DESIGNING PROTOTYPE OF THE SPARTAN II FPGA SLICE WITH THE CURRENT-MODE GATES

Oleg Maslennikow, Robert Berezowski, Przemyslaw Soltan and Magdalena Rajewska
Technical University of Koszalin, POLAND, e-mail: oleg@ie.tu.koszalin.pl

Abstract - The paper deals with the problem of designing main FPGA-cell blocks with the current-mode gates – novel digital gates operating with constant, continuous power supply current. Using proposed approaches to designing digital current-mode circuits, the current-mode prototypes of several digital circuits and the Spartan II slice are designed. The obtained circuits are characterized by smaller hardware overheads and switching noise level in comparison with the similar circuits based on the classical voltage type of gates. It proves the possibility of realization of field programmable mixed analog-digital array on a single die.

Index Terms - mixed analog-digital system, switching noise, current-mode gate, current-mode gate logic, field programmable array, system-on-chip.

I. INTRODUCTION

Modern application specific systems (for example, signal processing systems) compose both digital and analog parts, where first part usually is the specialized parallel processor, while analog part is the preprocessing and interface unit between digital part and external world [1]. Advances of the modern VLSI technology permit to implement such mixed systems on a single die. Moreover, the general trend is toward solutions, which guarantee high density and easy design. Reconfigurable arrays like to field programmable gate arrays (FPGA) [1, 2] are examples of these solutions. The recent birth of commercial interest in FPAs has led to the development of programmable analog arrays (FPAA) [3]. This enables to design and to implement on a common semiconductor substrate the whole programmable mixed analog-digital system.

However, the problem of influence of digital part to the analog part of the mixed system must be solved during system designing. The main problem is the substrate interferences. Switching transients (noise) of the digital part can perturb the analog part of a system by means of coupling through the substrate. There are several known solutions for substrate interference reduction - the use of physical separation of analog and digital circuits, guard rings, and a low inductance substrate bias. Each of these methods has own advantages and drawbacks. Another alternative approach for minimizing substrate cross-talk, is based on the implementation of the mixed system digital part with the current mode gates [4]. Due to the nearly constant value of power supply current at the different gate states (for example, logical "0" and "1"), the level of its noise is essentially lower in comparison with the classical voltage type gates. Moreover, based on the current-mode gates, several digital circuits were designed, which are characterized by lower hardware overheads (number of gates) in comparison with their prototypes constructed with classical voltage type gates [4, 5]. Note that physical and logical properties of the current-mode gates differ from corresponding properties of classical voltage-mode gates. Therefore, in this paper, we apply the proposed in ref. [4] design approaches to designing of current-mode the basic block of the Xilinx FPGA cells – the Spartan II FPGA’s slice. The paper purpose is the investigation of the possibility of designing and realization on a single die the whole programmable mixed analog-digital system.

2. CURRENT-MODE GATES AND OPERATIONS OVERVIEW

Fig. 1 represents a basic concept of the current-mode gate [4]. This circuit operates with a continuous, constant current drawn from the power supply and generates low values of current and voltage signals, which fulfill the requirement for minimizing the substrate interferences. Using a simple saturation-mode transistor model, the equation for the output current may be represented by the following form:

\[ I_o = I_q - K_2 \left( \frac{I_t}{K_1} + V_{T1} - V_{T2} - V \right)^2; \]

\[ K_i = \frac{K_i W_i}{2 L_i}, \quad i = 1,2 \]  \hspace{1cm} (1)

where: \( V_{T1}, V_{T2} \) – threshold voltages of n-MOS transistors. In this case, the output current \( I_o = I_q \) means the logical „1” at the output „out”. Logical „1” at the input corresponds to the value:

\[ I_i \geq K_1 \left( V + \sqrt{\frac{I_q}{K_2}} \right)^2. \]  \hspace{1cm} (2)
Then the output current \( I_o = 0 \) means the logical „0” on the output of the gate. Therefore, the circuit in Fig.1,a performs logical inversion (NOT-operation). Note, that when more than one excitation of value, which can fulfill equation (2) is given, then circuit in Fig. 1 performs logical negation of alternative (NOR-operation). Details of current-mode gate conception and electronic parameters of several current-mode gates are represented in the ref. [4,5].

