Quaternary Look-up Tables Using Voltage-Mode CMOS Logic Design

Ricardo Cunha, Henri Boudinov and Luigi Carro
Universidade Federal do Rio Grande do Sul, Porto Alegre - Brazil
ricardo@if.ufrgs.br, henry@if.ufrgs.br and carro@inf.ufrgs.br

Abstract

Data processing optimization is one of the main concerns for developing of multiple-valued logic. An advantage could be achieved by realization of new functions existing in non-binary logic. These new logic functions could be implemented using quaternary look-up tables. In this work, a quaternary multiplexer circuit is designed to implement any n-variable quaternary logic function based on its truth table. Voltage-mode CMOS with multi-threshold transistors and multi-Vdd quaternary design was suggested. The multiplexer circuit consists of quaternary Down Literal Circuits, binary inverters and binary pass transistor gates. All circuits were simulated with the Spice tool using TSMC 0.18 μm technology and have shown improvements in performance and power consumption and using less transistors than their equivalent binary circuits.

1. Introduction

In order to go beyond regular CMOS, several researchers have tried to compact more information in a single gate or wire. This has been successfully accomplished in flash memories, for example, where a single memory cell can hold 4 different logic values [1]. Regarding combinational circuits, most of the efforts have been targeted for the use of Multiple Valued Logic (MVL) Circuits [2]-[5] and several kinds of MVL circuits have been developed in the past decades in several different technologies from earlier works on bipolar technologies to novel solutions presented recently using floating gates [6], capacitive logic[7] and quantum devices. There are good implementations showing somewhat improvements compared to binary circuits, but none of them have at the same time all requirements to be used in VLSI circuits. Some realizations present static consumption, some circuits present low performance, some circuits consume too much area and almost no implementation has shown completeness.

Completeness means to have a complete set of logical circuits which permits one to make any logic operation using a specific technology. It is crucial to take advantage of all multiple valued logic potential because one of the main advantages of multiple valued logic is the possibility to optimize or minimize data processing. The processing improvement can be done by the use of new logical functions that are possible to be implemented using multiple valued logic. To assure completeness is necessary to demonstrate a general purpose circuit. The idea of a universal quaternary logic block has been proposed [9] and known like "T-gate". The use of quaternary look-up tables to implement such circuit was demonstrated in [10] using current-mode and voltage-mode circuits but these circuits present a high consumption and therefore do not match the power requirements for VLSI circuits.

In order to match VLSI requirements, this work present a quaternary general purpose circuit in voltage-mode technology that is the realization of multiplexer with high performance, negligible static and low dynamic consumption using less transistors than the equivalent binary circuit. The multiplexer was simulated with the Spice tool using TSMC 0.18 technology and applied to implement one and two variable quaternary functions. Simulations on binary equivalent circuit were also carried out to compare the results.

2. Binary and quaternary look-up tables

When processing capabilities of any radix logic are compared, a good metric is to compare the number of possible functions on these logics. Depending on the radix and the number of variables used, different logic functions are encountered. The number of possible functions is

$$N = b^{(b^n)} \qquad (1)$$

Proceedings of the 37th International Symposium on Multiple-Valued Logic (ISMVL'07)
0-7695-2831-7/07 $20.00 © 2007 IEEE

where b is the base or radix and n is the number of variables. In binary logic of one variable (b=2, n=1), there are 4 possible functions, shown in table I.

Table I. One variable binary functions

in	F1	F2	F3	F4
0	0	0	1	1
1	0	1	0	1

For a two variable binary logic (b=2, n=2), the number of available functions is 16 as we can see in table II.

Table II. Two variables binary functions

In A	In B	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15	F16
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

In quaternary logic of one variable (b=4, n=1), there are 256 possible functions. Some of these functions are show in table III.

