to the digital integrated circuit designer. The maximum operating frequency is determined by the delay of the longest path in the circuit. Circuit delays are usually extracted using timing analysis programs [1 – 3]. When timing analysis shows that the timing requirements are not being met, then the designer must redesign the circuit. To avoid circuit redesign, the designer needs to carry out maximal delay estimation as early as possible in the design process.

In this Letter we show that a versatile logic simulator is capable of producing an early estimation of circuit delays in quite an acceptable amount of CPU time. Only a small improvement is necessary in the simulation mechanism of a standard logic simulator to enable it to act as a timing simulator. We implement this improvement in the simulator's input language in the form of a generalised signal attribute modelling mechanism. The method we propose estimates the propagation delays of the longest structural paths for all signals in the circuit with only one run of the logic simulator.

When a digital circuit is simulated for one specific input vector, the time instant when the last signal transition occurs determines the delay of one (rising or falling) edge at that signal for the given input vector. To obtain the worst-case delays of both rising and falling signal transitions for any input vector, 2^n circuit simulations must be carried out, where n is the number of primary inputs. This approach is not feasible when n becomes large.

We propose a logic simulation based method for delay evaluation that assumes the simultaneous propagation of all input