

MICRO-POWER SIMPLE PROCESSING ELEMENT

Tatjana Nikolić¹, Goran Djordjević¹, and Mile Stojčev¹

¹Faculty of Electronic Engineering, A. Medvedeva 14, 18000 Niš, Serbia, {tatjanas,gdjordj, stojcev}@elfak.ni.ac.yu

Abstract--Nowadays the power consumption of VLSI ICs is considered as one of the most challenging problem related to high performance chip design. Numerous low power techniques have been proposed during the last 15 years [1]. In this paper the structure and instruction set of a simple two-phase processing element (PE), as a constituent of complex VLSI computing systems in form of 1D or 2D processor array, is presented. The micro-power PE's model is described writing an efficient register transfer logic (RTL) code, from a low-power standpoint, using clock gating technique. The PE is implemented in Xilinx Spartan2E FPGA technology. Simulation results show that power reduction of up to 10 times, in respect to standard (without clock gating) PE design solutions, at cost of moderate area overhead, can be achieved using micro-power (with clock gating) PE.

Index terms-- low power, clock gating, processing element

1. INTRODUCTION

According to the Moore's Law, integrated circuit densities and operating speeds continue to go up in unabated fashion. The result is that chips are becoming larger, faster and more complex and because of this, consuming increasing amounts of power. Thus, reducing power dissipation is a design goal for all types of VLSI circuits since excessive power dissipation results in increased packaging and cooling costs as well as potential reliability problems.

In order to achieve high performance VLSI computing systems three primary goals have to be successfully solved. The first one relates to high-speed computational PE capability, the second one is with an efficient data transfer among computational building blocks, and the third one is concerned with micro-power consumption. Processor arrays, as a

regular arrangement of PEs, are good candidate accelerator-architectures that are used in many VLSI computing systems with aim to achieve high computational and communication performance. A challenging problem which should be taken into consideration by the VLSI designers now relates both to increase performance and improve energy efficiency. The architectural PE design for low-power is in focus of our interest in this paper.

In particular we will consider RTL synthesis of a simple micro-power PE which represents a constituent of a complex VLSI computational system.

2. POWER CONSUMPTION IN CMOS VLSI

Power consumption in digital CMOS VLSI circuits consists of three components.

$$P = P_{dyn} + P_{sc} + P_{st} \quad (1)$$

The dynamic power component, P_{dyn} , is related to the charging and discharging of the load capacitance at the gate output. The P_{sc} , called short-circuit power, is caused by the short-circuit currents that arise when pairs of PMOS/NMOS transistors are conducting simultaneously. Finally, P_{st} , called leakage or static power, originates from subthreshold currents caused by transistors with low threshold voltages and from gate currents caused by reduced thickness of the gate oxide. For submicron technology (e.g., 0.25 μm), P_{dyn} is dominant (20% P_{sc} , 5% P_{st} and 75% P_{dyn}). For deep-submicron processes (e.g., <100 nm), P_{st} becomes more important (35-50% at 90 nm). Design methods for leakage power control are currently subject of intensive investigation [1]. Here we will concentrate on dynamic power-reduction in CMOS sub-micron technology.

Dynamic power for CMOS gate working in a synchronous environment is modeled by the following equation:

$$P_{dyn} = \frac{1}{2} C_{out} V_{DD}^2 f_{CLK} \alpha \quad (2)$$

where the term αC_{out} is generally called the switched capacitance, which measures the amount of capacitance that is charged or discharged in one clock cycle. V_{DD} is the supply voltage, and f_{CLK} the clock frequency. Due to quadratic dependence of P_{dyn} on V_{DD} , supply voltage scaling has been the most used approach to power optimization. The major shortcoming of this solution, however, is that lowering the supply voltage effects circuit speed. As a consequence, both design and technological solutions must be applied in order to compensate the decrease in circuit performance introduced by reduced voltage. In addition, the shift of industry from a supply voltage to a smaller one is quite expensive and slow due to, for instance, the compatibility issues of input-output signals with the peripheral circuits.

A similar problem, i.e. performance decrease is encountered when power optimization is obtained through frequency scaling. It happens, very often, that some functional units, in a VLSI circuit do not require peak performance for some operations, therefore selective frequency scaling on such units can be applied, at no penalty in the overall system speed.

In contrast, the reduction of the switching activity or the capacitance for a certain technology depends mainly on the designer's creativity. The reduction of switching activity requires among others a detailed analysis of signal transition probabilities, careful design of circuit nodes with high activity, balanced paths, and selection of appropriate logic style.

Our design choice, presented in this paper, is based on the most popular micro-power architecture technique used in designs today. It is based on clock-gating, whose objective is to stop the clock for some cycles of operation in order to achieve a reduction of the dynamic component of the power.

3. CLOCK-GATING TECHNIQUE

The clock gating technique is extensively used in the design of low-power circuit [2]. It involves dynamically shutting off the clock to portions of a design that are idle or are not performing useful computation. Fig. 1 depicts the concept of the clock gating using an AND gate. The basic idea is to AND the clock with an enable signal, so that the register receives a clock signal only when the enable is high.

In this paper we will consider an implementation of clock gating technique on simple PE as a constituent of 1D or 2D processor array. We define the processor array as an arrangement of PEs in an array (often rectangular) where data flows synchronously across the array between neighbors, usually with

different data flowing in different directions. More details concerning systolic array can be found in [3].

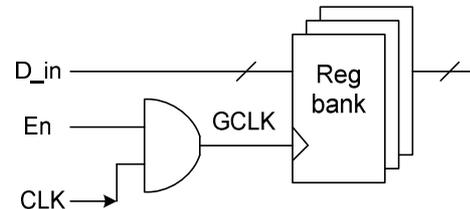


Fig. 1 Clock gating using a latch and AND gate

4. PROCESSING ELEMENT STRUCTURE

The function of a PE, described in this paper, is defined by the following formula:

$$c_{out} = (a_i \otimes b_i) \oplus c_{in} \quad (3)$$

where: a_i , b_i and c_{in} are n -bit input operands; c_{out} is n -bit output operand; symbols \otimes and \oplus correspond to one of the arithmetic/logic operations defined in Table 1.

Table 1. Instruction set

Mnemonic	Description
OR c, a, b	$c = a \text{ or } b$
AND c, a, b	$c = a \text{ and } b$
XOR c, a, b	$c = a \text{ xor } b$
NOT c, a	$c = \text{not } a$
ADD c, a, b	$c = a + b$
SUB c, a, b	$c = a - b$
MIN c, a, b	$c = \min\{a, b\}$
MAX c, a, b	$c = \max\{a, b\}$
CMIN c, a, b	$c = \begin{cases} a & \text{if } a = b \\ \min(\text{rangevalue}) \end{cases}$
CMAX c, a, b	$c = \begin{cases} a & \text{if } a = b \\ \max(\text{rangevalue}) \end{cases}$
NEG c, a	$c = -a$
MUL c, a, b	$c = a * b$
NOP	no operation
others operations	reserved for future use

Having in mind that in one instruction cycle the PE have to perform two operations, our design choice was to use folding technique (see fig. 2).

