FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

Hybrid Data Driven Clock Gating and Data Gating Technique for Better Saving Power in ALU RISC-V

Minh Huan Vo[1*]
[1]*Ho Chi Minh University of Technology and Education; huanvm@hcmute.edu.vn*

***Correspondence:** Minh Huan Vo

ABSTRACT- The study proposes a hybrid data driven clock gating and data gating technique which is applied to ALU in RISC-V. By doing so, the proposed low power technique can improve the power saving efficiency. The proposed low power technique is compared with various low power techniques such as latch-free based clock gating, latch-based clock gating, single data driven clock gating, and single data gating. The results show that the proposed low power ALU saves 46.67% power consumption compared to original ALU. The proposed ALU also shows better saving power than the latch-free based clock gating, latch-based clock gating, sdata driven clock gating, and data gating from 10.84% to 22.23%. The comparison is also implemented on CPU which consists of memory, ALU and control unit. The proposed low power CPU saves 12.11% at least compared to the original CPU. However, the proposed low power CPU is reduced to 15.1% maximum frequency operation compared to the original CPU. The area overhead of the proposed ALU also increased to 33 LUTS (8.2%) and 2 registers (1.6%) compared to the original ALU.

Keywords: ALU; Clock gating; Data driven clock gating; Data gating; RISC-V; Low power technique.

ARTICLE INFORMATION

Author(s): Minh Huan Vo
Received: 03/10/2023; **Accepted**: 28/12/2023; **Published**: 20/03/2024;
e-ISSN: 2347-470X;
Paper Id: IJEER 0310-01;
Citation: 10.37391/IJEER.120133
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-12/ijeer-120133.html

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

1. INTRODUCTION

RISC-V, also known as "RISC-five", is an open-source standards architecture. The RISC-V is designed based on the reduced instruction set architecture (RISC) principles with a flexible architecture to build systems from simple microprocessors to complex core systems [1]. ALU stands for Arithmetic Logic Unit, which is a digital circuit that performs arithmetic and logical operations on binary numbers. In RISC-V, the ALU is a key component of the CPU that performs arithmetic and logical operations on data. The ALU in RISC-V is designed to be simple and efficient, with a reduced instruction set that enables faster execution of instructions. In RISC-V, the ALU supports a range of operations, including addition, subtraction, multiplication, division, logical AND, OR, XOR, and bit shifting. It also supports comparison operations that are used for branching and conditional statements. The ALU in RISC-V is a combinational logic circuit that takes two inputs, performs the requested operation, and produces a single output. The inputs are typically the operands, and the output is the result of the operation. One of the unique features of RISC-V is its use of a register file, which stores the operands and results of ALU operations. This allows the ALU to operate quickly and efficiently, as it can access the registers directly without having to go through memory. The ALU takes input data from the CPU and produces output data based on the instructions given by the CPU. The output data can then be used by other parts of the CPU or sent to other components in a computer system [2-3].

Consumption power is a combination of static power dissipation and dynamic power dissipation [4]. The static power of the device is the leakage power from the transistor. Dynamic power is affected by data input and internal switching operation. Dynamic power is instantaneous and varies at every clock cycle. It depends on voltage level and logic resources. This also includes quiescent current from I/O, clocks and other circuits that need power when in use [5].

Clock gating is a common technique used in many low power circuits to reduce dynamic power dissipation [6-7]. The basic idea of the clock gating technique is that the clock signal should be turned off to introduce an unchanged signal into the circuit when a function block is inactive or in a standby state. As the circuit is not in use, the clock signal should be eliminated. The principle of clock gating techniques is to minimize the number of unnecessary clock switch to flip-flops (FFs) as they are idle or do not update new values. In a typical design, many flipflops usually only update to a new value under a certain condition. Therefore, as FF does not update the new value, the clock on FF can be stopped from transitioning. Currently, FF is still powered and remains at its current value. Clock gating helps reduce dynamic power because it prevents the logic level shifting of the clock from FF [8].

This study focuses on reducing the dynamic power of the ALU block. The more complex the ALU operations, the more energy is dissipated. The clock network is a major source of power dissipation in ALU block. So, we can significantly reduce the power by clock gating [9]. In particular, the basic idea of clock gating techniques is to not provide clocks for circuits when there

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

is no need to use them. Clock gating will avoid switching activities of the flip flops which in turn will reduce the power of the circuit [10].

The next section overviews the related clock gating techniques which have been applied to many circuits. *Section 3* explains the importance and how the ALU-RISC V works, focusing on the implementation of low power clock gating techniques on the ALU block with block diagrams of each technique. A comparative evaluation of low power ALU versions based on the factors of power consumption, overhead area and delay time shown in *section 3*.

The study shows experimental results on Vivado tool to prove the theoretical evaluation statements in *section 4*. The proposed technique brings and synthesizes the most power efficiency to the ALU block. The study applies various low-power techniques to the ALU block to conduct efficiency comparisons. Finally, the conclusion of operant simulation on Vivado and on the FPGA board is shown in *section V*.

2. RELATED CLOCK GATING TECHNQIUES

The main content of *section 2* focuses on researching and presenting basic ideas about low power techniques. The various clock gating techniques including latch free clock gating, latch-based clock gating, data driven clock gating, data gating have been applied to various applications such SRAM, multiplier, ALU, digital signal processing, counter, linear feedback shift register, and so on [11-16]. This section also presents proposed hybrid clock gating and data gating techniques to get the most optimal results in terms of power, and comparing other low power techniques.

