Implementation of Glitch-Free Clock Gating Circuit using Dual Muller C-Elements

eSim Research Migration Project

Thejesh Varma Vulchu
Rajiv Gandhi University of Knowledge Technologies, Nuzvid

Abstract

This project focuses on the design and implementation of a robust, speed-independent clock gating circuit using Muller C-elements to eliminate digital glitches. Standard clock gating often results in "runt pulses" when enable signals toggle asynchronously. By utilizing a dual-stage asynchronous architecture, this design ensures that transitions only occur during safe clock phases. The circuit was developed using KiCad and simulated via eSim's NgVeri and Ngspice engines. Results verify that the gated output provides full-width pulses, effectively maintaining signal integrity and reducing dynamic power consumption.

1 Introduction

Clock gating is a primary method for power optimization in modern System-on-Chip (SoC) designs. However, the integration of asynchronous control signals with high-frequency clocks often leads to metastability and timing glitches. Traditional AND-gate gating is insufficient for high-reliability applications where runt pulses can trigger false logic transitions. This project implements the design specified in patent EP2515197A1, which replaces standard latches with Muller C-elements to provide a self-timed synchronization window.

2 Circuit Description and Working Principle

The proposed circuit consists of two main stages of synchronization:

- **Stage 1 (Synchronization):** A Muller C-element receives the Enable signal and an inverted version of the Clock. This prevents the output from toggling high if the Enable arrives while the Clock is already high.
- **Stage 2 (Alignment):** A second Muller gate re-synchronizes the first stage's output with the delayed main Clock signal.
- **Delay Elements:** Digital buffers are placed in the clock path to compensate for the propagation delay of the Muller gates, ensuring perfectly aligned delivery to the final AND gate.

2.1 Circuit Diagram

Figure 1 shows the schematic of the proposed glitch-free clock gating circuit.

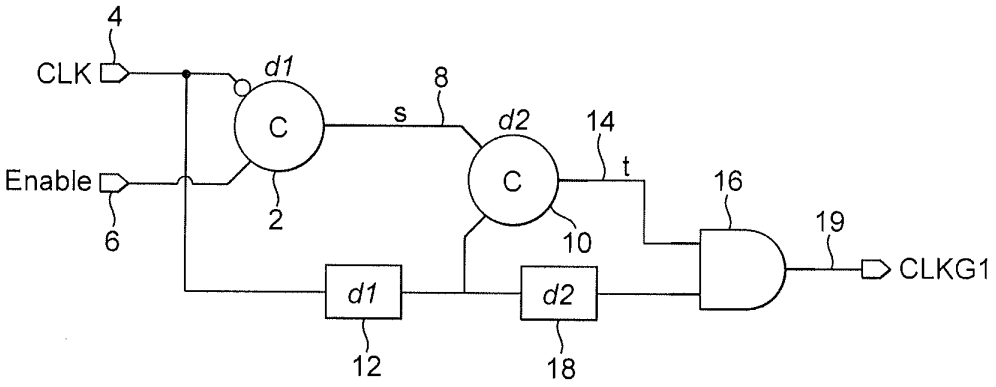

Figure 1: Circuit diagram of the dual Muller C-element based clock gating circuit.

2.2 Circuit Schematic in eSim

Figure 2 shows the schematic implementation of the proposed circuit in eSim.

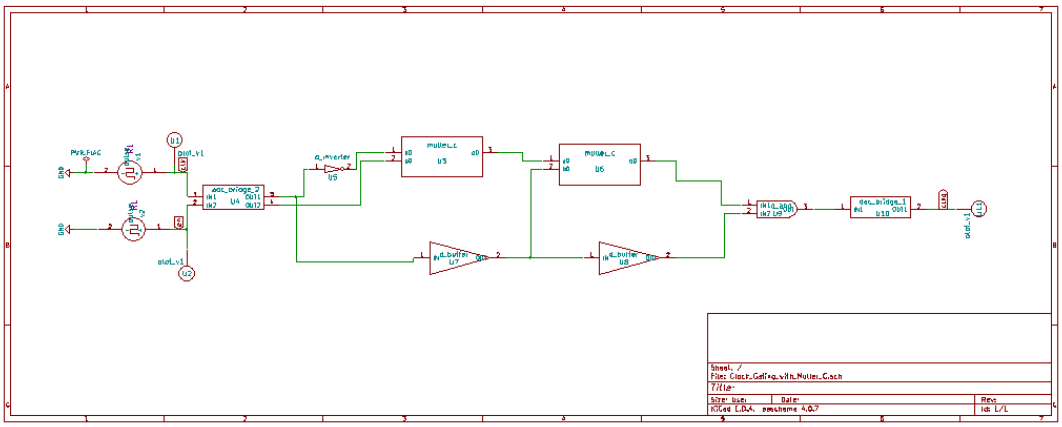

Figure 2: Circuit schematic implemented in eSim (KiCad + Ngspice/NgVeri flow).

