Compact four bit carry look-ahead CMOS adder in multi-output DCVS logic

G.A. Ruiz

Indexing terms: CMOS integrated circuits, Logic circuits

A four-bit carry look-ahead (CLA) CMOS adder based on transistor sharing in a multi-output differential cascode voltage switch (MODCVS) logic is presented. This adder uses a new enhanced CLA unit, which enables the generation of all output carries in one single compact gate structure. Simulation results using HSPICE with CMOS 1.0μm technology designs show that the four-bit adder proposed has 15.7% less transistors, 27.2% less silicon area, ~14% speed improvement, and a 29.1% reduction in average power consumption compared to a standard DCVS implementation.

Introduction: The differential cascade voltage switch DCVS logic is a differential precharged logic family made up of two complementary logic trees similar to domino logic. It is a complete logic family capable of generating any logic expression simply, and it often requires fewer transistors than traditional logic [1]. However, this logic demands a greater area for routing between logic gates since it requires twice as many interconnection lines. The DCVS logic has recently experienced a wide range of development due mainly to its application in self-timed circuits. In these circuits, the completion signals used by the handshaking requirements can be generated in a general way with DCVS logic [2].

An efficient style of domino CMOS logic, called multiple-output domino logic (MODL) [3], exploits the recursive property in the transistor blocks to extract multiple outputs from the same gate by inserting additional PMOS precharging devices. This technique is only effective in non-complementary logic (domino, NORA, pseudo-NMOS) and cannot usually be applied directly to complementary logic (conventional CMOS, DCVS and other differential logics). This Letter presents a compact four-bit carry look-ahead CLA adder in multi-output DCVS (MODCVS) logic. The CLA's recursive property has enabled its definition to be modified in order to extract multi-outputs in both complementary NMOS logic trees. As a result, the proposed adder has an area and speed efficient structure which can be used for application in self-timed circuits.

Four-bit CLA adder designed by MODCVS logic: High speed arithmetic units based on CLA principle remain dominant, since the carry delay can be improved by calculating the carries to each stage in parallel. Furthermore, the recursive properties of the carries have enabled the development of shared transistor structures in multi-output gates which have shown area and speed improvement compared with respect to single output circuits.

In a CLA gate, if C_{i-1} is the input carry for stage i, and A_i and B_i are the i bits of the input data, then the output carry C_i can be expressed as:

$$C_i = G_i + P_i C_{i-1} \tag{1}$$

where

$$G_i = A_i B_i \text{ and } P_i = A_i \oplus B_i \tag{2}$$

are G_i the generate signal, and P_i the propagate signal.

Expanding this yields:

$$C_i = G_i + P_i G_{i-1} + P_i P_{i-1} G_{i-2} + \ldots + P_i P_{i-1} \ldots P_1 C_0 \tag{3}$$

The sum is generated by

$$S_i = C_{i-1} \oplus A_i \oplus B_i = C_{i-1} \oplus P_i \tag{4}$$

In practice, the number of look-ahead stages is limited to four, since the switching time of the gate grows unacceptably large, due among other factors, to the influence of the body effect, the back gate effect and the high resistive path. The efficiency of the domino logic and the recursiveness in the definition of C_i enables all the carries to be generated in a single MODL gate as proposed in [3]. This gate uses 22 devices, as opposed to the 46 devices required by the equivalent implementation in four domino gates. However, this efficient four-bit MODL carry generator is not directly transferable to DCVS logic, since the complementary carries $\overline{C}_4$, $\overline{C}_3$, $\overline{C}_2$ and $\overline{C}_1$ need to be generated in the same gate. The lack of duality between the complementary logic trees can be solved by modifying the $\overline{C}_i$ definition. To do this, eqn. 1 is used as a starting point and the following operation is made:

$$\begin{aligned}\overline{C}_i = \overline{G}_i(\overline{P}_i + \overline{C}_{i-1}) &= \overline{G}_i(\overline{P}_i + P_i)(\overline{P}_i + \overline{C}_{i-1}) \\ &= \overline{G}_i\overline{P}_i + \overline{G}_i P_i \overline{C}_{i-1}\end{aligned} \tag{5}$$

but

$$\overline{G}_i P_i = P_i \text{ and } \overline{G}_i\overline{P}_i = \overline{A_i + B_i} = N_i \tag{6}$$

where N_i is the generate signal of $\overline{C}$, resulting in

$$\overline{C}_i = N_i + P_i \overline{C}_{i-1} \tag{7}$$

and expanding this yields

$$\overline{C}_i = N_i + P_i N_{i-1} + P_i P_{i-1} N_{i-2} + \ldots + P_i P_{i-1} \ldots P_1 \overline{C}_0 \tag{8}$$

It can be observed that the definition of $\overline{C}_i$ is similar to that of C_i, replacing N_i with G_i, and C_{i-1} with $\overline{C}_{-1}$. Moreover, is expressed through two mutually exclusive terms (note $N_i P_i = 0$) which, as with C_i (note $G_i P_i = 0$), avoids the problems of 'false' discharges which are produced at the output nodes due to OR-AND forms of MODL gates. This problem stems from a logic fault when a undesired discharge is produced at a lower AND dynamic output node when a higher OR dynamic node is pulled down.

