ALGORITHM FOR LOW POWER XOR GATE DECOMPOSITION

Peeyush Barskar

Sukhoi State Technical University of Gomel, Republic of Belarus

Scientific Adviser I. A. Murashko

Abstract. An analysis algorithm for the power consumption of a multi-input "exclusive-OR" gate, based on two, three, four and five-input XOR elements, is presented for the case when changes in logic levels at the inputs of the gate occur principally at different instants of time. The upper and lower bounds of the switching activity for different variants of realization of a multi-input XOR gate are obtained.

Introduction. Currently, due to rapid progress in the field of semiconductor integrated circuit technology, in particular the transition to nano-electronic technologies, new tasks arise for the logical synthesis of computing devices implemented on the basis of these technologies. One of such tasks is the development of methods for designing digital

devices with low power consumption. The urgency of this task is determined by the following main factors; the presence of many applications (portable personal computers, mobile communications, digital audio and video equipment), which must combine high speed with low power consumption; the need to reduce the level of energy consumption in order to achieve the necessary duration of autonomous work; the need to reduce the power consumption to solve the problem of heat dissipation, as this determines the weight and size of the devices; the need to reduce power consumption in order to solve the problem of performing effective testing of digital devices [1] (studies show that during testing, power consumption, and accordingly, dissipated power, can increase two to three times). Power consumption can be found so [2]:

$$P = 0{,}5V_{dd}^2 f_{CLK} \sum_{i=1}^{n} C_i^L WSA_i,$$

where C_i^L is the physical capacitance at the output of the node, V_{dd} is the supply voltage, *WSA* (referred to as the switching activity) is the average number of output transitions per clock cycle, f_{CLK} is the clock frequency, n is a number of nodes. We assume the zero delay models where gate delays are assumed to zero. Also we assume that load capacitance of every nodes is equal. We also assume that supply voltage and clock frequency is constant. Thus we must calculation switching activity for every node for estimation of power consumption circuits.

Algorithm for minimum switching activity. Power consumption at algorithm for switching activity related to properties of that particular algorithm techniques. So it should carefully selected for lowering the power consumption. For lowering the power, algorithm should be such that it should minimum number of switching requirements. That algorithm is more useful which have minimum number of operation because it will require less hardware. By increasing concurrency we can increase efficiency of that device.

In the past we synthesis of two-input XOR gate for minimum switching activity but we propose a novel low power Multi-input Algorithm for Switching Activity Reduction through operant decomposition [3]. For simplicity throughout this paper, we are proposing new algorithm to any type of *d*-input XOR gates, minimum switching activity following recursive algorithm can be used: To synthesize a multi-input XOR gate with a minimum switching activity, the following recursive algorithm can be used:

1. Find the output switching activity $WSAout = nx$.

2. Imagine the output switching activity as a sum of n numbers $WSAout = n_1 + n_2 + \ldots + n_d$ in such a way that the following restrictions are fulfilled:

2.a – every time output (root node) divided by d number of childs node,

2.b – all numbers are shareable by x in form of natural numbers,

2.c – in the values of these numbers must be equal or different combination by d in the amount of x. Let n is number of inputs multiple-input XOR gate, d is input type of XOR gate, i is natural number (*1,2,3*...), which is needed for modular arithmetic. Then we are following below table to finding combination of particular nodes input (table).

2.d – we are using above table to find input switching activity for child node in combination of d numbers, in this combination we are searching for greater or equal input value as like; ($n_1 \geq n_2 \geq \ldots \geq n_d$). Then we are using largest input switching activity value to subtract remaining all input switching activities value to fulfill this condition ($|n_1 - n_2 - \ldots - n_d| = 0$ *or* $|n_1 - n_2 - \ldots - n_d| = x$).

Formulas for finding child nodes input switching activity

WSA_{out}									
d	*4*				*3*			*2*	
n	*3*i+1*				*2*i+1*			*i+1*	
i mod d	*1*	*2*	*3*	*0*	*1*	*2*	*0*	*1*	*0*
n_i	$[n/d]$	$[n/d]$	$[n/d]$	$[n/d]$	$[n/d]+1$	$[n/d]+1$	$[n/d]$	$[n/d]$	$[n/d]$
n_1	n_i	n_i	n_i-1	n_i+1	n_i	n_i-1	n_i	n_i	n_i
n_2	n_i	n_i	n_i-1	n_i+1	n_i	n_i+1	n_i	n_i	n_i+1
n_3	n_i	n_i	n_i+2	n_i+1	$n-n_1-n_2$	$n-n_1-n_2$	n_i	–	–
n_4	n_i	$n-n_1-n_2-n_3$	n_i+2	$n-n_1-n_2-n_3$	–	–	–	–	–

3. If n_1 (n_2,... n_d) is equal to *0* or *x*, then the algorithm's work ends for it, otherwise $WSA_{out} = n_1$ ($SWA_{out} = n_2$, ...n_d) and for this value we repeat steps 2 and 3 of the algorithm.

Illustration. When searching for switching activity value, the algorithm needs to know the associated basic input switching activity (*x*) and explores the network in several steps. Each step will find nodes that are closer to the *x* until the contacted node value zero or equal in fig. 1.

