

## RTL\_EXERCISE\_1 BOUND FLASHER

---

Author	Nhóm 7
Date	2024/03/07
Version	1.1

Họ và tên	MSSV
Nguyễn Trần Quang Vũ	2115325
Lê Minh Chiến	2112933
Trần Đình Phong	2114404
Hoàng Nhật Quân	2014262

## Contents

1. Interface .....	2
2. Functional implementation. ....	3
3. Internal implementation. ....	5
3.1. Overall. ....	5
3.2. State Machine.....	7
4. History .....	8

## 1. Interface

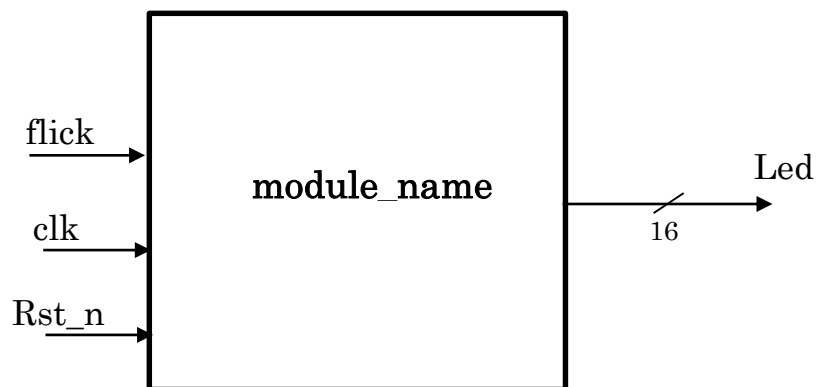


Figure 1: the figure of Bound Flasher System

Signal	Width	In/Out	Description
flick	1	In	Flick signal
clk	1	In	Clock signal
Rst_n	1	In	Reset signal
Led	16	Out	16 LEDs

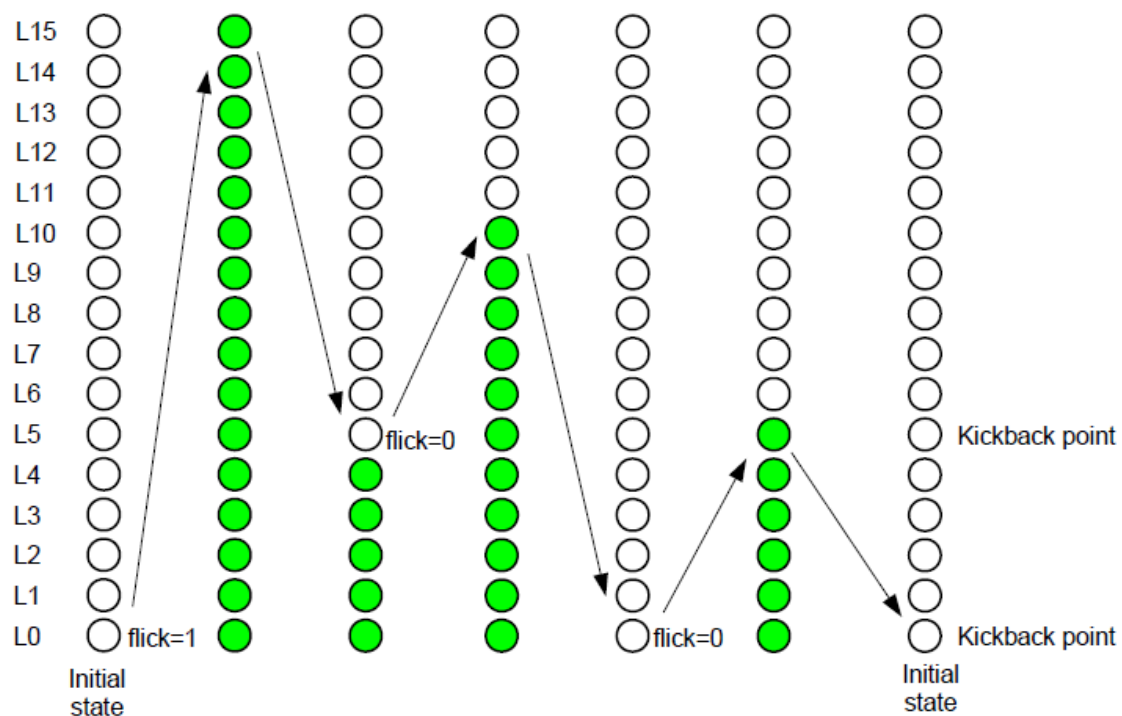
Table 1: Description of signals in Bound Flasher