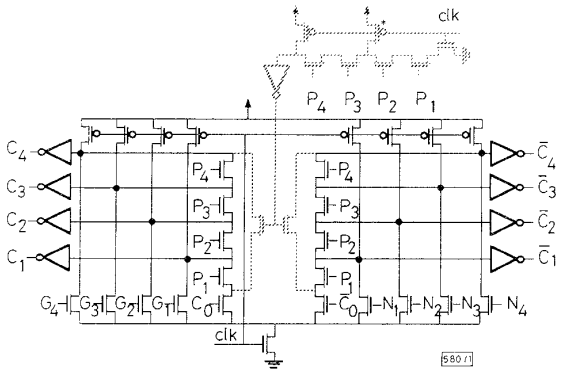

Fig. 1 *CLA gate in MODCVS logic with additional reduced carry propagation circuit*

– – – – reduced carry propagation circuit

Fig. 1 shows the compact CLA gate obtained when expressions eqns. 3 and 8 are implemented in MODCVS logic. The compact nature of the structure of this CLA is highlighted by comparing its 43 transistors with the 60 transistors which would be required in a conventional DCVS scheme which uses a MODL gate to generate the C_i, or with the 80 transistors which would be required in the equivalent four carry-generating DCVS gates. Furthermore, it uses 14 input lines (G_i, $\overline{N}_i$, P_i, C_0, $\overline{C}$) as opposed to the 18 input lines (P_i, $\overline{P}_i$, Gi, $\overline{G}$, C_0, $\overline{C}_0$) of a standard implementation. Both factors produce the result that this new enhanced CLA reduces silicon area, increases circuit performance and decreases power, because of the reduction of device count, wire length and input loading.

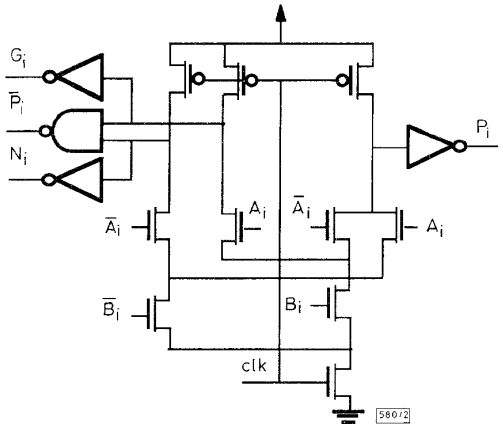

Fig. 2 *Generation of propagate and generate signals*

Moreover, the symmetry which exists between both complementary logic trees equilibrates the worst case delay of the outputs to the discharge of a chain of six transistors. The use of scaling techniques is advisable in the design stage, as this improves the gate speed and reduces the total channel area of the NMOS chain. Moreover, this CLA accepts the alternative charge compensation scheme derived from the domino logic based on PMOS transistors fedback from the output which solves the charge sharing and charge leakage problems. Fig. 2 shows the gate which generates the signals G_i, P_i, $\bar{P}_i$ and N_i according to eqns. 2 and 6. This implementation uses the recurrency of the logic to reduce the device count. Fig. 3 shows the EXOR gate which generates the output sum according to eqn. 4.

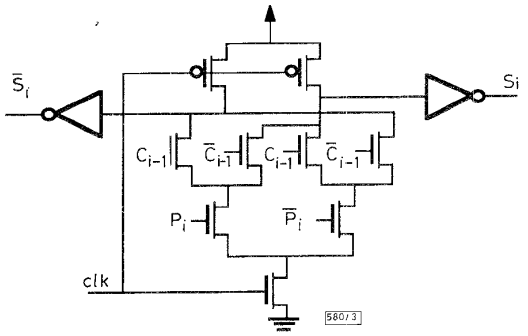

Fig. 3 *EXOR gate*

The worst case propagation time of the output carry (C_4, $\bar{C}_4$) can be improved through the additional circuitry shown in the dashed lines of the CLA gate which is made up of 11 transistors. The PMOS transistor marked by * is for avoiding the charge sharing problem. This circuit consists in a four-bit dynamic AND domino gate which turns on the two bypass transistors if all carry propagate signals are true. The overall speed in CLA four-bit parallel adders connected in series should be greatly reduced because all P_i are generated in parallel and are evaluated simultaneously in this circuitry.