Table III. Some of the one variable quaternary functions

In	F1	F2	F3	F4	F5	...	F109	...	F199	...	F229	...	F256
0	0	0	0	0	0	...	1	...	3	...	3	...	3
1	0	0	0	0	0	...	2	...	0	...	2	...	3
2	0	0	0	0	1	...	3	...	1	...	1	...	3
3	0	1	2	3	0	...	0	...	2	...	0	...	3

We can see in table III that there are many more possible functions in quaternary logic than in binary logic and almost all of these functions do not have any equivalence or correspondence in binary logic. They can be used do realize the needed operations in alternative ways or maybe create new paradigms for data processing. To exemplify some new functions, F109 shows the succession function that is a cyclic addition of one unit to the input, F199 shows the succession function that is a cyclic subtraction of one unit to the input and F229 shows the quaternary inversion function that is a diametric inversion of the input.

For two variable quaternary logic (b=4, n=2), there are exactly 4^{16} possible functions. This huge amount of logic functions include quaternay MIN and MAX functions that are non-binary correspondences of the binary AND and OR. Beyond basic functions like MIN and MAX, quaternary two variables logic also include arithmetic functions like the adder function and produt function as we can see in table IV.

Table IV. Sum, product, MIN and MAX functions as examples of two variables quaternary functions.

inputs		Sum		Product		Min	Max
A	B	S	Cs	P	Cp		
0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1
0	2	2	0	0	0	0	2
0	3	3	0	0	0	0	3
1	0	1	0	0	0	0	1
1	1	2	0	1	0	1	1
1	2	3	0	2	0	1	2
1	3	0	1	3	0	1	3
2	0	2	0	0	0	0	2
2	1	3	0	2	0	1	2
2	2	0	1	0	1	2	2
2	3	1	1	2	1	2	3
3	0	3	0	0	0	0	3
3	1	0	1	3	0	1	3
3	2	1	1	2	1	2	3
3	3	2	1	1	2	3	3

Together with MIN and MAX functions, table IV shows the quaternary sum and product functions of 2 numbers. Addition of two quaternary numbers results in a sum output bit (S) and a carry output bit (Cs). The quaternary product of 2 numbers results in a product output bit (P) and a carry output bit (Cp).

The new functions allowed by the use of quaternary logic can permit the organization of the needed operation in alternative ways using less components, with low power dissipation, reduction of critical paths and area savings.

Any operation on any radix logic can be done using look-up tables (LUTs). In discrete levels logic, an n-bit lookup table can be implemented with a multiplexer whose control input signals are the inputs of the LUT and whose inputs are constants. An n-bit LUT can encode any n-input logic function by modeling such functions as truth tables. This is an efficient way of encoding logic functions, and 4-bit binary LUTs are in fact the key component of modern FPGAs.

In digital circuit design, the multiplexer is a device that has multiple input streams and only one output stream. It forwards one of the input streams to the output stream based on the values of one or more "selection inputs" or control inputs. For example, a two-input multiplexer is a simple connection of logic gates whose output is either input A or input B depending on the value of a third input Z which selects the input. The schematic of a binary 1 bit multiplerxer is show in figure 1.

A 2 bit binary multiplexer is designed using 1 bit multiplexers as building blocks as shown in figure 2. This multiplexer have four inputs (A, B, C and D) and two control inputs (Z1 and Z2).

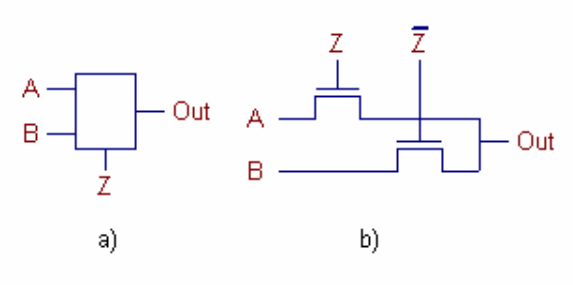

Figure 1. a) One bit binary multiplexer representation and b) schematic.