2.1 Latch Free Clock Gating Technique

In this technique, the AND logic gate is used to gate the clock for the entire circuit. The AND logic gate has two inputs, one of which is a clock signal, the other is a signal to enable the circuit. Whenever the enable signal is zero, the output of the AND gate is zero, the clock signal applied to the entire circuit will be interrupted.

The main limitation of this technique is that it can generate unwanted clock pulses known as glitch which leads the circuit to operate at those unwanted clocks. That will make the power of the circuit increase and reduce the efficiency of the power optimization. *Figure 1* shows the case of generating unwanted clock pulses in the circuit.

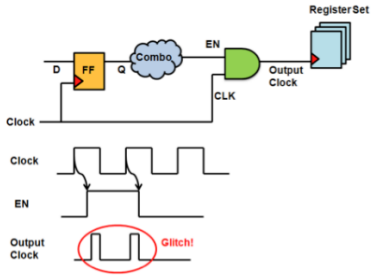

Figure 1: The case where the glitch occurs in the latch-free based clock gating technique

2.2 Latch Based Clock Gating Technique

Latch based clock gating adds a latch designed to hold the enable signal from the active edge of the clock to the inactive edge of the clock. Since the latch circuit captures the state of the enable signal and holds it until a complete clock is generated, the enable signal simply stabilizes around the rising edge of the clock.

The strength of this technique will overcome the weakness of latch-free clock gating which is to avoid unwanted clock pulses in the circuit. *Figure 2* shows the idea of the latch-based clock gating technique.

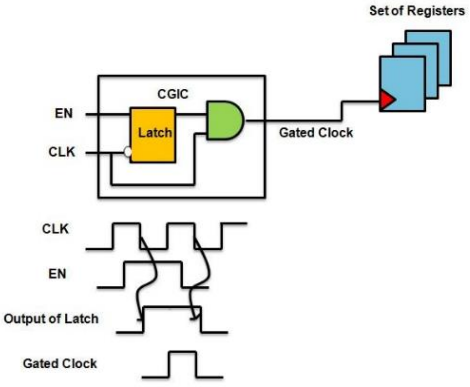

Figure 2: The idea of latch-based clock gating avoids glitch

2.3 Data Driven Clock Gating Technique

The basic idea of data driven clock gating is when the state of the flip flop does not change in the next clock, the clock will be gated. To detect the change of flipflop state, the XOR gate is used. The XOR gate compares the input and output of the flip flop at the current clock pulse and then decides whether to disable (gated) clock signal or not. *Figure 3* shows the idea of data driven clock gating technique.

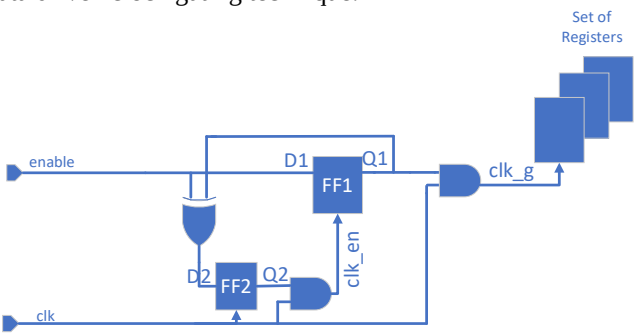

Figure 3: Block diagram of data driven clock gating

Similar to the latch-based clock gating, the data driven clock gating overcomes the weak point of the latch-free clock gating. The data driven clock gating avoids the generation of unwanted clock pulses due to asynchronous operation of the enable signal and clock pulse. In the block diagram shown in *figure 3*, the clock will control FF2 to generate a clk_en pulse. The clk_en pulse controls FF1 to allow the enable signal to be shifted from input D1 to output Q1 and generate a gating clock (clk_g). *Figure 4* shows the idea of data driven clock gating avoiding the generation of unwanted clock pulses (glitch).

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

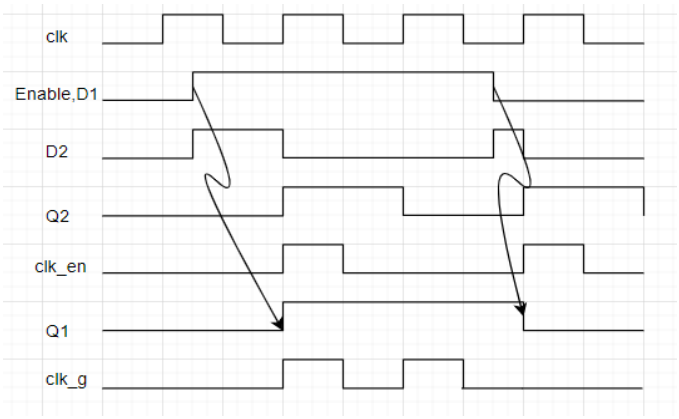

Figure 4: The idea of data driven clock gating avoids glitches

2.4 Data Gating Technique

Besides the clock gating technique helps to reduce the power of the circuit, another technique that also reduce the power of the digital circuit is the data gating technique. Data gating focuses on preventing operations that are not required by the entire circuit resulting in a significant reduction in wasted activities in the circuit. Contrary to the idea of the clock gating technique, the data gating technique only provides data to the circuit that is required to operate to compute and output the result. The circuits that do not have the request will not receive the data to perform computation. To get the best power results, the study suggests a method that combines two factors, clock gating and data gating, in which the clock gating technique uses data driven clock gating.

3. CLOCK GATING TECHNIQUES FOR ALU IN RISC

3.1 ALU Architecture

The CPU is mainly composed of three components consisting of memory, the control block and the ALU (Arithmetic Logic Unit). The ALU block is the core of all processors. It is the part where the CPU performs logical arithmetic operations. ALU stands for Arithmetic Logic Unit, which is a fundamental component of a computer's central processing unit (CPU). The ALU performs arithmetic and logic operations on binary numbers, such as addition, subtraction, bitwise AND, bitwise OR, and bitwise XOR. The ALU is responsible for carrying out the instructions of a computer program and performing the calculations required for those instructions. It is an essential part of the CPU that allows the computer to perform complex operations and make decisions based on input data.

The Arithmetic Logic Unit is a fundamental component within a CPU responsible for performing arithmetic and logical operations on binary data. The data pipelines within an ALU are the stages through which data moves and undergoes processing during these operations. These data pipelines commence with the fetching of operands from registers, where the incoming data awaits processing. Upon decoding instructions from the CPU's instruction set, the ALU navigates through distinct pathways tailored for arithmetic, logic, and shift operations. Each operation follows a dedicated pipeline, meticulously executing the specified computation. Control signals orchestrate the sequence and coordination of operations, guiding the data's movement through execution stages, resulting in the generation of computed results. Finally, these outcomes are momentarily stored in output registers within the ALU before being dispatched to memory or other registers, culminating the journey through the ALU's intricate data pipelines.

In modern VLSI technology, ALU sizes are decreasing and becoming more complex to efficiently serve mobile devices. So, the current ALU blocks are designed in such a way that they consume less power and perform operations at high speed and more precisely. The ALU block receives the operation request to be decoded, the output of the ALU block is the result of the operations of the two operands. The ALU block diagram is shown in *figure 5* below.

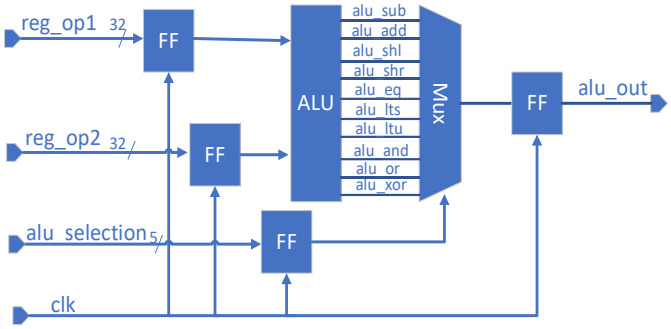

Figure 5: ALU in RISC-V architecture

The ALU block consists of two operands reg_op1 (32 bits) and reg_op2(32 bits), alu_selection signal (5 bits), clock signal, and output of alu_out (32 bits). The alu_out signal is the result output of ALU block. The ALU block operations include summation and subtraction of two operands (alu_sub, alu_add), left and right shift (alu_shl, alu_shr), comparison of two operands (alu_eq, alu_lts, alu_ltu), logical operations (alu_and, alu_or , alu_xor). The following table summarizes the operation of the ALU block in the RISC-V architecture.

Table 1: **Function and operation of ALU block**

alu_selection	Function	Operation
00001	reg_op1 – reg_op2	subtraction of two operands
01110,01111,10000 10001,10010,10011	reg_op1 + reg_op2	Summation of two operands
00110,01001, 01000,01010	$signed(reg_op1)< $signed(reg_op2)	Signed comparison of two operands
00111, 01011 01100, 01101	reg_op1 < reg_op2	Unsigned comparison of two operands
11010, 11011	reg_op1 << reg_op2[4:0]	Left shift

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

00010,00011	$signed({reg_op1[31], reg_op1}) >> reg_op2[4:0]	Right shift arithmetic
11100, 11101	reg_op1 >> reg_op2[4:0]	Logical right shift
11000, 11001	reg_op1 & reg_op2	Logic AND
10110, 10111	reg_op1 | reg_op2	Logic OR
10100, 10101	reg_op1 ^ reg_op2	Logic XOR

The following table shows the explanation for each instruction in the ALU block. The I-format instruction is used to operate between a register and a constant stored in the instruction.

Table 2: Detail instructions for each ALU block instruction

No.	alu_selection	Instruction	No.	alu_selection	Instruction
0	00000	instr_nop	15	01111	instr_auipc
1	00001	instr_sub	16	10000	instr_jal
2	00010	instr_sra	17	10001	instr_jalr
3	00011	instr_srai	18	10010	instr_addi
4	00100	instr_beq	19	10011	instr_add
5	00101	instr_bne	20	10100	instr_xor
6	00110	instr_bge	21	10101	instr_xori
7	00111	instr_bgeu	22	10110	instr_or
8	01000	instr_slti	23	10111	instr_ori
9	01001	instr_blt	24	11000	instr_and
10	01010	instr_slt	25	11001	instr_andi
11	01011	instr_sltiu	26	11010	instr_sll
12	01100	instr_bltu	27	11011	instr_slli
13	01101	instr_sltu	28	11100	instr_srl
14	01110	instr_lui	29	11101	instr_srli

In *figure 6*, the OR gate performs OR operations of the alu_selection signal bits. If the alu_selection input signals are zero, the ALU block enters the IDLE state. The OR gate output is zero and performs AND operation with the original clock. As a result, the clk_g will interrupt and does not give clock switching to the ALU block.

3.2 ALU Latch-Free Clock Gating

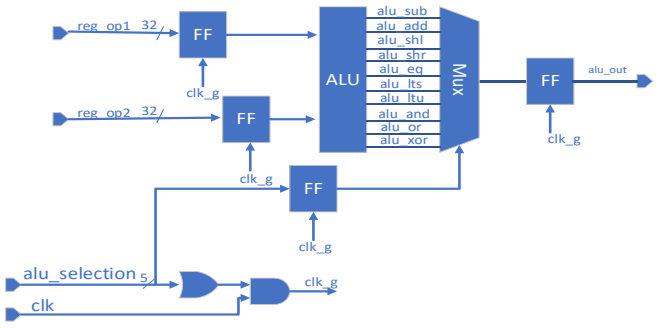

Figure 6: ALU block diagram using latch free clock gating technique

3.3 ALU Latch-based Clock Gating

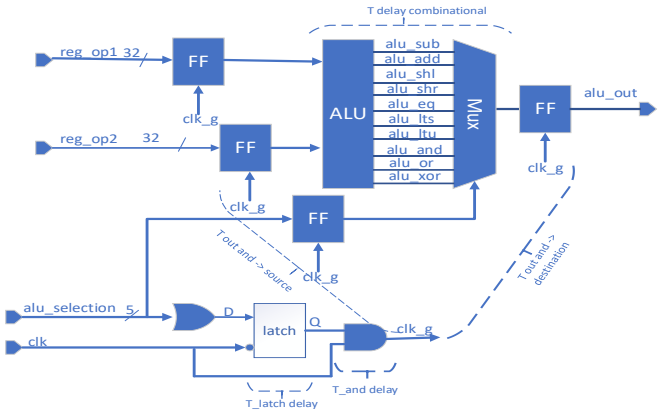

Figure 7: ALU block diagram using the latch-based clock gating technique

Figure 7 shows the idea of applying the latch-based clock gating technique to ALU block. Compared to the latch-free clock gating, the latch-based clock gating will avoid generating unwanted clock pulses. If the output of the OR gate has a logic level of 0, the ALU block will enter the idle state. Otherwise, if the output of the OR gate has a logic level of 1 (*i.e.* the ALU block is still active), the latch circuit will latch that logic value at the falling edge of the clock. The signal Q is the output of the latch circuit that will be ANDed with the original clock to generate clock_gating pulse (clk_g). If the latched value is 0, the clk_g pulse has a value of 0 because the ALU block will enter the IDLE state at this time and no clock is needed for operation. Reversely, if the value is latched to 1, the clk_g signal will be a clock signal and provide a pulse to the ALU block to perform normal computation operations.

3.4 ALU Data Driven Clock Gating

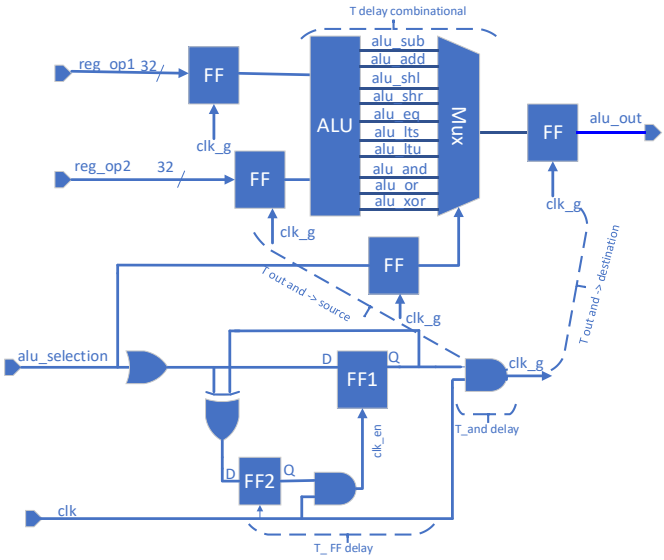

Figure 8: ALU block diagram using the data driven clock gating technique

Figure 8 shows an ALU block diagram using the data driven clock gating technique. Initially the Q signal output of flip flops

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

FF1 and FF2 is 0. As the data from alu_selection input is valid, the D pin of FF_1 will be 1. The circuit uses the XOR gate to detect data change before and after FF1. The output of the XOR gate equals 1 that is put into the D pin of FF2 at the rising edge of FF2. The input D will be shifted to the Q pin of FF2. The circuit uses an AND gate with two input signals. The first signal input is a clock pulse, the second signal input is output of FF2's Q pin. The output of AND gate will be used as clk_en signal to feed FF1 to shift FF1's D input to Q output. The FF1's Q output continues to be AND with the original clock signal to generate a clock gating signal (clk_g) that feeds the entire ALU circuit.

3.5 ALU with Clock Gating Technique

To have a clearer concept of how to apply data gating technique to the ALU block in RISC-V, *figure 9* is a block diagram showing the circuit details of the ALU block. As the ALU block is applied to clock pulse, the data from the two inputs reg_op1 and reg_op2 will be distributed into all computing circuits in the ALU block such as alu_sub, alu_add, alu_shl, alu_shr, alu_eq, alu_lts, alu_ltu, alu_and, alu_or, alu_xor. At each clock pulse, the alu_selection input can only decode one instruction, *i.e.*, only 1 out of 10 results is selected for the alu_out output in a pulse cycle.

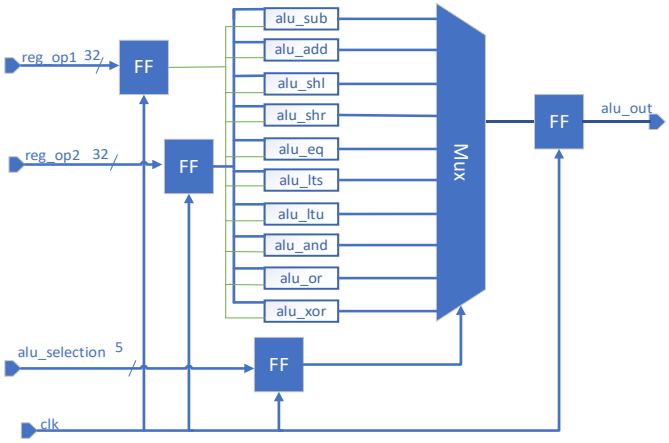

Figure 9: Detailed block diagram of the ALU block

The conventional data gating idea here is to add AND gates in front of the calculation circuits. The AND gate will bitwise decode the instruction signal with two operands reg_op1 and reg_op2. *Figure 10* is an ALU block diagram using data gating technique.

3.6 ALU With Hybrid Data Gating and Clock Gating Technique

The conventional data gating reduces the wasted activities in the circuit. The data gating technique provides data to the circuit to compute to result. Differently, clock gating reduces power consumption by selectively disabling the clock signal to certain circuit elements when they are not actively needed. This technique aims to save power by preventing unnecessary clock pulses from reaching specific parts of the circuitry that are idle or inactive at a given time. The study proposes a hybrid technique which combines clock gating and data gating to save more power consumption.

Because only one command signal is decoded at each clock pulse, only the computation circuit with the corresponding decoded signal has input data to be performed. The other left computation circuits have two data that will be kept 0. For example, assuming that the alu_selection signal decodes into ADD command. Normally, all ten computation circuits have to work and compute operations. At same time, these operation output results come to as inputs of MUX gate. Then, the decoded instruction signal will select the result of the adder circuit in MUX and give the output alu_out. The circuit that applies only data gating technique, only the alu_add circuit receives data from two operands, two inputs in the remaining circuits will equal 0. In order to optimize the power reduction results of the whole circuit, the study has incorporated data gating techniques into clock gating techniques to significantly reduce power. A clock gating circuit including FF1, FF2, AND, and XOR gates generates a clock gating signal (clk_g) that feeds the entire ALU circuit.

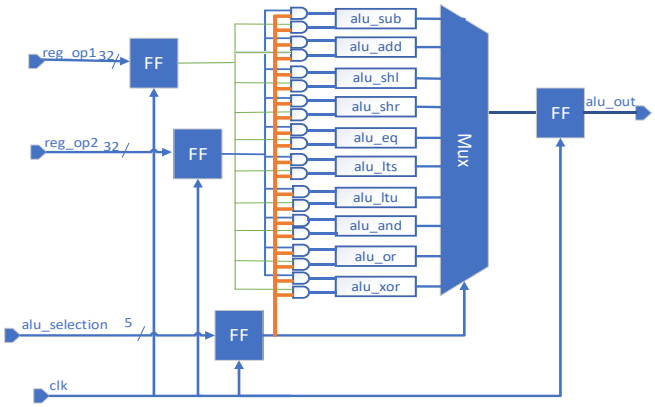

Figure 10: ALU block diagram using the data gating technique

As mentioned, in order to optimize the power reduction results of the whole circuit, the study has incorporated data gating techniques into clock gating techniques to significantly reduce power. The following *figure 11* shows the ALU block diagram that combines two techniques of data driven clock gating and data gating.

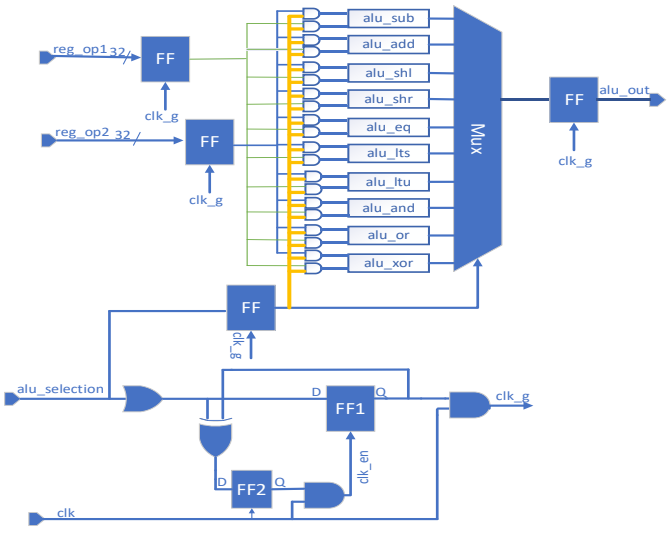

Figure 11: ALU block diagram with hybrid driven clock gating and data gating technique

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

3.7 Timing Analysis

To be able to clearly explain the influence of the two techniques on the timing of the two respective ALU blocks, the formula for calculating the worst negative slack (WNS) and the circuit diagram of each technique is presented. From the block diagram of the ALU block using the latch-based clock gating technique shown in *figure 7* with the source flipflops of reg_op2 signals and destination flipflops of the alu_out signals. The formula is defined to determine WNS of the ALU latch based clock gating shown as following.

WNS (setup)= Data Required Time – Data Arrival Time (1)

Data Required Time = Capture edge time + Destination clock path delay – Clock uncertainty – Setup time (2)

According to the circuit diagram of the latch-based clock gating technique, the values in the corresponding formula are as follows:

Data Arrival Time = Launch edge time + Source clock path delay + Datapath delay (3)

Capture edge time = 10ns = T_{period}.

Destination clock path delay = $T_{latch\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_destination\ (delay)}$ (4)

Clock uncertainty = 0.035ns, Setup Time = 0.025ns, Launch edge time = 0ns

Source clock path delay = $T_{latch\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_source(delay)}$ (5)

Datapath delay = $T_{delay\ combinational}$ (6)

Next, the ALU block diagram using the data driven clock gating technique is shown in *figure 8* with source flipflop of reg_op2 signals and destination flipflop of the alu_out signal.

Capture edge time = 10ns = T_{period}.

Destination clock path delay = $T_{FF\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_destination\ (delay)}$ (7)

Clock uncertainty = 0.035ns, Setup Time = 0.025ns, Launch edge time = 0ns

Source clock path delay = $T_{FF\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_source(delay)}$ (8)

Datapath delay = $T_{delay\ combinational}$

From the above values we can deduce the formula to calculate WNS for each respective technique.

$WNS_{latch\ based\ clock\ gating} = 10 + (T_{latch\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_destination\ (delay)}) - 0.035 - (-0.025) - 0 - (T_{latch\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_source\ (delay)}) - T_{delay\ combinational}$ (9)

$WNS_{data\ driven\ clock\ gating} = 10 + (T_{FF\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_destination\ (delay)}) - 0.035 - (-0.025) - 0 - (T_{FF\ delay} + T_{and\ delay} + T_{out\ and \rightarrow reg\_source(\ delay)}) - T_{delay\ combinational}$ (10)

Subtract *equation (9)* from *equation (10)*,

$WNS_{latch\ based\ clock\ gating} - WNS_{data\ driven\ clock\ gating} = T_{latch\ delay} - T_{latch\ delay} - T_{FF\ delay} + T_{FF\ delay} = 0;$

Inferring that the timing results between the two clock gating techniques are approximately equal if the source and destination of the two techniques are the same. This theorical result is compared with the result from Vivado's timing report in the following tables.

4. SIMULATION RESULTS

In this simulation, various low power techniques are compared to evaluate the efficiency of power saving. The hybrid clock gating and data gating technique is compared with various low power techniques such as latch-free based clock gating, latch-based clock gating, single data driven clock gating, and single data gating. Firstly, the latch-free based clock gating technique uses AND logic gate to gate the clock. The output of AND gate is controlled by enable signal and clock signal. Whenever the enable signal is zero, the AND gate output is zero. Secondly, the latch-based clock gating adds more a latch compared to the latch-free based clock gating technique. The latch gate holds the state of enable signal until a complete clock is generated. Thirdly, single data driven clock gating technique is based on the state of flipflop not changed in next clock. A XOR gate compares the input and output of flipflop to detect the change of the flipflop state. Fairly, the single data gating technique adds AND gate to prevent operations and reduce the activities in circuit. The clock gating output is used to activate the operation circuit. The global clock is gated by various clock gating techniques that reduce the switching activity and dynamic power.

4.1 Power Consumption

The simulation power results are measured on Vivado tool, using Vector SAIF file (Switching Activity Interchange Format). The SAIF method will perform more accurate energy estimation on the design. The Vivado power report assigns switching activity and static probabilities to estimate design dynamic power. Power report will take the SAIF file as input created after the simulation as the basis for estimation. Several low power techniques are compared such as latch-free based clock gating, latch-based clock gating, single data driven clock gating, single data gating, hybrid data gating and driven data clock gating as shown in *table 3*. Here the clock power, signal power and logic power are components of the dynamic power which is consumed by switching activities.

From the results of *table 3*, we conclude that the ALU block power using a combination of clock gating and data gating techniques has the most optimal results. Dynamic power consists of three components clock power, signal power, and

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

logic power. The proposed clock gating saves 46.67% consumption power compared to the original ALU. The latch-free clock gating, the latch-based clock gating, data driven clock gating and data gating saves 35.83%, 35.83%, 35.83%, and 24.44%, respectively. The results in *table 3* are compiled from the power report on Vivado.

Table 3: ALU RISC-V block in Power Optimization Results of Each Technique

	Clock(W)	Signal(W)	Logic(W)	Dynamic(W)	Reduced power (%)
Original ALU	$7.2x10^{-4}$	$3.24x10^{-3}$	$3.24x10^{-3}$	$7.2x10^{-3}$	0%
ALU latch-free clock gating	$6.6x10^{-4}$	$1.98x10^{-3}$	$1.98x10^{-3}$	$4.62x10^{-3}$	35.83%
ALU latch based clock gating	$6.6x10^{-4}$	$1.98x10^{-3}$	$1.98x10^{-3}$	$4.62x10^{-3}$	35.83%
ALU data driven clock gating	$6.6x10^{-4}$	$1.98x10^{-3}$	$1.98x10^{-3}$	$4.62x10^{-3}$	35.83%
ALU data gating	$6.8x10^{-4}$	$2.38x10^{-3}$	$2.38x10^{-3}$	$5.44x10^{-3}$	24.44%
ALU hybrid data gating and driven data clock gating	$6.4x10^{-4}$	$1.6x10^{-3}$	$1.6x10^{-3}$	$3.84x10^{-3}$	46.67%

4.2 Result of Area Utilization

The area results of each technique are summarized in *table 4*. The Table presents the area utilization in ALU consisting of LUT, Register, and Carry 8.

Table 4: Results of area utilization in ALU RISC-V of each clock gating technique

	LUT	Register	Carry8
Original ALU	403	125	10
ALU latch free clock gating	404	125	10
ALU latch based clock gating	405	126	10
ALU data driven clock gating	407	127	10
ALU data gating	432	125	10
ALU hybrid data gating and data driven clock gating	436	127	10

Based on the block diagram of each CG technique, *table 4* shows the original ALU consists of 403 LUTs, 125 registers, and 10 Carry8. The ALU latch-free clock gating increases LUTs from 403 to 404. This is because the latch free clock gating circuit adds OR, AND gates (LUT6) as shown in *figure 6*. The ALU latch based clock gating technique increases LUTs from 403 to 405. Registers increases from 125 to 126. The LUT increases to 2 including OR circuit (LUT5) and AND gate (LUT2). The increase in the Register is due to the latch circuit shown in *figure 7*.

The ALU data driven clock gating increases LUT to 4, from 403 to 407. The register increases to 2, from 125 to 127. This CG causes LUT to increase to 4 including one OR gate, 2 AND gates and 1 XOR gate. The Register increases to 2 consisting of 2 flip flops shown in *figure 8*. The ALU data gating technique increases LUTs from 403 to 432 that is due to the AND gate inserted before the computation circuits shown in *figure 11*.

ALU hybrid data driven clock gating and data driven clock gating is similar to the ALU data driven clock gating technique as shown in *figure 11*. From the aggregate results and block diagram of each CG technique, hybrid data driven clock gating and data gating techniques use the largest resources which consist of overhead of both data driven clock gating and data gating.

4.3 Timing Analysis Results

Timing analysis results of the original ALU block and comparison in each CG technique are summarized in the *table 5*.

Timing analysis results of each CG technique have variation in the range from 0.2ns to 0.3ns. The reason is due to the large number of I/Os of the ALU block. The problem of assigning all the input and output pins to the GPIO on the FPGA board is not possible, which leads to varied results as synthesizing. Vivado tool will create different paths, sources and destinations on its own, resulting in a different cumulative delay compared to the theory presented in *subsection 3.7*.

Table 5: Timing analysis results of original ALU and ALU blocks using various low power techniques

	WNS [ns]	Fmax [MHz]	Maximum Frequency Reduction [%]
ALU	7.5ns	400Mhz	0%
ALU latch free clock gating	7.018ns	335.3Mhz	16.2%
ALU latch based clock gating	7.263ns	365.4Mhz	8.7%
ALU data driven clock gating	7.343ns	376.4Mhz	5.9%
ALU data gating	6.862ns	318.7Mhz	20.3%
ALU data driven clock gating and data gating	7.089ns	343.5Mhz	14.1%

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

4.4 Tradeoff Metrics

From the results presented in *tables 3, 4 and 5*, there is an interrelationship between the three factors of power, area and timing. To reduce the power of the circuit, we have to sacrifice the timing factor and the overhead area by using some more flip flop gates or logic gates for low power techniques. This will also increase the delay of the whole circuit. In *figure 12*, a column chart showing the relationship between the three factors of power, area and timing of the original ALU block and hybrid data gating and data driven clock gating in ALU block.

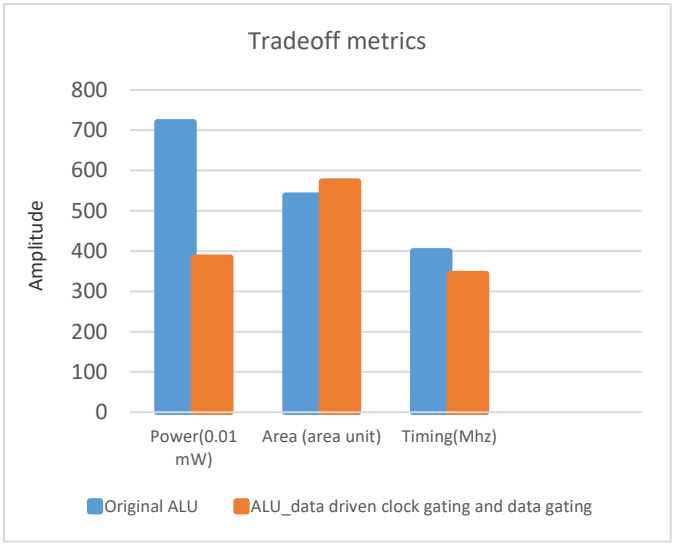

Figure 12: Column chart shows the relationship between power, area and timing

In column chart of *figure 12*, the *y-axis* shows the difference in power, area utilization and timing of the original ALU and ALU blocks using clock gating technique. The *x-axis* represents three metrics consisting of power, area, and timing. In order for the consumption power to decrease, the area utilization will increase, and the timing will decrease.

4.5 CPU Power Measurement Results Using the Proposed Low Power Technique

Table 6 compares the CPU power using the proposed low power techniques with the original CPU which the system consists of two components, a processor (CPU) which consists of ALU and memory to store data temporarily with a simulation time of 282075ns. The combination of data gating and data driven clock gating techniques shows better efficient saving power than original CPU technique. *Table 6* compares the CPU power results before and after using the low power techniques.