### 2. Functional implementation.

- Implement a 16-bits LEDs system
- System's Operation base on three input signal
  - Reset
  - Clock
  - Flick
- The system specification
  - Clock signal is provided for system inspire of function status. The function operate state's transition at positive edge of the clock signal.
  - Reset signal:
    - LOW-ACTIVE Reset = 0: System is restarted to Initial State.
    - HIGH-ACTIVE Reset = 1: System is started with initial state.
- Flick signal: special input for controlling state transfer.
- At the initial state, all lamps are OFF. If flick signal is ACTIVE, the flasher start operating:
  - The lamps are turned ON gradually from LEDs [0] to LEDs [15].
  - The LEDSs are turned OFF gradually from LEDs [15] to LEDs [5].
  - The LEDSs are turned ON gradually from LEDs [5] to LEDs [10].
  - The LEDSs are turned OFF gradually from LEDs [10] to LEDs [0].
  - The LEDSs are turned ON gradually from LEDs [0] to LEDs [5].
  - Finally, the LEDs s are turned OFF gradually from LEDSS [5] to LEDSS [0], return to initial state.
- Additional condition: At each kickback point (LEDs [5] and LEDs [0]), if flick signal is ACTIVE, the LEDs will go back and repeat that STATE. For simple, kickback point is considered only when the LEDs s are turned OFF gradually, except final state.

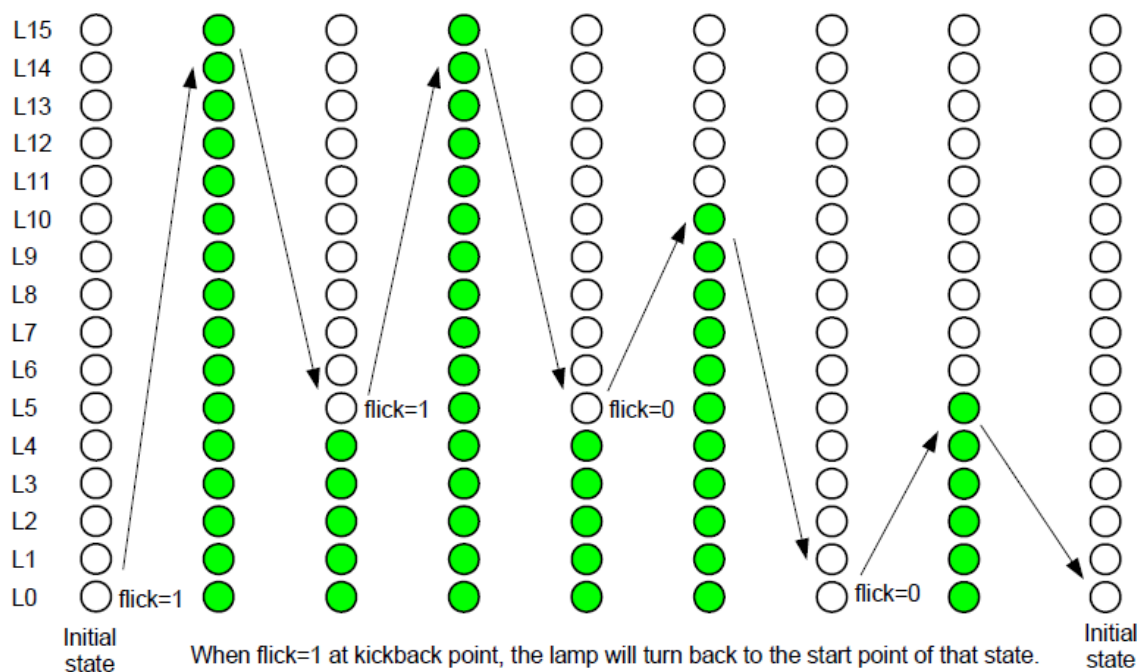
## RTL\_Exercise1 Bound Flasher

- Some insulations:
  - When flick = 0 at kickback points



- When flick = 1 at kickback points (lamp[5])