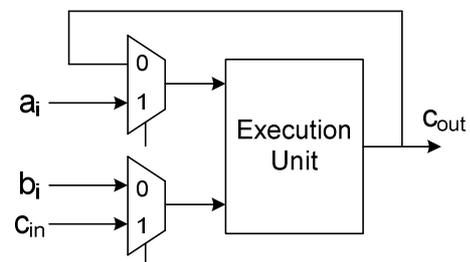


Fig. 2 Processing element

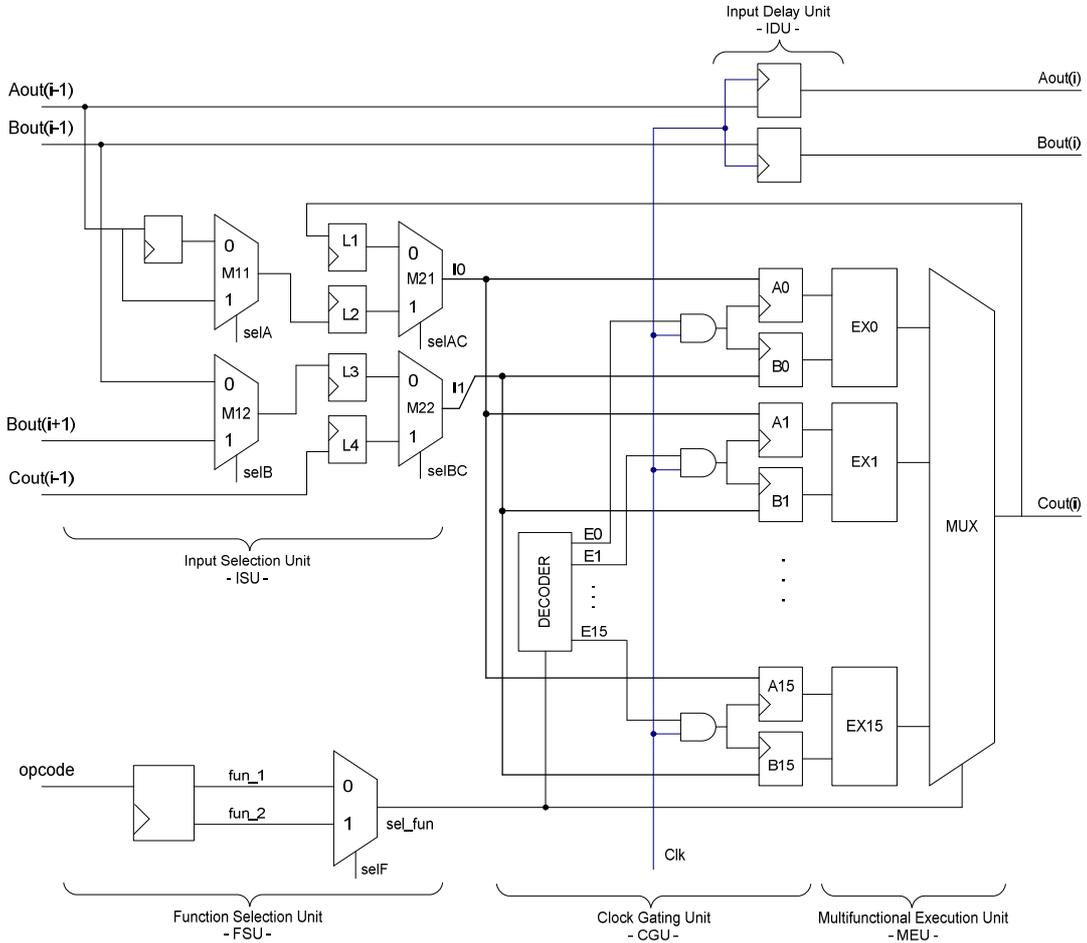


Fig. 3 Micro-power PE hardware structure

As can be seen from fig. 2, the PE consists of one multifunctional execution unit, EX, and two input multiplexers, MUX1 and MUX2. During the first cycle EX performs the operation \otimes on input operands a_i and b_i , and generates output result denoted as c' . During the second cycle EX perform the operation \oplus on input operands c' and c_{in} and generates output result c_{out} , and time delayed input operands, a_i and b_i . MUX1 and MUX2 during the corresponding cycles select correct pair of operands for EX

The internal PE hardware structure is pictured in fig. 3. As can be seen from fig. 3, the PE is composed of five hardware units denoted as: Input Selection Unit, ISU, Function Selection Unit, FSU, Clock Gating Unit, CGU, Multifunctional Execution Unit, MEU, and Input Delay Unit, IDU. The ISU selects input operands. It is implemented as a two stage selector logic. Multiplexers M11 and M12 are constituents of the first selector stage. Latches L1-L4 and multiplexers M21 and M22 are parts of the second selector logic. Latches L1-L4 are used for acquiring input operands. The FSU accepts opcodes of two operations and according to the cycle number (first or second), generates at its output opcode for \otimes or \oplus operation. The CGU is composed of a single decoder and 16 pairs of clock gated latches, one pair for each execution unit. The DECODER, according

to the value of the sel_fun code, activates one of the enable signals, E0 to E15, by which it allows latching of input operands. The MEU is composed of 16 different execution units. At its output the multiplexer MUX selects one of the 16 inputs. The IDU is used for delaying input operands a_i and b_i for one instruction cycle.

Power reduction using the proposal is achieved thanks to the following fact: Instead to switch inputs to all execution units (EX0 to EX15), only inputs to the enabled execution unit are switched. In this way, we significantly reduce switching activity, α , defined in (2) and directly decrease energy consumption.

The PE for which we have implemented the clock gating technique is referred as a *micro-power PE*. The energy reduction is achieved at cost of decreased PE's operating speed and increased Silicon area, i.e. the number of occupied gates.