Table 6: CPU power comparison with proposed low power technique

Frequency		Clock (W)	Signal (W)	Logic (W)	Dynamic (W)	Power Saving (%)
	Original CPU	3.64×10^{-3}	3.36×10^{-3}	3.08×10^{-3}	0.01008	14.88
100 Mhz	Proposed CPU	2.6×10^{-3}	3.12×10^{-3}	2.86×10^{-3}	8.58×10^{-3}	
200 Mhz	Original CPU	7.36×10^{-3}	6.9×10^{-3}	5.98×10^{-3}	0.02024	15.02
	Proposed CPU	5.16×10^{-3}	6.45×10^{-3}	5.59×10^{-3}	0.0172	
300 Mhz	Original CPU	0.01088	0.01024	9.6×10^{-3}	0.03072	12.11
	Proposed CPU	7.8×10^{-3}	0.0102	9×10^{-3}	0.027	
400 Mhz	Original CPU	0.01476	0.01394	0.0123	0.041	14.73
	Proposed CPU)	0.01064	0.01292	0.0114	0.03496	

5. CONCLUSION

The ALU in RISC-V is designed to lower power consumption. The data driven clock gating is combined with data gating technique to apply to the ALU. By doing so, the proposed technique can improve the power efficiency. The comparison among various clock gating techniques is implemented. The results show that the hybrid data driven clock gating and data gating save 46.67% power consumption compared to the original ALU. The proposed clock gating technique also shows better saving power than the latch-free based clock gating, the latch-based clock gating, data driven CG and data gating from 10.84% to 22.23%. The comparison is implemented on CPU and memory which the proposed low power technique saves 12.11% at least compared to the original CPU. However, the proposed low power CPU n is reduced to 15.1% maximum frequency operation compared to the original CPU. The area overhead of the proposed ALU also increased to 33 LUTS (8.2%) and 2 registers (1.6%) compared to the original ALU.

ACKNOWLEDGEMENTS

This work belongs to the project in 2023 funded by Ho Chi Minh City University of Technology and Education, Vietnam.

REFERENCES

[1] SHAKTI Development Team. RISC-V ASSEMBLY LANGUAGE PROGRAMMER MANUAL PART I. IIT-Madras 2022.

[2] Andrew Waterman, Krste Asanovic. RISC-V Foundation, The RISC-V Instruction Set Manual, Volume I: User-Level ISA Document Version 2.2. Berkeley 2017.

[3] John Winans. RISC-V Assembly Language Programming. Northern Illinois University 2022.

[4] Vo, M.H; Jung, C.M; Lee, E.S; Min, K.S. Dual-switch power gating revisited for small sleep energy loss and fast wake-up time in sub-45-nm nodes. IEICE Electronics Express 2011, 8, 232-238.

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 1 | Pages 238-246 | e-ISSN: 2347-470X

[5] Mai, K.N.T; Vo, M.H. Self-Power Gating Technique for Low Power Asynchronous Circuit. Journal of Institute of Korean Electrical and Electronics Engineering 2018, 22, 548-557.

[6] Tamil C. S; Shanmugasundaram N. Clock Gating Techniques: An Overview. Conference on Emerging Devices and Smart Systems (ICEDSS) 2018; pp. 217-221.

[7] Nirmaladevi, V.; Angel Pd. Design of Modified Data Driven Clock Gating And Look Ahead Clock Gating For Low Power. International Journal of Electrical & Electronics Research (IJEER). 2018, 6(2), 32-36.

[8] Vo, M. H. The Merged Clock Gating Architecture for Low Power Digital Clock Application On FPGA. 2018 International Conference on Advanced Technologies for Communications (ATC) 2018; pp. 282-286.

[9] Kandasamy, N.; Telagam, N.; Devisupraja, C. Design of a Low-Power ALU and Synchronous Counter Using Clock Gating Technique. Advanced Computing and Intelligent Engineering 2017.

[10] Sahu, P.; Santosh, K. A. Comparative Analysis of Different Clock Gating Techniques. 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE) 2020, pp. 1-6.

[11] Kavya, S. P, and Kariyappa, B. S. A Novel Approach for Power Optimization in Sequential Circuits Using Latch Based Clock Gating. International Journal of Electrical Engineering and Technology, 2020, 11(4), 349-359.

[12] Mendez, T.; Subramanya G. N. Design of a Low-power Computational Unit using a Pipelined Vedic Multiplier. 2023 International Conference for Advancement in Technology (ICONAT) 2023; pp. 1-6.

[13] Wimer, S.; Albahari, A. A Look-Ahead Clock Gating Based on Auto-Gated Flip-Flops. IEEE Transactions on Circuits and Systems 2014, 61(5), 1465-1472.

[14] Mohamed S.; Jaison, S. B.; Anto, B. M. Design of low power 16-bit counter with Programmable Combinational Logic and Integrated Clock Gating using 16-nm technology. International Journal of Electronics 2020, 163 - 179.

[15] Nayakulu, A. R.; Kodati; S. P. Low Power Clock Gating Method with Subword based Signal Range Matching Technique. Indian journal of science and technology 2016.

[16] Kumar, C.; Madhavi, B.; Kishore, K. Enhanced Clock Gating Technique for Power Optimization in SRAM. Journal of Automation, Mobile Robotics and Intelligent Systems 2021, 32-38.