When flick=1 at kickback points (lamp[5])



## 3. Internal implementation.

### 3.1. Overall.

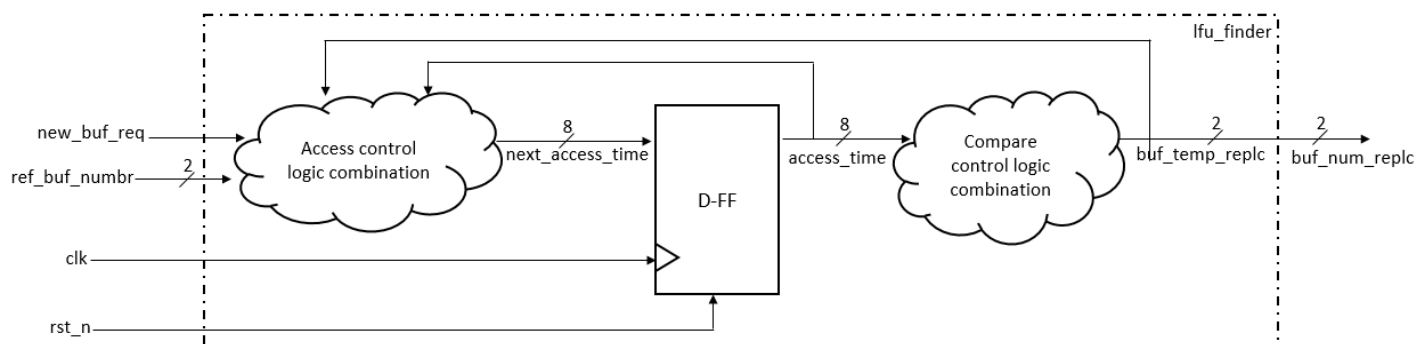
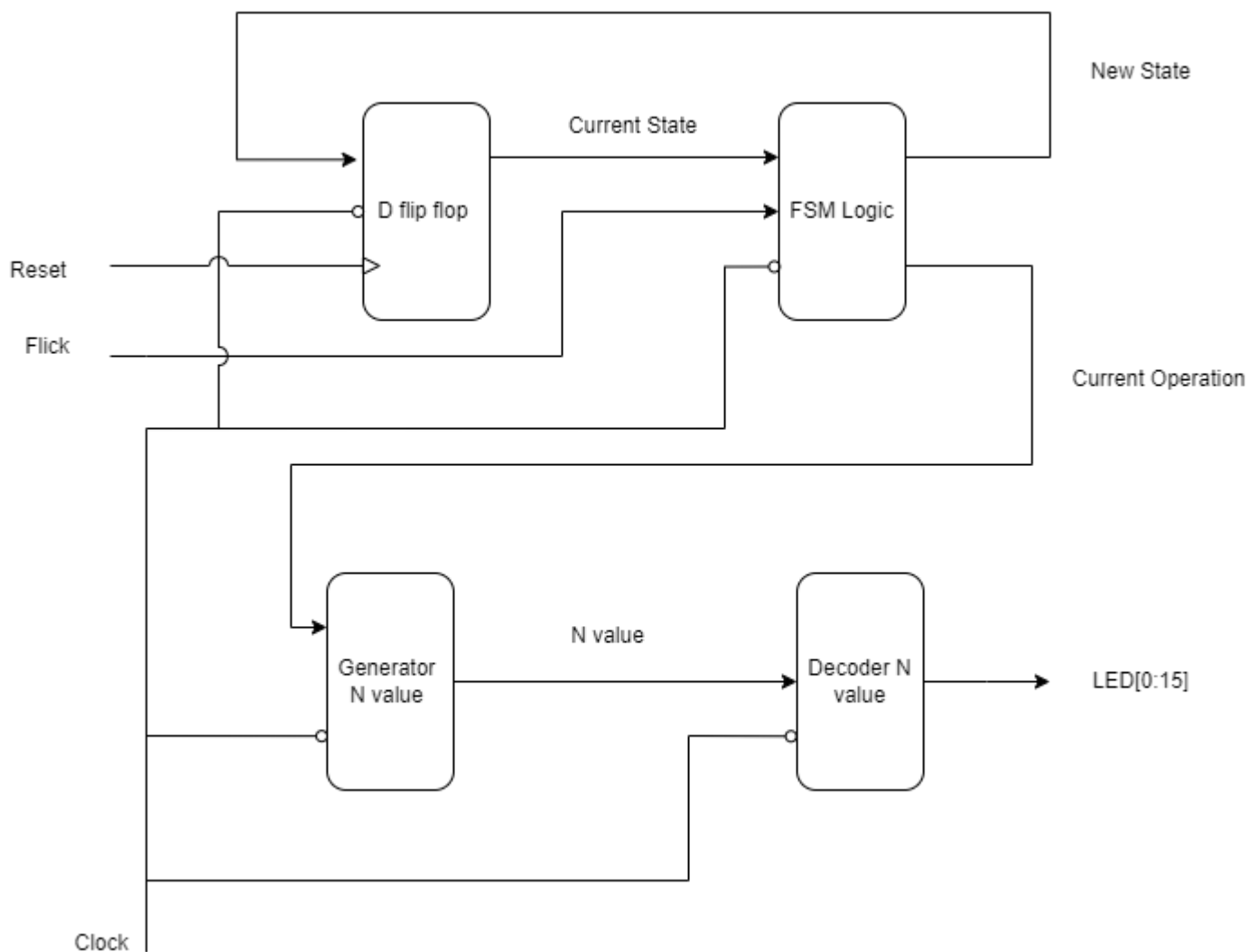