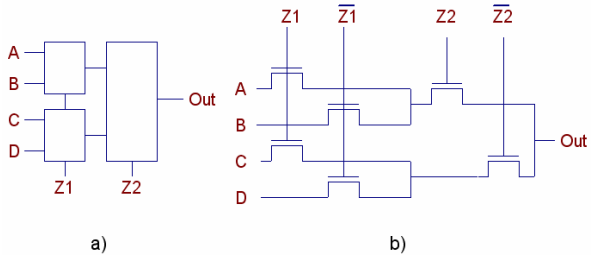

Figure 2. a) Block diagram and b) schematic of a binary MUX 4:1.

For a quaternary multiplexer, all inputs and the output are quaternary and it can have four quaternary input streams using only one quaternary control signal. So the simplier quaternary MUX can implement all possible functions of one variable allowed in this logic (see table I). The block representation of the quaternary multiplexer is shown in the figure 3.

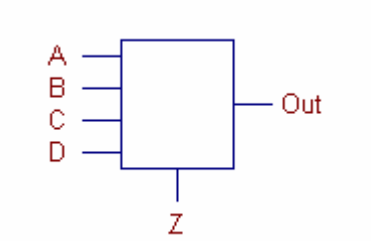

Figure 3. Representation of the quaternary MUX 4:1.

In binary logic to implement these 256 functions would be necessary 8 binary inputs and 3 control signals as shown in figure 4.

The design of the quaternary multiplexer circuit will be presented in details in the next topic. The quaternary multiplexer can be used in the same fashion as in binary logic to implement a two variable LUT as we can see in figure 5.

The two variable multiplexer has sixteen quaternary inputs (A, B, C, ..., and P) and only one quaternary output. Z1 and Z2 are the control inputs that set one and only one of the inputs to the output. This circuit implements 4^{16} functions.

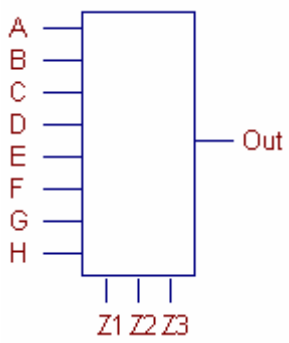

Figure 4. Binary MUX 8:1.

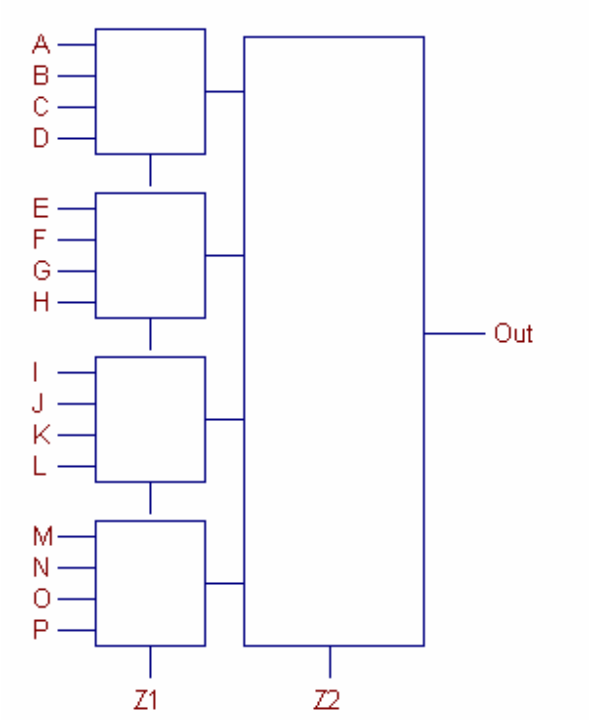

Figure 5. Block diagram of a quaternary MUX 16:1.

3. Quaternary multiplexer design

The proposed multiplexer circuit is based on voltage-mode quaternary logic CMOS circuits presented in a previous work [11]. These circuits use several different transistors with different threshold voltages and operate with four voltage levels, corresponding to 0V common terminal and three power supply lines of 1V, 2V and 3V. The quaternary MUX is designed using Down Literal Circuits (DLCs), binary inverters and pass transistor gates. Down Literal Circuits performs the functions shown in table V. There are 3 possible Down Literal Circuits in quaternary logic, named DLC1, DLC2 and DLC3.

