CONFIGURABLE RING OSCILLATOR FOR LOGIC CELL EVALUATION

[1]Fábio R. Pereira, [1]Carlos A. F. da Silva, [1]Dionatan S. Moura, [1]Leomar S. Rosa Jr.,
[2]André I. Reis, [1]Renato P. Ribas

[1]Instituto de Informática – UFRGS, Porto Alegre, Brazil
[2]Nangate Inc., Menlo Park, CA, Herlev, Denmark
{frpereira, csilva, dsmoura, leomarjr, rpribas@inf.ufrgs.br}, are@nangate.com

ABSTRACT

This paper addresses the subject of how to characterize a logic cell considering propagation delay, power dissipation and input capacitances. This represents an important issue to compare different logic styles and topologies. It proposes an automatic method to extract these measurements in a few simulations. An example of logic function, already presented in other works, is considered in order to validate the proposed methodology.

1. INTRODUCTION

The design style used in logic gates basically influences the speed, size and power dissipation of a circuit. The circuit delay is determined by the number of inversion levels, the number of transistors in series, the transistor sizes, and the capacitances present around each logic cell in the circuit. Circuit size depends on the number of transistors and their sizes, and the wiring complexity. Power dissipation is dependent on the switching activity, leakage behavior, and node capacitances. All these characteristics may vary considerably from one logic style to another.

Several logic styles are demonstrated in the literature, such as PTL or Pass Transistor Logic [3], CSP (Complementary Series-Parallel, also known as static CMOS logic family) [4], NCSP (Non-Complementary Series-Parallel) [5-6], and LBBDD (Lower Bound Networks derived from BDDs) [7]. Each one of these logic style presents particular properties and benefits. This way, the proper choice of one design style to implement a given logic function is crucial for the circuit optimization. This situation leads to the following question: 'how a given logic function implemented in different design styles can be compared ?'

The common approach is to perform the electrical simulation of the logic cells. In this strategy the delay time of each cell are usually evaluated, as well as the power dissipation. The area is normally evaluated in a separated way, by transistor count or by analyzing the size of the transistors. Also, it is possible to use some combined figure-of-merit to get a better evaluation of the logic cells. For example, it is possible to use the 'delay x power consumption' relationship as a metric. The main drawback in this approach is that the effect of the input capacitances presented on the cell is ignored. In other words, when these cells are simulated and evaluated in a circuit, the effect of the capacitances may modify the behavior of the cell, since it depends also on the input signal slope.

One well known structure that permits to evaluate the 'input capacitances x delay' relationship is the ring oscillator [8]. This structure is widely used to evaluate the performance of novel process technologies. Basically, it is composed by an odd number of inverters to oscillate the signal.

In this works, the ring oscillator architecture for complex gates analysis is proposed. The architecture is composed by configurable nodes to permit the simulation of different loads on the inputs/outputs of the cells. The architecture is able to provide delay and power consumption information taking into account both input and output capacitance effects simultaneously.

The remaining of this paper is organized as follows. Section 2 presents the proposed architecture. Section 3 discusses about the proposed measurements. In Section 4 some results are presented. Finally, Section 5 presents the conclusions.

2. PROPOSED ARCHITECTURE

The proposed ring oscillator, illustrated in Fig. 1, consists of odd number of cells, or stages, which are called here as 'enclosure cell'. These cells have just one input and one output connected in the signal path. The output of each cell is connected on one input of the subsequent cell. The ring oscillator requirement of odd stages must be respected to guarantee the oscillation

.

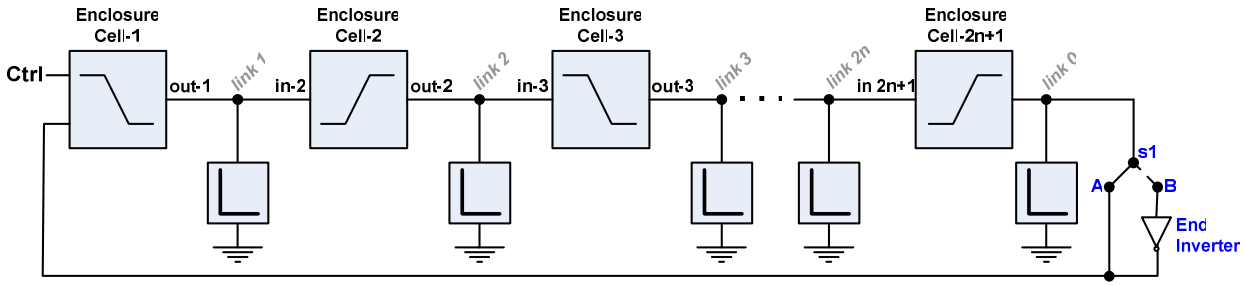

Fig. 1 – Ring oscillator block diagram.