Figure 3.1 : Block diagram of Bound Flasher



## RTL\_Exercise1 Bound Flasher

Signal	Width	Type	Description
<i>Clock</i>	1 bit	Input	Đồng bộ hóa hoạt động của các khối chức năng
<i>Flick</i>	1 bit	Input	Nếu Flick có giá trị ở mức cao khi các đèn tắt lần lượt thì tiến hành lặp lại trạng thái này, nếu Flick active ở trạng thái cuối thì ta quay lại trạng thái ban đầu
<i>New State</i>	3 bit	Logic	Thể hiện cho trạng thái kế tiếp của hệ thống
<i>Current State</i>	3 bit	Logic	Thể hiện cho trạng thái hiện tại của hệ thống
<i>Current Operation</i>	2 bit	Logic	Thể hiện cho hành vi hiện tại của hệ thống dựa theo Current State và là nhân tố để quyết định giá trị của N
<i>N value</i>	32 bit (Thật sự chỉ cần 4 bit)	Integer	Giá trị được sử dụng để điều khiển trạng thái bật tắt của 16 bóng đèn, giá trị này sẽ được giải mã thông qua Decoder
<i>Led</i>	16 bit	Ouput	Thể hiện tín hiệu điện cho 16 chiếc đèn led sau khi được giải mã

Block	Description
<i>D Flip flop</i>	Tín hiệu đầu vào bao gồm : New State, Reset và Clock D flip flop được dùng để chuyển trạng thái kế tiếp của module (New State) thành trạng thái hiện tại (Current State) khi có cạnh lên của Clock, nếu có tín hiệu Reset thì trạng thái được đưa về trạng thái Init
<i>FSM Logic</i>	Tín hiệu đầu vào : Current State, Flick và Clock Mô hình máy trạng thái là module nhận Current State và Flick để từ đó có thể xác định được trạng thái tiếp theo của hệ thống, đồng thời tiến hành thực hiện hành vi của trạng thái hiện tại, việc này chỉ thực hiện khi có cạnh lên của Clock

