|  |
| --- |
|  |
| **RTL\_EXERCISE\_1 BOUND FLASHER** |
|  |
| |  |  | | --- | --- | | Author | Trần Anh Kiệt | | Date | 5/3/2024 | | Version | 1.1 | |
|  |
|  |

|  |
| --- |
|  |

Contents

[1. Interface 2](#_Toc478478267)

[2. Functional implementation. 3](#_Toc478478268)

[3. Internal implementation. 5](#_Toc478478269)

[3.1. Overall. 5](#_Toc478478270)

[3.2. State Machine 6](#_Toc478478271)

[4. History 7](#_Toc478478272)

# 1. Interface

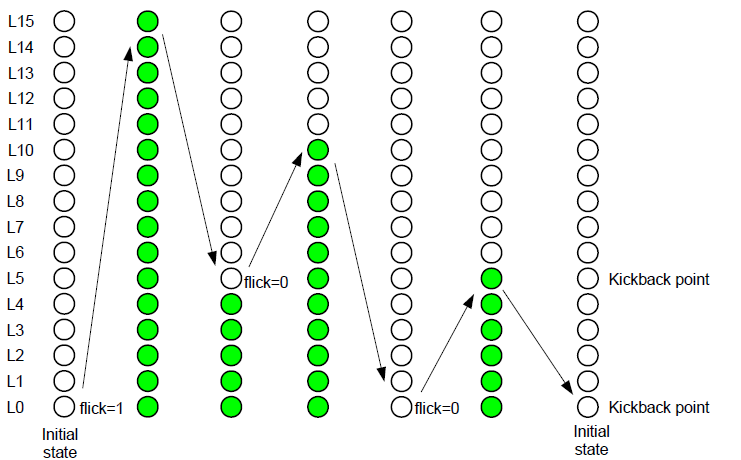
|  |
| --- |
| **Bounce\_flash**  clock  16  lamps  flick  reset |
| Figure 1: the figure of Bound Flasher System |

|  |  |  |  |
| --- | --- | --- | --- |
| Signal | Width | In/Out | Description |
| Clock | 1 | In | The clock signal that drives the state changes and timing within the module. Typically connected to the system clock. |
| Reset | 1 | In | Active-low reset signal. When asserted (0), it initializes the module's state to INIT and resets the lamps and lamp\_counter. |
| Flick | 1 | In | Input signal used to trigger specific lamp behavior. When asserted (1), it activates flicker logic affecting the current lamp pattern. |
| lamps | 16 | Out | Output signal representing the state of 16 lamps. Each bit corresponds to a lamp, where 1 indicates on and 0 indicates off. |

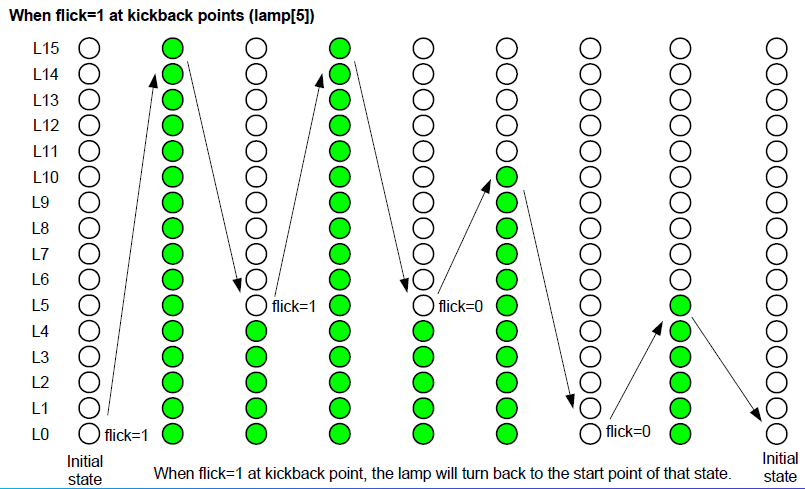
Table 1: Description of signals in Bound Flasher

