BOUND FLASHER VERIFICATION

Author	Trần Anh Tài, Nguyễn Hữu Thông, Võ Thị Hoàng Yến
Date	2024/03/26
Version	1.1

Bound Flasher Verification

Contents

1. Combination block verification:

1.1. Next state generator verification:
(Source: next_state_generator_tb.v)

1.1.1. Description:
The testbench **next_state_generator_tb.v** is designed to verify the functionality of the **next_state_generator** module. This module is responsible for generating the next state and controlling aspects of a counter based on various input signals. The testbench provides different input combinations and checks if the outputs of the module match the expected behavior under those conditions.

1.1.2. Expected result:
The module to correctly compute the next main state, set appropriate values for counter loading and enable signals, and determine the count state based on the current main state and input conditions. Specific expectations may include:

- Correct transition to ONLED0_15_STATE when flick is asserted in INIT_STATE.
- Transition to OFFLED15_5_STATE when the counter reaches 16 in ONLED0_15_STATE.
- Immediate counter load to 16 and enable counter loading when kickback_match is asserted in OFFLED15_5_STATE.
- And so on for other state transitions and control signals.

1.1.3. Result:

```
----------------------------------------------------------------------
Test case 0:  Initializate with flick = 0 => Output: FSM dont active
----------------------------------------------------------------------
Main State: 000 -> Next Main State: xxx
Counter Load: xxxxx
Counter Load Enable: x
Count State: xx
----------------------------------------------------------------------------------
Test case 1: Flick occurs in INIT_STATE => Output: Count_state = 01 (Count up )
----------------------------------------------------------------------------------
Main State: 000 -> Next Main State: 001
Counter Load: 00000
Counter Load Enable: 0
Count State: 01
-------------------------------------------------------------------------------------------
Test case 2: Counter reaches 16 in ONLED0_15_STATE => Output: Count_state = 01 (Count up )
-------------------------------------------------------------------------------------------
Main State: 001 -> Next Main State: 010
Counter Load: 10000
Counter Load Enable: 0, Count State: 10
--------------------------------------------------
Test case 3: Kickback occurs in OFFLED15_5_STATE
--------------------------------------------------
Main State: 010 -> Next Main State: 010
Counter Load: 10000
Counter Load Enable: 1
Count State: 10
--------------------------------------------------
Test case 4: Counter reaches 0 in OFFLED5_0_STATE
--------------------------------------------------
Main State: 110 -> Next Main State: 000
Counter Load: 00000
Counter Load Enable: 1
Count State: 00
main.v:203: $finish called at 90 (1s)
```

Output of next_state_generator_tb.v

1.1.4. Explanation:

- In test case 0, flick = 0 so FSM dont active. FSM active with flick = 1.
- Test case 1 and 2 represent the state changes of the FSM.
- Test case 3 represents the kickback point with flick = 1. The FSM returns the count state at 10, indicating count up.
- Test case 4 represents the off LED state, decrementing the counter with count state as 00.

1.2. Next counter generator verification:

(Source: next_counter_generator_tb.v)

1.2.1. Description:

The testbench **next_counter_generator_tb.v** is designed to verify the functionality of the **next_counter_generator** module. This module is responsible for generating the next state of a counter based on input signals such as the current counter value, a load value, an enable signal for loading the counter, and a count state indicating whether to disable counting, count up, or count down.

1.2.2. Expected result:

When running the testbench, we expect to observe correct behavior from the next_counter_generator module. Specifically, we expect the output counter_n to reflect the appropriate next state based on the input signals provided during simulation.

1.2.3. Result:

```
---------------State default--------------
Input: counter=00001, counter_load=xxxxx, counter_load_en=x, count_state=xx, Output: counter_n=00001
------------State COUNT_INIT------------
Input: counter=00001, counter_load=xxxxx, counter_load_en=0, count_state=00, Output: counter_n=00000
----------------State COUNT_UP_EN----------
Initial counter: 00000
Input: counter=00000, counter_load=xxxxx, counter_load_en=0, count_state=01, Output: counter_n=00001
Counter after count up: 00001
--------------State COUNT_DOWN_EN--------------
Initial counter: 11111
Input: counter=11111, counter_load=xxxxx, counter_load_en=0, count_state=10, Output: counter_n=11110
Counter after count down: 11110
----------If Enable counter load---------
Initial counter load: 10101
Counter after load: 11110
Input: counter=11111, counter_load=10101, counter_load_en=1, count_state=10, Output: counter_n=10101
```

Output of next_counter_generator_tb.v

1.2.4. Explanation:

In the Count_up state, the initial counter is 00000, and then it transitions to 00001.
In the Count_down state, the initial counter is 11111, and then it transitions to 11110.

1.3. Kickback match generator verification:
(Source: kickback_match_generator_tb.v)

1.3.1. Description:
The testbench kickback_match_generator_tb.v is designed to verify the functionality of the kickback_match_generator module. This module is responsible for generating the kickback_match signal based on the input signals flick and counter. Specifically, the kickback_match signal is asserted when flick is active and the counter value is either 0 or 5.

1.3.2. Expected result:
The expected result is that the kickback_match signal is asserted only when flick is active and the counter value is either 0 or 5. In other words:
- If flick is active and counter is 0 or 5, then kickback_match should be asserted.
- If flick is inactive or counter is not 0 or 5, then kickback_match should be deasserted.

1.3.3. Result:
```
time =                  20, flick = 0, counter =  0, kickback_match = 0
time =                  40, flick = 0, counter =  5, kickback_match = 0
time =                  60, flick = 0, counter =  0, kickback_match = 0
time =                  80, flick = 1, counter =  0, kickback_match = 1
time =                 100, flick = 1, counter =  5, kickback_match = 1
```
Output of kickback_match_generator_tb.v

1.3.4. Explanation:
At time = 20 and 40: when flick = 0, counter = 0 and 5, kickback_match is not activated.

At time = 80 and 100: when flick = 1, counter = 0 and 5, kickback_match is activated.

1.4. Decoder verification:
(Source: decoder_tb.v)

1.4.1. Description:
The decoder_tb module is a testbench for the decoder_under module used to verify input values and decode those values into corresponding sets of 1-bit.

1.4.2. Expected result:
For an input number n, the decoder sets n bits to 1. If the input number is greater than 31, it wraps around and starts setting bits from 0 again.

1.4.3. Result:

```
Input: 11, Output: 0000000000000000000011111111111
Input: 31, Output: 1111111111111111111111111111111
Input: 35, Output: 0000000000000000000000000000111
```

Output of decoder_tb.v

1.4.4. Explanation:

- For an input of 11, the result is 11 bits set to 1.
- For an input of 31, 31 bits are set to 1.
- For an input of 35, 3 bits are set to 1. (Overflow case)

2. Sequential block verification:

2.1. Bound Flasher FSM verification:

(Source: bound_flasher_fsm_tb.v)

2.1.1. Description:

The testbench **bound_flahser_fsm_tb** module is used to verify a combination block. In this testbench, we just verify load data on the rising edge of the clock and load data on falling edge of the negative reset signal.

2.1.2. Expected result:

When clk posedge, and rst_n negedge , then main_state = main_state_n.
When clk posedge, and rst_n posedge, then main_state = INIT_STATE = 3’b000.

2.1.3. Result:

```
Initializate clk = 0, rst_n = 1, main_state_n = 001, main_state = xxx
clk = 1, rst_n = 0, main_state_n = 001, main_state = 001
clk = 0, rst_n = 0, main_state_n = 001, main_state = 000
clk = 1, rst_n = 0, main_state_n = 001, main_state = 000
clk = 0, rst_n = 1, main_state_n = 001, main_state = 000
clk = 1, rst_n = 1, main_state_n = 001, main_state = 001
clk = 0, rst_n = 1, main_state_n = 001, main_state = 001
clk = 1, rst_n = 1, main_state_n = 001, main_state = 001
clk = 0, rst_n = 1, main_state_n = 001, main_state = 001
clk = 1, rst_n = 0, main_state_n = 001, main_state = 001
clk = 0, rst_n = 0, main_state_n = 001, main_state = 000
clk = 1, rst_n = 1, main_state_n = 001, main_state = 000
clk = 0, rst_n = 0, main_state_n = 001, main_state = 000
clk = 1, rst_n = 0, main_state_n = 001, main_state = 000
clk = 0, rst_n = 0, main_state_n = 001, main_state = 000
main.v:107: $finish called at 1080 (1s)
```

Output of bound_flasher_fsm_tb.v

2.1.4. Explanation:

- When clk posedge, and rst_n negedge , then main_state = main_state_n = 3’b001.
- When clk posedge, and rst_n posedge, then main_state = INIT_STATE = 3’b000.
- When clk negedge, and rst_n posedge, then main_state doesn't change previous state.
- When clk negedge, and rst_n negedge, then main_state = INIT_STATE = 3’b000.

2.2. Counter verification:

(Source: counter_tb.v)

2.2.1. Description:

The testbench **counter_tb** module is used to verify a combination block. In this testbench, we just verify load data on the rising edge of the clock and load data on falling edge of the negative reset signal.