<i>Generator N value</i>	<p>Tín hiệu đầu vào : Current Operation</p> <p>Dựa vào hành vi hiện tại (Current Operation) được đưa ra từ FSM Logic, block này tạo ra giá trị N dùng để điều khiển 16 đèn led, giá trị N này có thể lớn hơn, nhỏ hơn hoặc giống như giá trị N được sinh ra ở hành vi của trạng thái cũ.</p>
<i>Decoder N value</i>	<p>Tín hiệu đầu vào : N value</p> <p>Giải mã giá trị của N để tạo ra tín hiệu điện điều khiển 16 đèn led, hình thành trạng thái như chúng ta mong muốn</p>

Table 3.1: Block diagram of Bound Flasher Description

## 3.2. State Machine

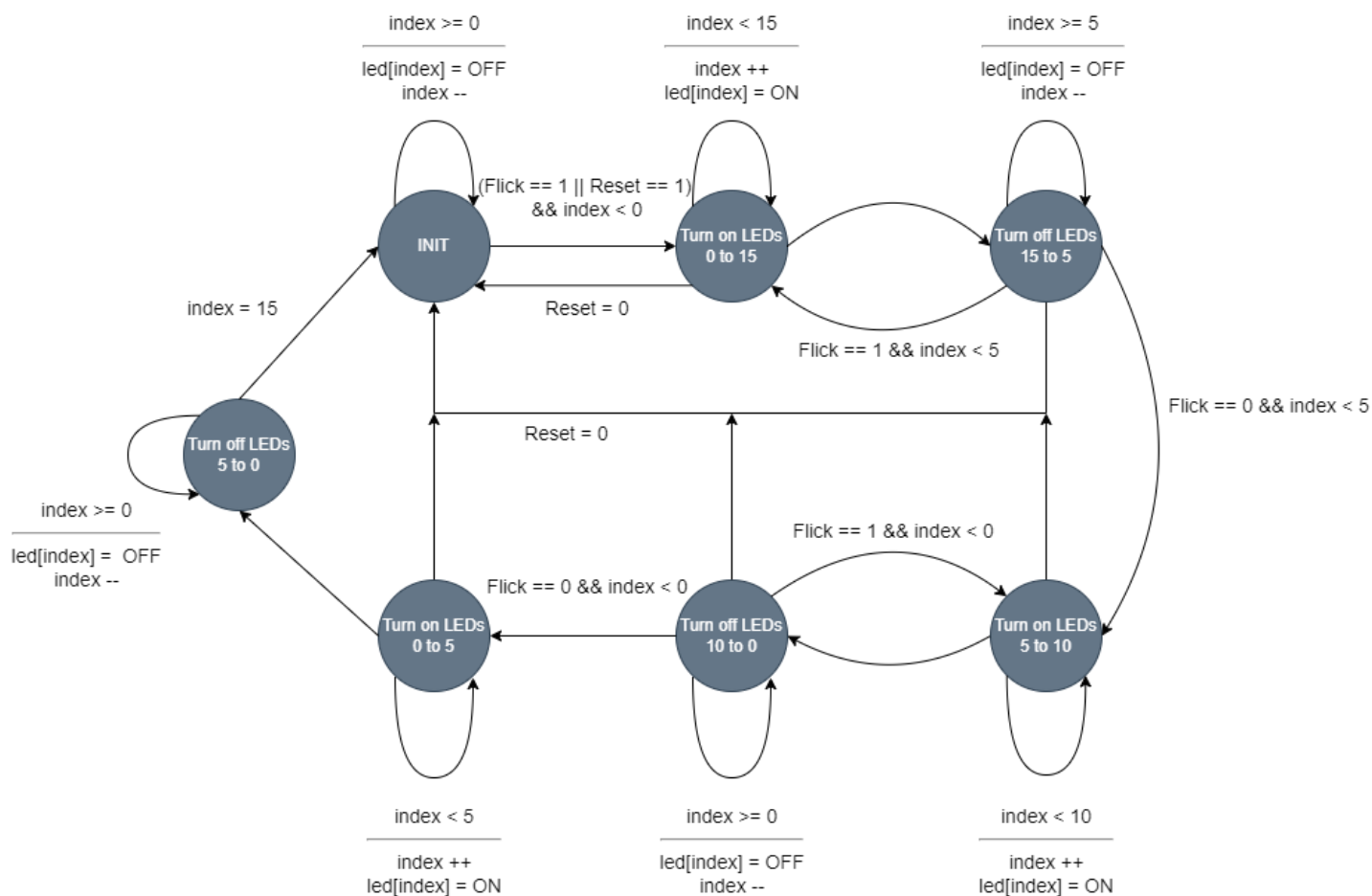




Figure 3.2: State Machine of Bound Flasher

Variable name	Description
<i>LED</i>	16 bits tín hiệu đầu ra dùng để điều khiển sáng tắt của bóng đèn. Có 2 giá trị là ON và OFF có thể hiểu là tín hiệu 1 và 0 cho biết bóng đèn sáng hay tắt.
<i>index</i>	Thể hiện cho chỉ số của bóng đèn, ví dụ với $index = 0$ thì $LED[index] = ON$ có nghĩa là bóng đèn 0 sẽ được bật sáng và ngược lại nếu là OFF thì bóng đèn 0 sẽ bị tắt.
<i>FLick</i>	Là tín hiệu góp phần quyết định trạng thái tiếp theo của hệ thống, và chỉ xem xét tại điểm kickback (kickback point là các trạng thái đếm xuống bao gồm 15 to 5 và 10 to 0 ).
<i>Reset</i>	Là tín hiệu quyết định hệ thống sẽ được khởi chạy hay sẽ được đưa về trạng thái ban đầu, $reset = 0$ sẽ đưa toàn bộ hệ thống về trạng thái ban đầu trong khi đó $reset = 1$ sẽ khởi động hệ thống.