# 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.
* Some insulations:
* When flick = 0 at kickback points



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



# 3. Internal implementation.

## 3.1. Overall.

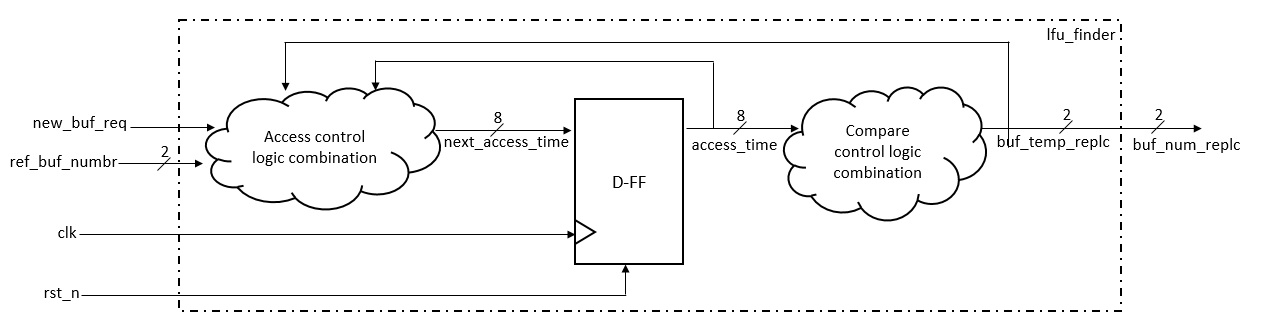


Figure 3.1: Block diagram of Bound Flasher

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | **Signal** | **Width** | **In/Out** | **Description** | | --- | --- | --- | --- | | new\_buf\_req | 1 | In | Signal to request a new buffer. | | ref\_buf\_numbr | 2 | In | Reference buffer number input signal. | | clk | 1 | In | The clock signal that drives the state changes and timing within the module. | | rst\_n | 1 | In | Active-low reset signal. When asserted (0), it initializes the module’s state. | | next\_access\_time | 8 | Out | Next access time output from access control logic combination. | | access\_time | 8 | Out | Access time output from D-FF component after processing next\_access\_time. | | buf\_temp\_replc | 2 | Out | Buffer template replacement output after comparison in control logic combination. | | buf\_num\_replc | 2 | Out | Buffer number replacement output after comparison in control logic combination. | |

Table 3.1: Block diagram of Bound Flasher Description

## 3.2. State Machine

A diagram of a flowchart

Description automatically generated

Figure 3.2: State Machine of Bound Flasher

| Variable Name | Description |
| --- | --- |
| current\_state | Holds the current state of the state machine. |
| next\_state | Intended to hold the next state (not used in the provided code). |
| flick\_active | A flag to indicate if the flick signal has been activated. |
| lamp\_counter | A counter to keep track of the current lamp pattern. |

Table 3.2: variable name of State machine

| **State Name** | **Description** |
| --- | --- |
| **INIT** | Initial state where all lamps are turned off. The system transitions to **TURN\_ON\_TO\_5** next. |
| **TURN\_ON\_TO\_5** | Lamps are sequentially turned on from 0 to 5. After reaching 5, transitions to **TURN\_OFF\_TO\_0**. |
| **TURN\_OFF\_TO\_0** | Lamps are sequentially turned off from 5 to 0. After all are off, transitions to **TURN\_ON\_TO\_10**. |
| **TURN\_ON\_TO\_10** | Lamps are sequentially turned on from 0 to 10. If **flick** is activated at lamp 5, it flicks before continuing. After reaching 10, transitions to **TURN\_OFF\_TO\_5**. |
| **TURN\_OFF\_TO\_5** | Lamps are sequentially turned off from 10 to 5. After reaching 5, transitions to **TURN\_ON\_TO\_15**. |
| **TURN\_ON\_TO\_15** | Lamps are sequentially turned on from 5 to 15. If **flick** is activated at lamp 10, it flicks before continuing. After reaching 15, transitions to **TURN\_OFF\_TO\_0\_AGAIN**. |
| **TURN\_OFF\_TO\_0\_AGAIN** | Lamps are sequentially turned off from 15 to 0. After all are off, transitions to **BLINK**. |
| **BLINK** | All lamps are turned on, then all are turned off after a brief moment. The system then resets to **INIT**. |

Table 3.3: state name of State machine

# 4. History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Modified part | Description |
| 2017/03/28 |  | All | New creation |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |