SJIF Impact Factor: 7.001| ISI I.F.Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 6 | Issue: 1 | January 2021 - Peer Reviewed Journal

DECODING TECHNIQUE FOR LOW POWER DESIGN IN XILINX

Rajdip Das
Department of Electronics and Communication Engineering, Supreme Knowledge Foundation Group of Institutions 1, Khan Road Mankundu, Hooghly, 712123, West Bengal, India

Article DOI: https://doi.org/10.36713/epra6163

ABSTRACT

This research paper is a survey of the current status of research and practice in various disciplines of low power VLSI developments. The paper briefly discusses the rationale of the contemporary, and concentrates on low power design, it presents the metrics and techniques that are used to access the merits of the assorted proposed for the improved energy efficiency. Power dissipation [1], [5] has become an important consideration in terms of performance and scope for VLSI chip design. The research paper describes the decoding strategies, methodology, and techniques for low power system design. Here also we have proposed the decoding technique and compared with silent coding to scale back the transition state and conserved the power which is additionally described during this research.

KEYWORDS—*Power minimization, decoding technique, silent coding.*

I. INTRODUCTION

More and more sophisticated signal processing systems are being introduced with a VLSI chip as the degree of integration continues to expand; these signal processing applications require not only high computing capabilities but also considerable energy consumption. The efficiency and field remain two key design objectives, electricity consumption in today's VLSI device design has become an important issue. Two mains constrain are required for low power VLSI systems [2], [3]. Firstly, with the continuous increase in operating frequency and chip processing power, a big current is needed and thus the heat must be eliminated by proper cooling techniques because of its high energy consumption. Secondly, battery life is limited in handheld electronic devices; low power architecture ensures that these lightweight devices have an increased running period. Diwakar Tiwary [4] gave some ideas about different power management techniques which has opened up a new gateway in the field to meet futuristic challenges.

However, the critical guiding force is that lack of adequate power to detect statistically significant change is becoming a major issue incorporating more transistors on a single chip or a module for multi-chip. Preliminary results have shown the feasibility of energy consumption that would significantly reduce the heat resulting from VLSI circuit packaging, which includes output processes and programs. After a few dormant years recently there has been a rapid turnover of designing a power reduction technique [6] for high-performance criteria. These strategies were able to yield a reduction in the severity of low power architecture, such as the clock gating for complex power reduction or several (multi-V_t) levels to minimize leakage current are the existing technologies [7] which are already well-established and funded.

The main aim of this research is to reduce the no of transitions state. As we decrease the transitions, we can conserve the power in a much more efficient way. In this paper, we have decreased the number of transitions more than the decoding technique. The rest of the paper is organized as follows: The low power decoding methodology are presented in Section III, the proposed architecture of the system in Section IV and experimental results in Section V. Finally, the conclusion and future works are discussed in Section VI.

II. LITERATURE SURVEY

There are different encoding techniques for low power design to scale back the transition, which means to cut back the switching activity so that we can conserve the power. Several techniques are accustomed which implements sequential data and a few are used for parallel data. Some of the examples of coding techniques are differential coding techniques, sparse encoding, and limited-weight codes (LWC) are summarized below.

A. Differential Coding

Differential coding [8] comes under the algebraic coding category which transmits XOR bit in between two

SJIF Impact Factor: 7.001| ISI I.F.Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 6 | Issue: 1 | January 2021 - Peer Reviewed Journal

consecutive values. In certain scenarios, its bus value shows a strong connection between them and due to this striking feature, the hamming distance (HD) between two successive values is either large or small. For example, most of the least significant bits (LSB) are identical, triggering a small HD for two succeeding values. On a similar scale, most significant bits (MSB) show a contrasting feature because of sign extension, which results in large HD. In both the situation, the use of differential coding reduces bit transition state, as a result, almost all the bits within codeword (CW) become 0, and if delayed it becomes 1

.

B. Sparse Encoding and limited-weight codes(LWC)

The Sparse Encoding technique offers an encoding scheme to reduce the number of 1s. A K-LWC is a sparse code illustration that corresponds to a community of CWs weighing up to K [9]. Note that in this term, the weight of a CW is the 1s number. We refer the reader to previous works [10], [11] for a statistical foundation of LWC. LWCs are constrained by the high logical complexity of their encoder and decoder and are not ideal for communication on the chip.

The above coding does not use any redundancy or metadata in either space or time. Space redundancy describes the use of additional bus lines and time redundancy demonstrates the use of extra transfer cycles.

III. METHODOLOGY

We proposed a decoding technique which helps to reduce the transition. Our decoding technique is applicable in sequential data. It helps to consume power as well as save battery life. We implemented this Technique in **'**Xilinx**'.** It briefly explains below.

A. Proposed Decoding Technique

- Case 1: If L_1 gets 1 then it will check from a_0 to a_4. When it will get different bit after two same bit then swapping occurs in that position.
- Case 2: If L_2 gets 1 then it will check from a_3 to a_7. When it will get different bit after two same bit then swapping occurs in that position.
- Case 3: When $L_1 = L_2=0$ then the result will get after decoding will be same as encoding.
- Case 4: When $L_1=L_2=1$ then checking occurs from a_0 to a_7.We summarized it in Table II, as shown below.

TABLE II

Encoding	L1 L2	Decoding
11000110	1 1	10101010
00110001	1 0	01010001
10000110	0 1	10001010
11111111	0 0	11111111

Fig 2. Example of decoding output

B. Proposed Technique Implementation in Xilinx

Xilinx is a synthesis tool that changes over Schematic/HDL design entry into practically proportional rationale logic gates on Xilinx FPGA, with streamlined speed and region. It has some unique features i.e., it has mixed mode HDL design entry, Xilinx ISE permits mix with different synthesis engine from mentor graphics/exemplar, synopsis, and simplicity (XST is restrictive Synthesis Tool of Xilinx.). We have selected Xilinx for speed, reliability, logic density and stability. We have also implemented the encoding, and decoding code in Xilinx. We have included decoding code for reducing the transition state and data redundancy in Xilinx.

Decoding Code in Xilinx

```
entity decode is
Port (I: in STD_LOGIC_VECTOR (0 to 7);
o_new: out STD_LOGIC_VECTOR (0 to 7);
l1: in STD_LOGIC;
l2: in STD_LOGIC);
end decod;
architecture Behavioural of decod is
begin
process(I)
variable j, k: integer: =0;
variable o: std_logic_vector (0 to 7);
begin
o: =I;
if (l1='1' and l2='0') then
j: =0;
for k in 0 to 2 loop
if(j<3) then
if (o(j) = o(j+1) and o(j+1)/=o(j+2) then
o(j+1): =not o(j+1);
o(j+2): =not o(j+2);
--o(j): =o(j);
J: =j+2;
end if;
j: =j+1;
end if;
end loop;
elseif (l1='0' and l2='1') then
j: =3;
for k in 0 to 2 loop
if (j<6) then
if(o(j)=o(j+1) and o(j+1)/=o(j+2) then
o(j+1): = not o(j+1);
o(j+2): = not o(j+2);
j: =j+2;
end if;
end loop;
elseif (l1='0' and l2='1') then
j: =0;
for k in 0 to 5 loop
if(j<6) then
if (o(j) = o(j+1) and o(j+1)/=o(j+2) then
o(j+1): =not o(j+1);
o(j+2): =not o(j+2);
j: =j+2;
end if;
j: =j+1;
end if;
```

SJIF Impact Factor: 7.001| ISI I.F.Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 6 | Issue: 1 | January 2021 - Peer Reviewed Journal

```
end loop;
end if;
o_new<=o;
end process;
end Behavioural;
```