Table 3.2: variable name of State machine

State name	Description
<i>INIT</i>	Là trạng thái khởi tạo, khi này tất cả các bóng đèn đều tắt
<i>Turn on LEDs 0 to 15</i>	Là trạng thái bật lần lượt tất cả 16 bóng đèn từ đèn 0 đến đèn 15
<i>Turn off LEDs 15 to 5</i>	Là trạng thái tắt lần lượt các bóng đèn từ đèn 15 xuống đèn 5
<i>Turn on LEDs 5 to 10</i>	Là trạng thái bật lần lượt các bóng đèn từ đèn 5 đến đèn 10
<i>Turn off LEDs 10 to 0</i>	Là trạng thái tắt lần lượt các bóng đèn từ đèn 10 xuống đèn 0
<i>Turn on LEDs 0 to 5</i>	Là trạng thái bật lần lượt các bóng đèn từ đèn 0 đến đèn 5
<i>Turn off LEDs 5 to 0</i>	Là trạng thái tắt lần lượt các bóng đèn từ đèn 5 xuống đèn 0

Table 3.3: state name of State machine

#### 4. History

Date	Author	Modified part	Description
2024/03/07	Nguyễn Trần Quang Vũ - 2115325	Internal Implement	Vẽ sơ đồ thiết kế Bound Flasher
2024/03/07	Lê Minh Chiến - 2112933	Internal Implement	Tạo và điền vào bảng dữ liệu mô tả sơ đồ thiết kế Bound Flasher
2024/03/07	Trần Đình Phong - 2114404	Internal Implement	Vẽ sơ đồ máy trạng thái mô tả hoạt động của Bound Flasher
2024/03/07	Hoàng Nhật Quân - 2014262	Internal Implement & Interface	Tạo và điền bảng dữ liệu mô tả tên biến, các trạng thái của Bound Flasher, đồng thời hoàn thiện bảng Function ở phần Interface