Proceedings of the 37th International Symposium on Multiple-Valued Logic (ISMVL'07)
0-7695-2831-7/07 $20.00 © 2007 IEEE

Table V. Down literal circuits truth table.

In	Down Literal Circuits Outputs		
	DLC 1	DLC 2	DLC 3
0	3	3	3
1	0	3	3
2	0	0	3
3	0	0	0

The Down Literal Circuits are designed in CMOS technology with 3 different threshold voltages for PMOS transistors and 3 different threshold voltages for NMOS transistors. Table VI shows Vt values relative to the transistors source-bulk voltages.

Table VI. Transistor V_T values related to Vs.

	T1	T2	T3	T4	T5	T6
Vt	-2.2	2.2	-1.2	0.2	-0.2	1.2
Type	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS

Each DLC uses one PMOS and one NMOS transistors from table VI connected as shown in figure 6. DLC1 is designed substituting the PMOS transistor of a binary inverter with T1 and the NMOS with T4. DLC2 is designed using T3 and T6 and DLC3 using T5 and T2. In these circuits only one transistor is conducting for each of the four input value. In DLC1, with 0V at the input T1 is conducting and T4 is OFF so that the output is connected to 3V while with 1V, 2V and 3V at the input, T1 is off and T4 is conducting and the output goes to ground. In DLC2, T3 is conducting for 0V and 1V at the input while T6 is conducting for 2V and 3V at the input. For DLC3, T5 is on for 0V, 1V and 3V while T2 is conducting for 3V at the input. Drain currents versus gate voltage for transistors used in DLC are also shown in figure 6.

The quaternary multiplexer 4:1 is designed using the 3 DLCs, 3 binary inverters and 6 pairs of pass transistor gates and its schematic is presented in figure 7. This circuit has 4 quaternary inputs (A, B, C and D), one quaternary output (Out) and 1 quaternary control signal (Z) that sets the output to one of the 4 inputs. The quaternary control signal is split in 3 different binary control signals by DLCs. These binary control signals are applied to the pass transistors gates. In general case the pass transistors are a pair of a NMOS and a PMOS transistors that are used to transmit any input signal without degradation. For constant inputs one can use only 1 NMOS transistor to transmit the 0V signal and only 1 PMOS to transmit the 1V, 2V and 3V saving area.

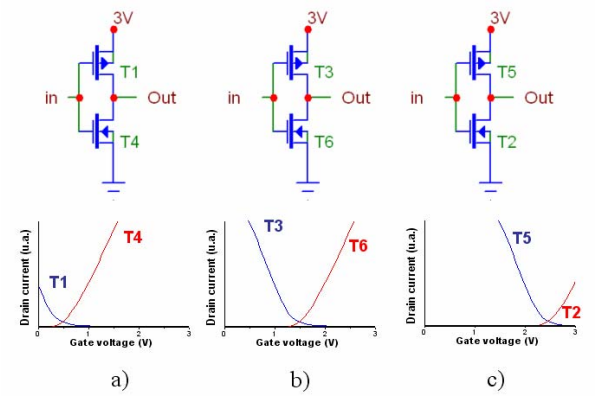

Figure 6. Schematics of a) DLC1, b) DLC2 and c) DLC3 and drain currents versus gate voltages of transistors used.