Table 1: Comparison between conventiona and proposed adder

4 bit			
	Conventional	Proposed	Reduction ratio
Area	289 × 315μm²	251 × 264μm²	27.2%
Number of transistors	209	176	15.8%
Worst-case carry	2.91ns	2.58ns	11.3%
Worst-case sum	3.70ns	3.24ns	12.4%
Average power consumption (at 1MHz)	3.81mW	2.7mW	29.1%

32 bit			
	Conventional	Proposed	Reduction ratio
Worst-case carry	18.87ns	15.43ns	18.2%
Worst-case sum	19.51ns	16.10ns	17.5%
Average power consumption (at 1MHz)	29.7mW	20.3mW	31.6%

Two four-bit CLA adders have been designed in a double-metal 1.0μm CMOS technology using the enhanced CLA in MODCVS proposed, and a conventional DCVS scheme with a MODL CLA to generate the C_i. In turn, these adders have been used to design 32 bit CLA adders. The simulation of these circuits has been carried out using an HSPICE circuit simulator at level six with V_{DD} = 5V and C_L = 0.2pF (Table 1). The input operands were A = 0xF, $\bar{A}$ = 0x0, B = 0x0 and $\bar{B}$ = 0xF in the case of four-bit, and A = 0xFFFFFFFF, $\bar{A}$ = 0x0000000, B = 0x0000000 and $\bar{B}$ = 0xFFFFFFFF in the case of 32 bit, with input carry C_0 = 0 and $\bar{C}_0$ = 1, which corresponds to the worst case. The CLA adder proposed reduces the device count, wire length, fan-in and fan-out, conferring substantial advantages in terms of speed, area and power consumption over standard CLA in DCVS logic.

7 June 1996
Electronics Letters Online No: 19961082

G.A. Ruiz (*Departamento de Electrónica y Computadores, Facultad de Ciencias, Universidad de Cantabria, Avda. Castros s/n, 39005 Santander, Spain*)

References

1 CHU, LK.M., and PUFFREY, D.L.: 'Design procedures for Differential Cascode Voltage Switch Circuits', *IEEE J. Solid-State Circuits*, 1986, **21**, (6), pp. 1082–1087
2 MENG, T.H.Y.: 'Synchronisation design for digital systems' (Kluwer, Norwood, MA, 1991)
3 HWANG, I.S., and FISHER, A.L.: 'Ultrafast compact 32-bit CMOS adders in multiple-output domino logic', *IEEE J. Solid-State Circuits*, 1989, **24**, (2), pp. 358–369

Parallel pixel processing using programmable gate arrays

D.M. Budgett, P.E. Tang, J.H. Sharp, C.R. Chatwin, R.C.D. Young, R.K. Wang and B.F. Scott

Indexing terms: Programmable logic devices, Image processing, Parallel algorithms

A reconfigurable hardware design permits very fast feature extraction from high frame rate video images. By implementing parallel pixel processing paths in programmable gate arrays, a wide range of image processing algorithms can be implemented in realtime.

Introduction: Programmable gate arrays (PGAs) can be used to perform image analysis tasks at very high frame rates. Fast cameras output data at 16 million pixel/s. Using a digital signal processor, only simple real-time algorithms (e.g. thresholding) can be implemented at these data rates. However, PGAs can implement a wide range of algorithms at high speed by using parallel pixel processing paths.

A hybrid optical/digital correlator system is currently under construction at Glasgow University as part of the Brite EuRAM II project contract number BRE2-CT93-0542 [1]. Objects contained in the field of view of an input CCD camera are compared against a database of images stored in a holographic memory [2]. This system will be capable of computing the correlation between a 512 × 512 pixel input image with a database of images at a rate of 2900 frame/s. Application areas include high speed quality control in manufacturing processes and real-time object identification and tracking.

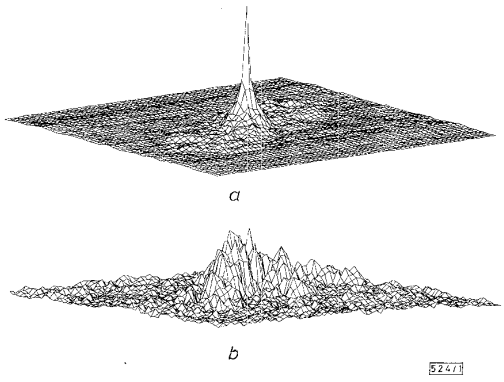

Fig. 1 *Energy distribution in correlation plane*
a Sharp peak indicates good match between images
b Broad distribution indicates poor match between images

Each time a correlation between the input and a database image is performed, a correlation plane output image is produced. Fig. 1*a* illustrates the energy distribution in the correlation plane when