There are four types of gates in the current-mode gate technique. The gate of the first type is named an inverter. It implements the logical function \( Y_1 = \overline{X} \) of the inversion in according to the following expression:

\[
Y_1 = \overline{X} = \begin{cases} 1 & \text{if } X = 0, 1, 2, 3, \ldots \\ 0 & \text{if } X = 1, 2, 3, \ldots 
\end{cases}
\tag{3}
\]

Gates of the second type are named the anti-inverters. They implement the logical function \( Y_2 = \overline{\overline{X}} \) of the anti-inversion in according to the following expression:

\[
Y_2 = \overline{\overline{X}} = \begin{cases} 0 & \text{if } X = 0, 1, 2, 3, \ldots \\ -1 & \text{if } X = 1, 2, 3, \ldots 
\end{cases}
\tag{4}
\]

The third type of the current-mode gate is named an inverter-inverter (or double-inverter). Its carried out logical function \( Y_3 = \overline{\overline{X}} \) is represented below:

\[
Y_3 = \overline{\overline{X}} = \begin{cases} 0 & \text{if } X = 0, 1, 2, 3, \ldots \\ 1 & \text{if } X = 1, 2, 3, \ldots 
\end{cases}
\tag{5}
\]

The fourth type of the current-mode gate is named an inverter - anti-inverter. It implements the logical function \( Y_4 = \overline{\overline{X}} \) which is represented by expression (6).

\[
Y_4 = \overline{\overline{X}} = \begin{cases} -1 & \text{if } X = 0, 1, 2, 3, \ldots \\ 0 & \text{if } X = 1, 2, 3, \ldots 
\end{cases}
\tag{6}
\]

Moreover, all current-mode gates have only one input, while the arbitrary gate may to contain several outputs, possibly, different types. For example, the graphical representation of the current-mode gate with the inverter \( Y_1 \), anti-inverter \( Y_2 \), double inverter \( Y_3 \) and inverter - anti-inverter \( Y_4 \) outputs is shown in Fig.2.

Therefore, there are only four types of elementary operations in the current mode logic inversion and double-inversion (scallion) operations. The addition operation corresponds, at the physical level, to the addition of currents, each from which represents the value of the corresponding operand. In the functional level this means the association of all operand lines into one node. Similarly, a arithmetic subtraction operation in this techniques, at the physical level, is performed by the subtraction of currents. At the functional level, this means (for example, for expression \((X-Y)\)) the association the line of the operand \( X \) with the output of the anti-inverter gate connected to the line of the operand \( Y \), where \( Y \in \{0,1\} \). Examples of realization of operations \((X+Y)\) and \((X-Y)\) are shown in Fig. 3.

\[
\begin{align*}
Y &= X + Y \\
Y &= X - Y
\end{align*}
\]

Fig. 2. Current-mode gate with four different output types

Fig. 3. Realization of addition and subtraction operations in the current-mode technique

It follows from the expressions (3) – (6), that arbitrary logical variable in this logic is (in a general case) a multi-valued one. Moreover, the value of the variable (or function) appeared on any gate output belongs to the set \{-1, 0, 1\}, while the value of the variable appeared on any gate input (for example, as a result of an addition or subtraction operations) belongs to the set of integer numbers from the interval \([-\infty, \infty]\). Due to such logical properties, the Boolean algebra identities are not suitable for the current-mode algebra. However, it has been proved in the papers [4,5], that all Boolean operations can be realized with current-mode gates.

Using proposed approaches, the functional schemes of the several current-mode digital circuits - adders, decoders, multiplexers, triggers, counters and some others were designed. The obtained circuits are characterized by smaller (up to 35%) hardware overheads in comparison with the similar circuits based on the classical voltage type gates. As an example, the current-mode version of one-bit adder is represented in Fig.4, where outputs of sum \( s_i \) and output carry bit \( c_{i+1} \) are the following logic functions:

\[
c_{i+1} = a_i + b_i + \hat{c}_i \quad \text{and}
\]

\[
s_i = c_{i+1} + (a_i + b_i + \hat{c}_i) + \overline{a_i} + \overline{b_i} + \overline{\hat{c}_i} .
\tag{7}
\]

Fig. 4. The 6 gates version of the one-bit adder

3. MAIN BLOCKS OF XILINX SPARTAN II SERIES FPGA

The Spartan ® -II Field-Programmable Gate Array family has a regular, flexible, programmable architecture of Configurable Logic Blocks (CLBs), surrounded by a perimeter of programmable Input/Output Blocks (IOBs) (see Fig. 5). Spartan-II FPGAs are customized by loading configuration data
into internal static memory cells. Stored values in these cells determine logic functions and interconnections implemented in the FPGA.