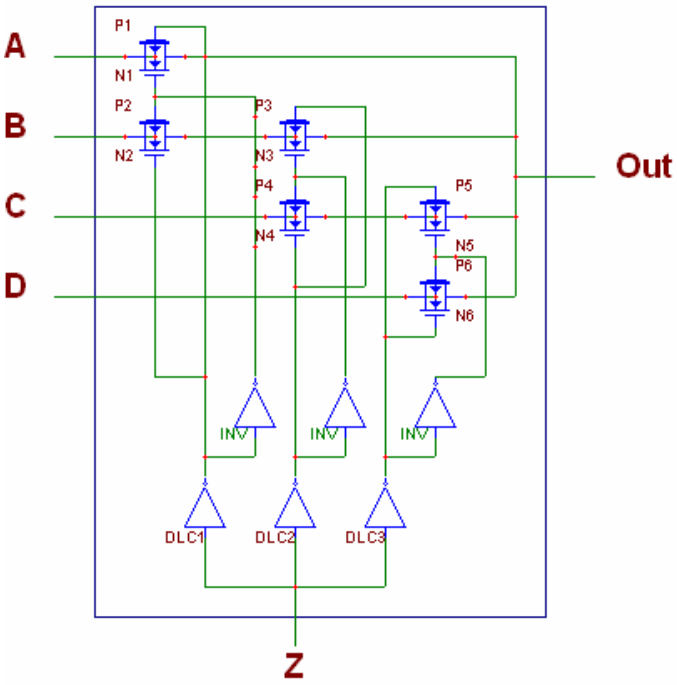

Figure 7. Quaternary MUX 4:1 schematic circuit.

Each control input value implies a different set of DLCs outputs that are used to control the pass gates in such a way that when the input is 0V, 1V, 2V and 3V, the output is connected to A, B, C and D respectively. When the control input (Z) is 0V, all DLCs outputs are high. The output of DLC1 and its negative are applied to the NMOS pass transistor gate N1 and PMOS pass transistor P1, respectively, setting the output to the A value. In the same time these two signals are also applied to the PMOS P2 and to NMOS N2, respectively turning them off, therefore opening the path between B and output. The DLC2 and DLC3 output and their negatives are used to turn off P4, N4, P6 and N6 opening the paths between output and C and

D inputs. If Z is set to 1V, DLC1 output is 0 and P1 and N1 are both off, while P2 and N2 are on and the output is connected to B while the path between output and A input is opened. If Z is set to 2V, DLC2 output is low turning P3 and N3 off while P4 and N4 are on and the output goes to the C value. If Z is set to 3V, DLC3 output is low and P6 and N6 are on leading the output to the D value. This circuit uses 24 transistors in the general application, where inputs are not constant and can assume any of the 4 logic levels, and 18 transistors where inputs are constants. For constant inputs it is not necessary the use of pairs of pass transistor gates, just NMOS transistors to transmit ground signal and PMOS transistors to transmit VDD signals.

4. Simulation results

Simulations of the multiplexer circuit and its equivalent circuit in binary logic have been performed. The simulations were carried out with the Spice tool using TSMC 0.18µm technology.

The multiplexer was simulated for all 12 possible transitions of the control input for all possible functions of one variable. Figure 8 shows as an example, the output transients of the quaternary inversion function realized with the multiplexer. A binary inverter with dimensions of L=0.18µm and W=0.81µm was used as a load capacitance at the output for all simulated circuits.

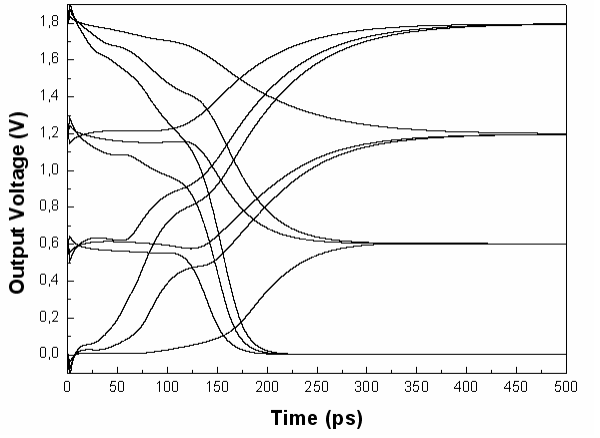

Figure 8. Multiplexer output transients for all possible transitions of the quaternary control input terminal.

Simulations of all 256 possible quaternary functions of MUX 4:1 presented critical propagation delay times of 0.513ns and 0.346ns for the control signals and inputs transitions, respectively. The power consumption was 47µW at 1GHz and the circuit has 24 transistors. The equivalent binary circuit (fig. 4) uses 3 inverter circuits and 24 pairs of pass transistors to implement 256 functions, which is a total of 54 transistors. In this circuit the corresponding propagation delays and power consumption are 0.665ns, 0.664ns and 84 µW at 1GHz.