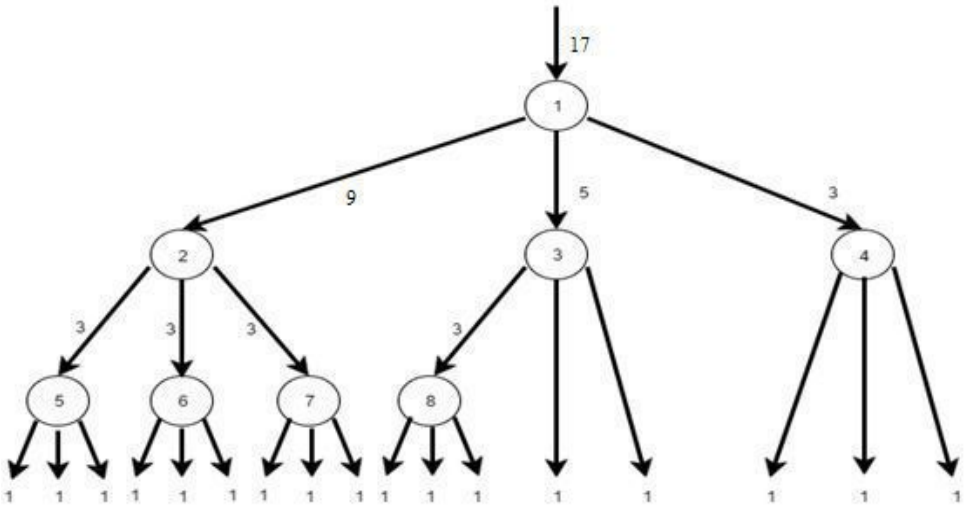

Fig. 1. Example for decomposition 17-input XOR gate with type of three-input XOR gates with minimal switching activity

We create tree is a (possibly linear) data structure made up of nodes or vertices and edges without having any cycle. Then the tree that is consists of a root node and potentially many levels of additional nodes that form a hierarchy. Then we assume root node is output switching activity, we back down root node to child node and we compare to input switching activity that time it is equal to *x* or zero, we quit this process another voice child node converse notion of a parent and repeat this process.

In this example we used design of a 17-input XOR gate (*n=17*) type of *3*-input (*d=3*) with a minimum switching activity is shown in figure. Let the switching activity of all inputs *x* be the same and equal to 1. Then:

1. Find the switching activity of the output $WSA_{out} = nx = 17*1=17$ (Fig. 1).

1.2. Imagine $WSA_{out} = n_1 + n_2 + n_3$. Taking into account table 1 to finding combination of this nodes input switching activity, we obtain $n_1 = 9$, $n_2 = 5$ and $n_3 = 3$ (node 1 in Fig. 1).

1.3. The values of n_1, n_2 and n_3 is not equal to 1, so for each of them we repeat steps *2* and *3*: $WSA_{out} = n_1 = 8$.

2.2. Imagine again $WSA_{out} = n_1 + n_2 + n_3 = 3+3+3$ (for node 2 in Fig. 1).

2.3. This three $n_1 = 3$, $n_2 = 3$ and $n_3=3$ is not equal to 1, so for each of them we repeat steps 2 and 3: $WSA_{out} = n_1 = 3$.

5.2. Imagine once more time $WSA_{out} = n_1 + n_2 + n_3 = 1+1+1$ (node 5 in Fig. 1).

5.3. All values are equal to 0 and 1, so for them the algorithm's work ends.

$WSA_{out} = n_2 = 3$.

6.2 Imagine $WSA_{out} = n_1 + n_2 + n_3 = 1+1+1$ (node 6 in Fig. 1).

6.3. All values are equal to 1, so for them the algorithm's work ends.

$WSA_{out} = n_3 = 3$.

7.2. Imagine $WSA_{out} = n_1 + n_2 + n_3 = 1+1+1$ (node 7 in Fig. 1).

7.3. All values are equal to 1, so for them the algorithm's work ends.

$WSA_{out} = n_2 = 5$.

3.2. Imagine $WSA_{out} = n_1 + n_2 + n_3 = 3+1+1$ (node 3 in Fig. 1).

3.3. The values of n_1 is not equal to 0 or 1, so for it n_1 we repeat steps *2* and *3*. n_2 and n_3 are equal to 0 and 1, so for them the algorithm's work ends.

$WSA_{out} = n_1 = 3$.

8.2. Imagine $WSA_{out} = n_1 + n_2 + n_3 = 1+1+1$ (node 8 in Fig. 1).

8.3. All values are equal to 1, so for them the algorithm's work ends.

$WSA_{out} = n_3 = 3$.

4.2. Imagine $WSA_{out} = n_1 + n_2 + n_3 = 1+1+1$ (node 4 in Fig. 1).

3. All values are equal to 1, so for them the algorithm's work ends.

Conclusion. The paper analyzes the power of multi-input XOR with *d* type gate. Here we reviewed the different low power algorithm at each level of XOR gate design. It is shown that the implementation of multi-input gate in the cell, XOR is a significant increase in switching activity, which leads to an increase in energy consumption. We use particular algorithm according to specification. Presented algorithm synthesis of multi-gate, which minimizes the buildup of a cart-switching activity and provides examples of minimum implementation.

References

1. Yarmolik V., Murashko I. A peak-power estimation for digital circuits design // Fifth International Conference «New Information Technologies» October 29–31, 2002. – Minsk : BSEU, 2002. – P. 34–38.
2. Zhou H., Wong D. F. Optimal Low Power XOR Gate Decomposition. – Design Automation Conference IEEE, 06 August 2002. – DOI: 10.1109/DAC.2000.855286.
3. Murashko I. Power consumption analysis of XOR based circuits. – Informatics. – 2006, № 1 (9). – P. 97–103. [In Russian]