Fig. 5. Simplified block diagram of Spartan II series FPGA’s

These values reload if necessary to change the function of the device.

The basic building block of the Spartan-II CLB is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and a storage element (see Fig. 6). The output from the function generator in each LC drives both the CLB output and the D input of the flip-flop. Each Spartan-II CLB contains four LCs, organized in two similar slices; a single slice is shown in Fig. 6. In addition to the four basic LCs, the Spartan-II CLB contains logic that combines function generators to provide functions of five or six inputs.

Spartan-II function generators are implemented as 4-input look-up tables (LUTs). Four independent inputs are provided to each of two function generators (F1-F4 and G1-G4) of single slice. These function generators, with outputs labeled O, are each capable of implementing any arbitrary defined Boolean function of four inputs.

The propagation delay is therefore independent of the function implemented. In addition to operating as a function generator, each LUT can provide a 16 x 1-bit synchronous RAM or ROM. Furthermore, the Spartan-II function generator can also provide a 16-bit shift register that is ideal for capturing high-speed or burst-mode data. This mode can also be used to store data in applications such as Digital Signal Processing.

4. DESIGN OF CURRENT-MODE PROTOTYPE OF THE SPARTAN II FPGAs FUNCTION GENERATOR

The table with LUT operating modes is showed in the Tabl. 1, where X denotes the arbitrary value from the set \{0,1\} and Q_i denotes the state of i-th RAM/ROM cell.

Analysis of the function generator operating shows that it may be composed of \(2^n\) cells one-bit FIFO-buffer (where \(n\) is the number of LUT inputs, \(n=4\) for F’ and G’ generators), 4-inputs write decoder and \(2^n\) inputs multiplexer. The example of such function generator realization is represented in the Fig. 6. It composes of 16 D-triggers 1-16 array, which are a base of the 16-cell one bit FIFO block, the multiplexer MUX, write decoder DC and some logic controlled by corresponding mode bits FIFO, RAM, ROM, LUT of configuration memory.

Fig. 6. Simplified block diagram of Spartan II FPGA slice
TABLE 1. Function generator operating modes

<table>
<thead>
<tr>
<th>N</th>
<th>Operation mode</th>
<th>R/W</th>
<th>G1 - G4</th>
<th>Clk</th>
<th>Out</th>
<th>In</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Function generator</td>
<td>X</td>
<td>Function arguments</td>
<td>X</td>
<td>function value</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>FIFO</td>
<td>X</td>
<td>XXXX</td>
<td>X</td>
<td>Input data</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Write RAM</td>
<td>0</td>
<td>Address</td>
<td>X</td>
<td>Input data</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Read RAM</td>
<td>1</td>
<td>Address (i)</td>
<td>X</td>
<td>(Q_i)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>ROM</td>
<td>X</td>
<td>Address (i)</td>
<td>X</td>
<td>(Q_i)</td>
<td></td>
</tr>
</tbody>
</table>

In the function generator mode, serial loading (with clock signals CLK) the values of target logical function \(Y(G4,G3,G2,G1)\) from input In to corresponding D-trigger is carried out during FPGA configuration. For example, the value of the function \(Y(0,0,0,0)\) must be written in the D-trigger 1, the value of function \(Y(0,0,0,1)\) must be written in the D-trigger 2, etc. During normal operating in the function generator mode, all D-triggers save the values of the target function, and the code on the inputs G4-G1 determines the number of D-trigger, which is connected to the generator output OUT.

Thus, in order to deriving of the current-mode prototype of this function generator circuit, the current-mode prototypes of it blocks such as D-trigger (see Fig.7), decoder and multiplexer circuits have been designed in the ref. [4] using the proposed procedure to designing of binary current-mode circuits. The current-mode prototype of whole function generator circuit is represented in the Fig. 8.

5. CONCLUSIONS

Using proposed approaches to designing digital current-mode circuits, the current-mode prototypes of several digital circuits as well as the main basic block of FPGA cell – the look-up-table have been designed. The obtained circuits are characterized by smaller hardware overheads (up to 20%) and switching noise level (up to 10 times) in comparison with the similar circuits based on the classical voltage type of gates. Design and realization of a part of the full-custom ASIC CMOS-chip, which consists of above mentioned current-mode circuits has been performed after verification of their VHDL-models. Experimental verification of all current-mode circuits proved the correctness of their operating and proved the possibility of realization of field programmable mixed analog-digital array on a single die.

REFERENCES


The work is supported by the grant KBN 7T11B 004 20.