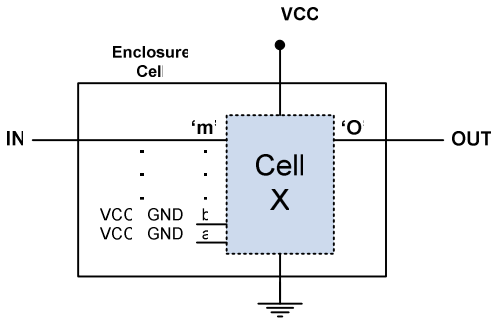

Fig. 2 – 'Enclosure cell' illustration.

2.1. Enclosure Cell

The 'enclosure cell', exhibited in Fig. 2, is equal to a package in which it can insert any X-Cell whose function 'o'(a, b, c..., 'm') describes its behavior. Thus, it is necessary to connect one of the inputs (a, b, c..., m) to the input 'IN', which implies 'm' interconnection possibilities. For each possible interconnection the 'm-1' remaining inputs are connected to '1' or '0' logic levels in 2^{m-1} different ways.

2.2. Input Vector

The input vectors are classified in three subsets D, E and N, as illustrate in Fig. 3. The 'enclosure cell – 1' has an additional input 'ctrl' to be explained further.

The subset D is composed by input vectors which well define the 'o' state independent of 'm' state, in other words, they disable the 'm' input to change the output. The subset E, in turn, contains inputs which although allows the 'm' input changes the output these values came the same: 'o' = 'm'. Finally, the subset N contains the inputs which allows 'm' change 'o' and 'm' = '!o', that means, 'm' is not equal to 'o'. This classification was automated by a java programming.

The subset D not applies to the ring oscillator because is impossible to oscillate with this input. For the group E is possible make the ring oscillates just switching the S1 to B which inserts an inverter in series with the ring. This trick is not recommended as the inverter adds its delay in the total ring delay increasing the oscillation period.

If it is inevitable the designer can decrease the consequent error using a bigger number of cascaded cells with penalty in simulation time. In this study the subset used was N and the S1 switched to A brings oscillation without errors in measures. However, not all inputs even functions posses the subset N.

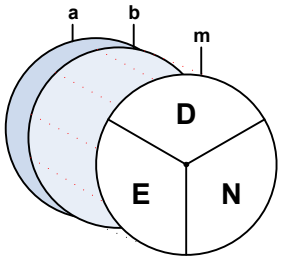

Fig. 3 – Input vector subsets.

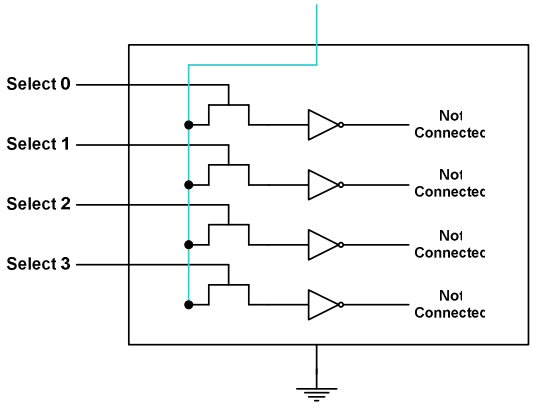

Fig. 4 – L-cell (or load cell).

2.2. Load Cell

Between the 'enclosure cells' was added a programable L-cell (load cell) which function is change the fanin and fanout of the cells in a controlled way as shows Fig. 4. These cells change the input/output slopes, frequency and consumption.