2.3 Digital Implementation of Muller C-element in eSim

The digital implementation of a Muller C-element in eSim is achieved by modeling its asynchronous behavior in Verilog and integrating it via the NgVeri interface. The Verilog code defines the "rendezvous" logic where the output only transitions when both inputs reach a common state, effectively functioning as a state-holding latch. In the schematic, these digital blocks are connected to analog signals through ADC and DAC bridges, which manage the signal level switching required for mixed-signal simulation. This approach ensures that the gate maintains high timing accuracy and speed-independence, which are critical for glitch-free clock gating.

```
‘timescale 1ns/1ps

module muller_c (
    input wire a,
    input wire b,
    output reg q
);

    always @(*) begin
        if (a == 1’b1 && b == 1’b1)
            q <= 1’b1;
        else if (a == 1’b0 && b == 1’b0)
            q <= 1’b0;
    end
endmodule
```

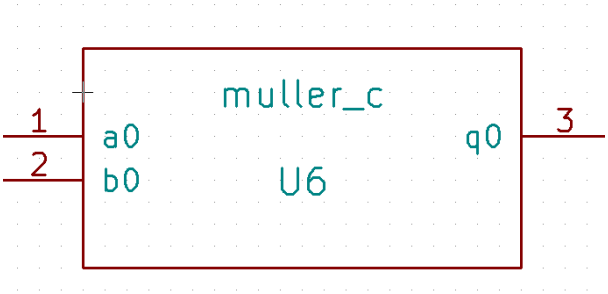

Figure 3: Verilog-based Muller C-element integrated in eSim using NgVeri with ADC/DAC bridges.

3 Methodology

The implementation followed the eSim mixed-signal simulation flow:

1. **Digital Modeling:** The Muller C-element was modeled in Verilog and integrated via NgVeri.
2. **Schematic Capture:** Components were placed in KiCad (Eeschema) using eSim component libraries.
3. **Bridging:** ADC and DAC bridges were utilized to interface the digital Muller blocks with analog pulse sources.
4. **Analysis:** Transient analysis was performed in Ngspice to verify the output across multiple clock periods.

4 Results and Discussion

The simulation output demonstrates that the gated clock signal ($v(clkg)$) transitions only at valid clock boundaries. Even when the Enable signal transitions during a high clock phase, the Muller-based synchronization waits for the subsequent rising edge to trigger the output.

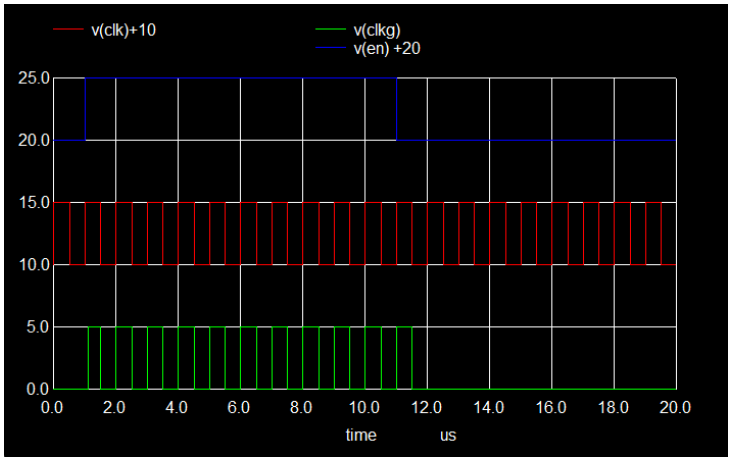

Figure 4: Case1: Enable triggered at clock transition

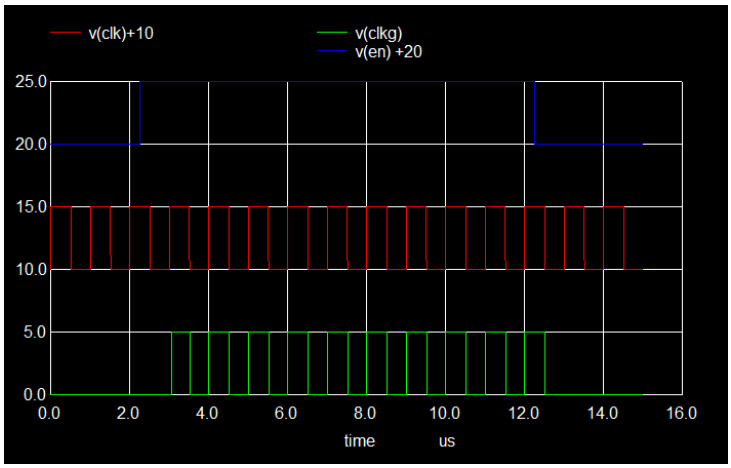

Figure 5: Case2: Enable triggered at after clock transition

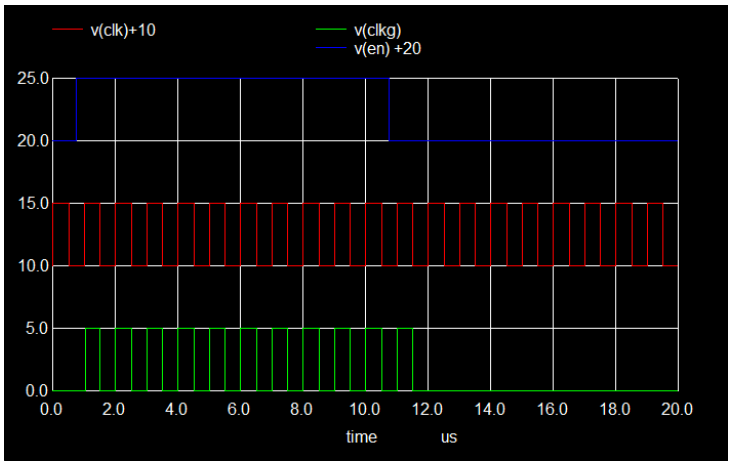

Figure 6: Case3: Enable triggered before clock transition

5 Conclusion

The design successfully suppresses all potential glitches at the output of the clock gating cell. The asynchronous nature of the Muller C-element makes the circuit robust against timing

skews, making it an ideal candidate for low-power, high-reliability digital systems.

References

1. Clock gating circuit using a Muller C-element, Patent EP2515197A1. patents.google.com
2. A Design of a Fast and Area Efficient Multi-Input Muller C-element. ResearchGate