2.2.2. Expected result:

When clk posedge, and rst_n negedge , then counter = counter_n
When clk posedge, and rst_n posedge, then counter= INIT_STATE = 5’b00000.

2.2.3. Result:

```
Initialize clk = 0, rst_n = 1, counter_n = 10101, counter = xxxxx
 clk = 0, rst_n = 0, counter_n = 10101, counter = 00000
clk = 1, rst_n = 0, counter_n = 10101, counter = 00000
 clk = 0, rst_n = 0, counter_n = 10101, counter = 00000
 clk = 1, rst_n = 1, counter_n = 10101, counter = 00000
clk = 0, rst_n = 1, counter_n = 10101, counter = 00000
clk = 1, rst_n = 1, counter_n = 10101, counter = 10101
 clk = 0, rst_n = 1, counter_n = 10101, counter = 10101
 clk = 1, rst_n = 0, counter_n = 10101, counter = 10101
 clk = 0, rst_n = 0, counter_n = 10101, counter = 00000
clk = 1, rst_n = 0, counter_n = 10101, counter = 00000
main.v:80: $finish called at 1065 (1s)
```

Output of counter_tb.v

2.2.4. Explanation:

- When clk posedge, and rst_n negedge , then counter = counter_n = 5’b10101.
- When clk posedge, and rst_n posedge, then counter= INIT_STATE = 5’b00000.
- When clk negedge, and rst_n posedge, then counter doesn't change previous state.
- When clk negedge, and rst_n negedge, then counte_n = INIT_STATE = 5’b00000.

3. System verification:

3.1. System verification 0:

(Source: system_verification_0_tb.v)

3.1.1. Description:

The testbench **system_verification_0_tb** module is used to verify the reset signal in the system. The reset signal in the system is an asynchronous signal and active low, which will set all registers to initial value.

3.1.2. Expected result:

Whenever the reset signal is falling, all state registers and counter will be reset to their initial values, and the system will be "frozen" as long as the reset signal remains LOW level

3.1.3. Result:

```
--------------------------------------
Time slot:40    LED_STATE:     0000000000000000,Flick:x,Reset:0
Time slot:42    LED_STATE:     0000000000000000,Flick:x,Reset:0
Time slot:44    LED_STATE:     0000000000000000,Flick:x,Reset:0
Time slot:46    LED_STATE:     0000000000000000,Flick:x,Reset:1
Time slot:48    LED_STATE:     0000000000000000,Flick:x,Reset:1
Time slot:50    LED_STATE:     0000000000000000,Flick:x,Reset:1
Time slot:52    LED_STATE:     0000000000000000,Flick:1,Reset:1
Time slot:54    LED_STATE:     0000000000000001,Flick:1,Reset:1
Time slot:56    LED_STATE:     0000000000000011,Flick:1,Reset:1
Time slot:58    LED_STATE:     0000000000000111,Flick:0,Reset:1
Time slot:60    LED_STATE:     0000000000001111,Flick:0,Reset:1
Time slot:62    LED_STATE:     0000000000011111,Flick:0,Reset:1
Time slot:64    LED_STATE:     0000000000111111,Flick:0,Reset:1
Time slot:66    LED_STATE:     0000000001111111,Flick:0,Reset:1
Time slot:68    LED_STATE:     0000000011111111,Flick:0,Reset:1
Time slot:70    LED_STATE:     0000000111111111,Flick:0,Reset:1
Time slot:72    LED_STATE:     0000001111111111,Flick:0,Reset:1
Time slot:74    LED_STATE:     0000011111111111,Flick:0,Reset:1
Time slot:76    LED_STATE:     0000111111111111,Flick:0,Reset:1
Time slot:78    LED_STATE:     0001111111111111,Flick:0,Reset:0
Time slot:80    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:82    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:84    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:86    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:88    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:90    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:92    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:94    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:96    LED_STATE:     0000000000000000,Flick:0,Reset:0
Time slot:98    LED_STATE:     0000000000000000,Flick:0,Reset:1
Time slot:100   LED_STATE:     0000000000000000,Flick:0,Reset:1
Time slot:102   LED_STATE:     0000000000000000,Flick:1,Reset:1
Time slot:104   LED_STATE:     0000000000000001,Flick:1,Reset:1
Time slot:106   LED_STATE:     0000000000000011,Flick:1,Reset:1
Time slot:108   LED_STATE:     0000000000000111,Flick:1,Reset:1
Time slot:110   LED_STATE:     0000000000001111,Flick:1,Reset:1
Time slot:112   LED_STATE:     0000000000011111,Flick:1,Reset:1
Time slot:114   LED_STATE:     0000000000111111,Flick:1,Reset:1
Time slot:116   LED_STATE:     0000000001111111,Flick:1,Reset:1
Time slot:118   LED_STATE:     0000000011111111,Flick:1,Reset:1
Time slot:120   LED_STATE:     0000000111111111,Flick:1,Reset:1
Time slot:122   LED_STATE:     0000001111111111,Flick:1,Reset:0
Time slot:124   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:126   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:128   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:130   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:132   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:134   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:136   LED_STATE:     0000000000000000,Flick:1,Reset:0
Time slot:138   LED_STATE:     0000000000000000,Flick:1,Reset:0
```