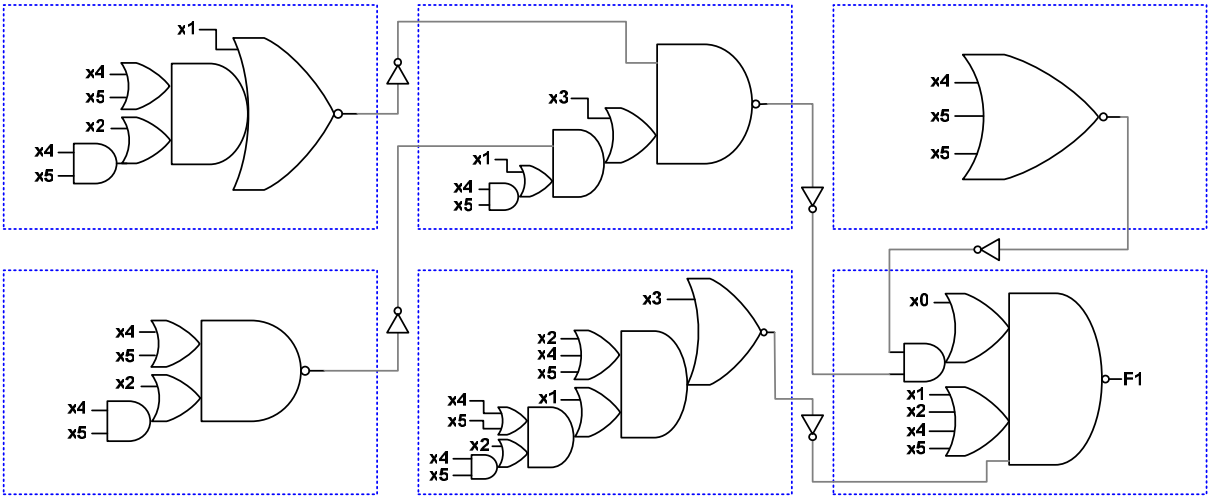

Fig. 5 – Function F1 mapped in static CMOS style by ELIS tool [9].

3. PROPOSED MEASURES

3.1. Hspice Simulations

In the present study, Hspice simulator [1] was adopted to extract power measures, frequency and capacitance. As this tool needs the definition of initial condition to converge the 'enclosure cell – 1', an additional bit was added, which is linked to a step source when the simulation starts. For each 'n' input vector of N an interconnection Ctrl(n) is defined.

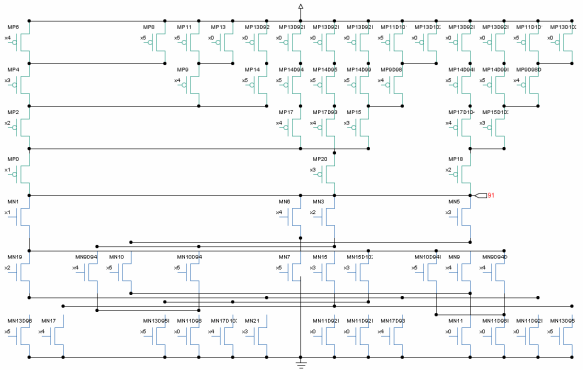

Fig. 6 – Function F1 mapped in LBBDD style.

In this work, the simulation of the circuit that implements the function bellow was used as the test vehicle:

```
F1=!x0*(!x3*(!x1*(!x5+!x4)+(!x2*(!x5+!x4)+!x4*!x
5))+(!x1*(!x2*(!x5+!x4)+!x4*!x5)+(!x2*!x4*!x5)))
+!x1*(!x2*!x4*!x5)+!x3*(!x1*(!x2*(!x5+!x4)+!x4*!
x5)+(!x2*!x4*!x5))
```

This function was synthesized in tree different ways: CSP via ELIS [9], NCSP [2] e finally LBBDD [7]. As a matter of simplicity, it just evaluates one input vector 'n' for each circuit input and it measures the following values with respective commands:

a) Frequency - found through period as follows:

```
.measure tran period trig V(link21) val=fifty%
rise=1 targ V(link21) val=fifty% rise=2
```

That means, the time between two oscillations in the same ring point, the 21st cell output in this case.

b) Time delay - measured tdhl (high-low time delay) and tdlh (low-high time delay) as well as the average value.

```
.measure tran tdlh trig V(link0) val=fifty%
fall=1 targ V(link1) val=fifty% rise=2
```

The rise (tr) and fall (tf) times were obtained in same way. As follows the 'tr' example:

```
.measure tran trelo1 trig V(link1) val=ten%
rise=1 targ V(link1) val=ninety% rise=1
```

c) Power dissipation - The dynamic consumption was obtained through product of voltage by average current measure during one cycle. The cell consumption is 1/23 of total value. As follows the hspice example:

```
.measure tran avgi avg I(Vdd1) from=5ns to='5ns
+ period'
```

The static consumption was obtained through a dc simulation. However, it was not necessary to use the ring oscillator. As follows the Hspice example:

```
.dc SWEEP DATA = inputs
.DATA inputs
        Index   Vx5     Vx1     Vx0
        1       0       0       0
        2       0       0       3.3
        .       .       ..      .
   .ENDDATA
```

With these results it is possible to calculate the average consumption, the low consumption vector and the maximum consumption.

d) Capacitance - obtained through current average divided by voltage. It uses a zero voltage source in series with the measure point to obtain the current average. As follows the hspice example:

```
.measure tran ielo1r integ I(Vzero) from=tstrt
to='tstrt + period/4'
```

Next, 'ielo1' is divided by voltage Vcc. The same measure was used to get the load cell capacitance.