Test Bench of Decoding

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
--Uncomment the following library declaration if using
--srithmatic functions with Signed or Undigned values
--USE ieee.numeric_std.ALL;
ENTITY testdecod IS
END testdecod;
ARCHITECTURE behavior of testdecod ID
---Component Declaration for the Unit Under Test (UUT)
COMPONENT decod
PORT (
I: IN std_logic_vector (0 to 7);
O_new: OUT std_logic_vector (0 to 7);
l1: IN std_logic;
l2: IN std_logic;
);
END COMPONENT;
--Inputs
signal I: std_logic_vector (0 downto 0): =(others=>'0');
signal l1: std_logic: = '0';
signal l2: std_logic: = '0';
--Outputs
signal O_new: std_logic_vector (0 downto 0);
--appropriate port name
--constant<clock>_period: time: = 10ns;
BEGIN
--Instantinate the Unit Under Test (UUT)
Uut: decod PORT MAP (
I=>I,
O_new=>O_new,
l1=>l1,
l2=>l2
);
--Stimulus process
stim_proc: process
begin
--hold reset state for 100ns.
wait for 100 ns;
l1<='1';
wait for 10 ns;
l2<='1';
I<="11000110";
Wait for 10 ns;
---insert stimulus here
--wait;
end process;
END;
```

IV. PROPOSED ARCHITECTURE

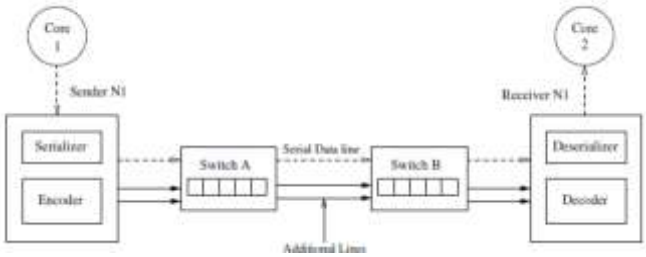

Fig 3. Proposed Architecture of The System

V. EXPERIMENTAL RESULTS

Preliminary functional experiments were performed to test the working functioning of CPU. Fig. 4 denotes the output of the decoding code followed by Fig.5 with RTL Schematic view which is simulated in the Xilinx ISE, and it is seen that when the reset signal is set to '1' value, it puts the CPU in the state of reset 1, which is the first condition of reset order. After the reset signal is fixed to '0', the CPU can have the ability to execute the reset_sequence. From the above factors it is concluded that there are two most excellent signals to watch out for, such as current_state and next_state. It was interesting to note that, when the reset input was set to '1', the CPU remained in reset 1 condition and when the signal reset is fixed to '0', on the rising boundary of the signal clock, current_state proceeds to reset 2 condition. At each signal clock rising edge which causes the CPU to proceed to the next step. A set of experiments were undertaken using different parameters each time until the source array is copied to the destination array. The simulator begins the simulation process. If the simulation is run ahead 100 nanoseconds, we observed that the CPU will start the reset sequence as the information is collected. This test is designed to verify the validity of functionality and the routed design. Note that, instead of one transition, the waveforms that are generated have number of transitions which are settled out. This transition encoding technique gives better results than other encoding techniques.

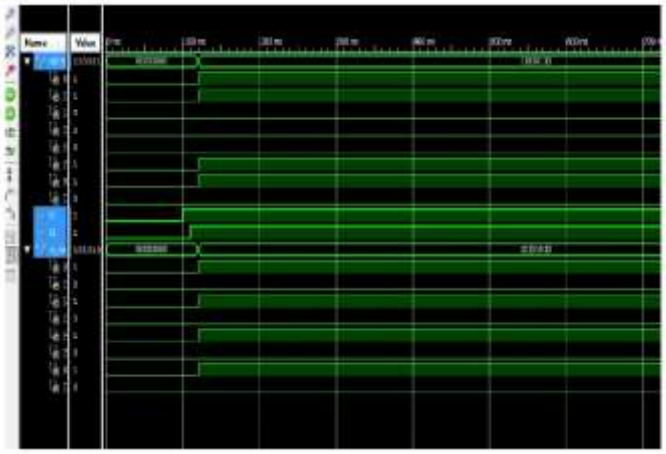

Fig 4. Output of Decoding reducing transtion state

SJIF Impact Factor: 7.001| ISI I.F.Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 6 | Issue: 1 | January 2021 - Peer Reviewed Journal

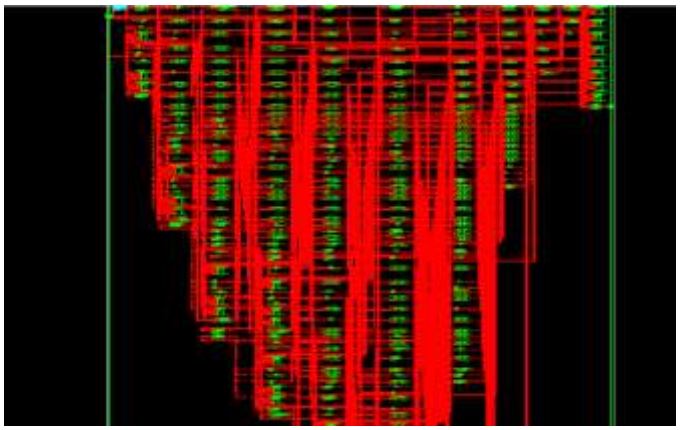

Fig 8. RTL Schematic View