Another crucial property of the proposal relates to its possibility to implement the method at the RTL level of abstraction. In this way the proposal can be implemented independently of the target technology. In other words, standard HDL code (VHDL, Verilog) as well as standard CAD tools intended for simulation and synthesis can be used

Table 2. Implementation results for Xilinx Spartan2e FPGA

num. of input operands bits	standard PE		micro-power PE		area overhead [%]	power reduction [%]
	gate count	power [mW]	gate count	power [mW]		
8 bit	2665	108.79	4599	20.03	72.6	81.6
12 bit	4680	250.70	7465	51.55	59.5	79.4
16 bit	7089	926.32	10747	125.62	51.6	86.4
24 bit	13240	1845.35	18956	271.40	43.2	85.3
32 bit	21057	7932.00	29064	970.27	38.0	87.8

VHDL code fragment by which we implement clock gating technique is realized as a single process, and is given in fig. 4. For driving the corresponding pair of latches, a CASE statement is used. In term of sel_fun, the outputs I0 and I1 are latched into selected pair of latches, A_i and B_i, i=0, ..., 15.

```

process(clk)
begin
  if(clk'event and clk='1') then
    case(sel_fun) is
      when "0000" => A0<=I0; B0<=I1;
      when "0001" => A1<=I0; B1<=I1;
      .
      .
      .
      when others => A15<=I0; B15<=I1;
    end case;
  end if;
end process;

```

Fig. 4. Clock gating technique in VHDL

5. RESULTS

In order to evaluate the performance of a proposal we have implemented both the standard and the micro-power PE on Xilinx FPGA technology using a software tool ISE 9.1. We have considered and analyzed several *n*-bit PEs which can manipulate with 8-, 12-, 16-, 24- and 32-bits operands. All PEs, at behavioral level, are described using VHDL code. During the phase of HDL code-creation special care was devoted to power consumption optimization by using clock gating technique (see for example fig. 4).

The PEs are implemented on Spartan2E FPGA circuit. Parameters that have influence on power consumption were selected to be identical in all cases. For evaluation of power consumption the Xilinx Xpower tool was used. As performance metrics the area overhead, AO, and power reduction factor, PRF, were used. The metrics AO and PRF are defined as

$$AO = \frac{\mu PE_gate_count - sPE_gate_count}{sPE_gate_count}$$

$$PRF = \frac{sPE_power - \mu PE_power}{sPE_power}$$

where: μPE - micro-power PE; sPE - standard PE.

The obtained results which correspond to standard- and micro-power-PE, for the device XC2S300E from Spartan2E family are given in Table II. PE's hardware complexity is presented in columns 2 and

4, and is specified as an equivalent gate count. In column 3 and 5 PE's power consumption in mW is given. The last two columns include AO and PRF metrics specified in %.

By analyzing the obtained results we can conclude the following: 1) The micro-power-PE has significantly lower power consumption, in respect to standard-PE. For analyzed values of parameter *n*, the PRF is within the range of 80-90%; 2) The proposed method is more area efficient for larger *n*. Namely, with increasing *n* the AO decreases from 72.6% for 8-bit PE down to 38% for 32-bit PE.

6. CONCLUSION

The main intent of this paper is to provide front-end-designers with guidelines and good design practices for writing efficient register transfer logic (RTL) code from a low-power standpoint. It is suitable for engineers that are already familiar with RTL coding for synthesis, but are not necessarily aware of low/power techniques.

Clock gating technique is one of the most successful and widely used techniques for power reduction which is suitable for RTL implementation. In this paper the hardware structure, instruction set, and implementation of a simple micro-power processing element based on FPGA technology are described. The processing element can be used as a basic building block of regular 1D and 2D processing array. For power reduction, the clock gating technique was used. The obtained results show that power reduction up to 87.8%, with area overhead of 38.0% can be achieved.

7. REFERENCES

- [1] C. Piguet, ed, *Low-Power Electronics Design*, Boca Raton, CRC Press, 2005
- [2] R. Mehra, B. Pangrle, *Synopsys Low-Power Design Flow*, pp. 40.1-40.20 in C. Piguet, ed, *Low-Power Electronics Design*, Boca Raton, CRC Press, 2005
- [3] M.O. Esonu, A.J. Al-Khalili, S. Hariri, D. Al-Khalili, "Systolic Arrays: How to Choose them", IEE Proceedings-E, Vol. 139, No. 3, pp. 179-188, 1992.