For two variables quaternary logic, we simulated some functions to estimate performance and power dissipation. The simulations showed a mean power dissipation of 134 µW at 1GHz, propagation delays of 1.048ns and 0.687ns and the circuit uses 120 transistors. The equivalent binary circuit is a multiplexer 32:1 (which is a combination of four binary MUX 8:1 and one binary MUX 4:1) with 5 control inputs and 236 transistors. The simulations show a mean power dissipation of 450 µW at 1GHz and propagation delays of 1.10ns and 1.07ns. A summary of simulations are shown in table VII.

The better performance and lower power dissipation for quaternary logic, compared to binary logic, is mainly due to the lower number of transistors, consequently decreasing capacitances. Decreasing the total capacitance, dynamic consumption and propagation delay are reduced.

Table VII. Comparison between quaternary and binary multiplexer circuits.

Multiplexer circuit	Propagation Delay [ns]		Mean Power dissipation at 1Ghz [µW]	Number of transistors
	Control signals transitions	Input signals transitions		
Quaternary MUX 4:1	0.513	0.346	47	24
Binary MUX 8:1	0.665	0.664	84	54
Quaternary MUX 16:1	1.048	0.687	134	120
Binary MUX 32:1	1.099	1.066	450	236

5. Conclusion

This work presented a new way to implement quaternary look-up tables using a multiplexer circuit to implement any quaternary logic function based on its truth table. This circuit can be used in combinational logic or as a building block in FPGAs. Simulation results have shown the advantages of the quaternary implementation compared to the binary equivalent circuits. The quaternary implementations have shown higher performance, lower power dissipation and use less transistors than the equivalent binary circuits for one and for two quaternary variables.

6. References

[1] G. Atwood et al., "Intel strataFlash™ memory technology overview," *Intel Technol. J.*, vol. Q4, 1997.

[2] E. Dubrova, "Multiple-Valued Logic in VLSI: Challenges and Opportunities", Proceedings of *NORCHIP'99*, pp. 340-350, 1999.

[3] K. C. Smith, "The prospects for multivalued logic: A technology and applications view", *IEEE Trans. on Computers* vol. C-30, no. 9, pp. 619-634, 1981.

[4] S. L. Hurst, "Multiple-Valued Logic - Its status and its future", *IEEE Trans. on Computers* C-33, no. 12, pp. 1160-1179, (1984).

[5] J. T. Butler, Multiple-valued logic, *IEEE Potentials* vol. 14, no. 2, pp. 11 – 14, 1995.

[6] Y. Berg, S. Aunet, O. Mirmotahari, and M. Høvin, "Novel Recharge. Semi-Floating-Gate CMOS Logic For Multiple-Valued Systems," Proc. of the *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2003.

[7] A. Schmid, Y. Leblebici. "Realisation of multiple-valued functions using the capacitive threshold logic gate", Proc. of *IEE Computer and Digital Techniques*, v. 151, n. 6, pp. 435-447, 2004.

[8] A. Raychowdhury, K. Roy, "Carbon-nanotube-based voltage-mode multiple-valued logic design", *IEEE Trans. Nanotechnology*, Mar. 2005, 4(2), 168-179.

[9] K.C. Smith, "Multiple-valued logic: A tutorial and appreciation", *IEEE Computer*, vol.21, pp.17–27, 1988.

[10] Sheikholeslami, A.; Yoshimura, R.; Gulak, P.G; "Look-up tables (LUTs) for multiple-valued logic, combinational logic", Proc. of the 28th *International Symposium on Multiple-valued Logic*, pp 264-269, 1998.

[11] R.C.G. da Silva, H. Boudinov, L. Carro; "A novel Voltage-mode CMOS quaternary logic design", *IEEE Trans. Elec. Dev.*, v. 53, n. 6, pp 1480-1483, Jun 2006.

Proceedings of the 37th International Symposium on Multiple-Valued Logic (ISMVL'07)
0-7695-2831-7/07 $20.00 © 2007 IEEE