VI. CONCLUSION & FUTURE WORKS

In our proposed decoding technique, the main motto is to attenuate the number of transitions within the serial circuit to cut back dynamic power consumption because of switching activity within the capacitances. The decoding structure has been implemented that always ends up in a significant reduction within switching activity. The framework is tested for various sorts of data streams while accounting for the info correlations. However, the addition of two extra lines causes a touch area and power overhead and it requires special attention.

Future work should focus on the minimization of transition state by correct techniques of the region, and the overhead efficiency. We are still seeking to use such a decoding strategy that reduces the area overhead due to encoder and decoder circuit.

ACKNOWLEDGMENT

I want to acknowledge my deep sense of gratitude to my research guide Ms. Somrita Ghosh who directed and guided me with her timely advice and constant inspiration which eased the task of my research work.

REFERENCES

1. *An Implementation of Integrable Low Power Techniques for Modern Cell-Based VLSI Designs ,Ming-Chung Lee and Herming Chiueh ,doi 10.1109/ICECS.2006.379932.*
2. *V. Sharma, S. Kumar,"Low-Power 1-bit CMOS Full Adder using Subthreshold Conduction Region", International Journal of Scientific & Engineering Research, Vol. 2, June 2011.*
3. *S. Hanson, B. Zhai, K. Bernstein, D.Blaauw, A. Bryant, L.Chang, K. K. Das, W. Haensch, E. J. Nowak, D. Sylvester, "Ultra -Low Voltage, Minimum-Energy CMOS", IBM Journal of Research and Development, Vol. 50, No. 4-5, pp. 469-490, 2006.*
4. *Power Management for Low Power VLSI design, Diwakar Tiwary,Mohd Javeed, 2018 IJCRT Volume 6, Issue 2 April 2018 | ISSN: 2320-2882.*
5. *Various Methodologies for Low-Power VLSI designs, Dr. R. Prakash Rao, JASC: Journal of Applied Science and Computations Volume 5, Issue 10, October/2018, ISSN NO: 1076-5131.*
6. *Rahul. M.Rao, Jeffery L.Burns, Richard B.Brown, "Circuit Techniques for gate and subthreshold leakage minimization in future CMOS technologies" Proc. ISLPED, pp70-73, 2002.*
7. *Prasad Subramanian, "Power management for optimal power design", ESILICON, Corp.2010.*
8. *D. Jahier Pagliari, E. Macii, and M. Poncino, "Approximate differential encoding for energy-efficient serial communication," in Great Lakes Symposium on VLSI. ACM, 2016, pp. 421–426.*
9. *M. R. Stan and W. P. Burleson, "Limited-weight codes for low-power I/O," in International Workshop on low power design, vol. 6, no. 3, 1994, pp. 6–8.*
10. *M. Choudhury, K. Ringgenberg, S. Rixner, and K. Mohanram, "Single-ended Coding Techniques for Off-chip Interconnects to Commodity Memory," in Design, Automation & Test in Europe Conference & Exhibition, 2007. DATE'07. IEEE, 2007, pp.1–6.*
11. *Y. Song, M. N. Bojnordi, and E. Ipek, "Energy-efficient data movement with sparse transition encoding," in Computer Design (ICCD), 2015 33rd IEEE International Conference on, 2015, pp. 399–402.*