4. RESULTS

The F1 function mapped through ELIS tool [9] in CSP logic style brings six cells plus five inverters, as showed in Fig. 5. In LBBDD and NCSP logic styles, F1 was mapped on a unique cell as shown in Fig. 6 and Fig. 7. The environment adopted in the aveluation was Hspice simulator in a SUN machine. The technology was CMOS 180nm and the transistor width sizing was 0.24µm. Table I shows the summary of obtained results: minimum frequency (freq), average dynamic power (power), average static power (leakage) and maximum average time delay (delay).

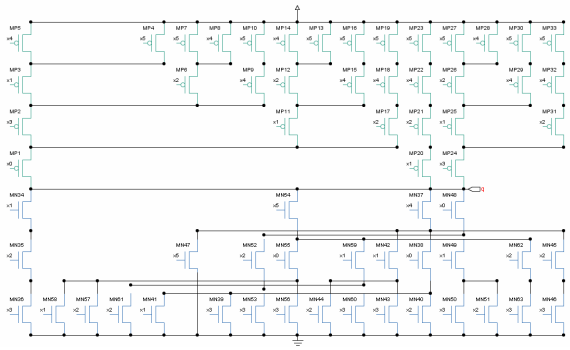

Fig. 7 – Function F1 mapped in NCSP style.

Table I – F1 function implementations comparison.

	freq	power	leakage	delay
NCSP	6,23E+07	3,49E-05	3,09E-10	2,29E-10
LBBDD	4,94E+07	3,67E-05	3,05E-10	2,89E-10
CSP	4,38E+07	6,11E-05	1,76E-09	3,28E-10

Table I shows three figures of merit to three different ways of synthesize the function F1.

The NCSP and LBBDD topologies reach similar results as concern of time delay and consumption. It is due to this methods synthesize the function in a single cell. The CSP method implemented by ELIS tool has a higher power dissipation and delay because synthesizes the function in a grater number of cells. In the frequency operation aspect the NCSP approach reaches higher speed than both other methods. This could be explained by the advantage of the method in the propagation delay. The Fig. 8 shows a plotting of relative data present in Table I.

5. CONCLUSION

As presented in this work, cell characterization could be quickly obtained if an appropriate methodology is applied. The known resources as a simulation tool could be a powerful aid in this task and in the different synthesis methods evaluation. As a future work, other cells will be tested, including those ones that compose mapped circuits by technology mapping tools, and close the test flow with the automation of the input vector sets allowing to test all input-vector present in N set.

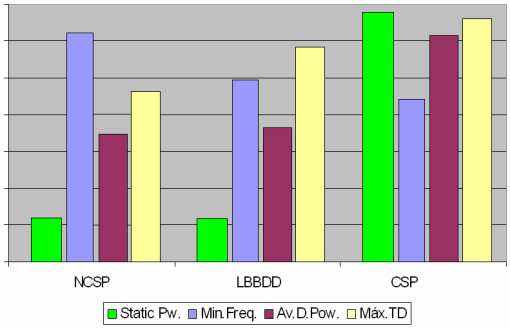

Fig. 8 – Relative results of F1 implementations.

6. REFERENCES

[1] HSPICE Home Page http://www.synopsys.com/products/mixedsignal/hspice.

[2] F. R. Schneider, A. I. Reis, "New CMOS Logic Style Base don the Minimum Theoretical Number of Transistor in Series" Norchip Conference, 2006, pp.85-88

[3] M. Avci and T. Yildirim, "General design method for complementary pass transistor logic circuits". Electronics Letters, Vol.: 39 , Number: 1 , 9 Jan. 2003. Pages:46 – 48.

[4] N.H.E. Weste and K. Eshraghian, Principles of CMOS VLSI Design: A System Perspective, Addison Wesley, Boston, MA, USA, 1999.

[5] F.R.Schneider, R.P.Ribas, S.S.Sapatnekar, A.I.Reis, "Exact lower bound for the number of switches in series to implement a combinational logic cell", *ICCD 2005*, pp. 357 – 362.

[6] F.R.Schneider, A.I.Reis, "Fast CMOS Logic Style Using Minimum Transistor Stack for Pull-up and Pull-down Networks", *IWLS 2006*, pp. 134 – 141.

[7] L.S.da Rosa Jr., F.Marques, T.M.G.Cardoso, R.P.Ribas, A.I.Reis. "BDDs and transistor networks with minimum pull-up/pull-down chains", *IWLS 2006*, pp. 142 – 149.

[8] M. Bhushan, M.B. Ketchen, S. Polonsky and A. Gattiker, "Ring Oscillator Based Technique for Measuring Variability Statistics", ICMT, 2006.

[9] ELIS Home Page http://www.inf.ufrgs.br/lagarto