Output of system_verification_0_tb.v

3.1.4. Explanation:
The reset signal is LOW at *time slot 78* and *time slot 122*, and after each of these states, the 16-bit LED state returns to its initial state and remains frozen until the reset is HIGH.

3.2. System verification 1:
(Source: system_verification_1_tb.v)

3.2.1. Description:
The testbench **system_verification_1_tb** module is used to verify the normal state machine in system. In this testbench, the *flick* signal is always equal to 0 except in INIT_STATE to verify normal state transition.

3.2.2. Expected result:
The state machine will operate as follows:

- INIT_STATE to ONLED0_15: “flick” signal is asserted
- ONLED0_15 to OFFLED15_5: The LEDs are turned on gradually from LED[0] to LED[15].
  (Duration: 16 cycles)
- OFFLED15_5 to ONLED5_10: The LEDs are turned off gradually from LED[15] to LED[5].
  (Duration: 11 cycles)
- ONLED5_10 to OFFLED10_0: The LEDs are turned on gradually from LED[5] to LED[10].
  (Duration: 6 cycles)
- OFFLED10_0 to ONLED0_5: The LEDs are turned off gradually from LED[10] to LED[0].
  (Duration: 11 cycles)
- ONLED0_5 to OFFLED5_0: The LEDs are turned on gradually from LED[0] to LED[5].
  (Duration: 6 cycles)
- OFFLED5_0 to INIT_STATE: The LEDs are turned off gradually from LED[5] to LED[0].
  (Duration: 6 cycles)

3.2.3. Result:

```
Time slot:42    LED_STATE:      0000000000000000,Flick:x
Time slot:44    LED_STATE:      0000000000000000,Flick:x
Time slot:46    LED_STATE:      0000000000000000,Flick:x
Time slot:48    LED_STATE:      0000000000000000,Flick:x
Time slot:50    LED_STATE:      0000000000000000,Flick:x
Time slot:52    LED_STATE:      0000000000000000,Flick:1
Time slot:54    LED_STATE:      0000000000000001,Flick:1
Time slot:56    LED_STATE:      0000000000000011,Flick:1
Time slot:58    LED_STATE:      0000000000000111,Flick:0
Time slot:60    LED_STATE:      0000000000001111,Flick:0
Time slot:62    LED_STATE:      0000000000011111,Flick:0
Time slot:64    LED_STATE:      0000000000111111,Flick:0
Time slot:66    LED_STATE:      0000000001111111,Flick:0
Time slot:68    LED_STATE:      0000000011111111,Flick:0
Time slot:70    LED_STATE:      0000000111111111,Flick:0
Time slot:72    LED_STATE:      0000001111111111,Flick:0
Time slot:74    LED_STATE:      0000011111111111,Flick:0
Time slot:76    LED_STATE:      0000111111111111,Flick:0
Time slot:78    LED_STATE:      0001111111111111,Flick:0
Time slot:80    LED_STATE:      0011111111111111,Flick:0
Time slot:82    LED_STATE:      0111111111111111,Flick:0
Time slot:84    LED_STATE:      1111111111111111,Flick:0
Time slot:86    LED_STATE:      0111111111111111,Flick:0
Time slot:88    LED_STATE:      0011111111111111,Flick:0
Time slot:90    LED_STATE:      0001111111111111,Flick:0
Time slot:92    LED_STATE:      0000111111111111,Flick:0
Time slot:94    LED_STATE:      0000011111111111,Flick:0
Time slot:96    LED_STATE:      0000001111111111,Flick:0
Time slot:98    LED_STATE:      0000000111111111,Flick:0
Time slot:100   LED_STATE:      0000000011111111,Flick:0
Time slot:102   LED_STATE:      0000000001111111,Flick:0
Time slot:104   LED_STATE:      0000000000111111,Flick:0
Time slot:106   LED_STATE:      0000000000011111,Flick:0
Time slot:108   LED_STATE:      0000000000111111,Flick:0
Time slot:110   LED_STATE:      0000000001111111,Flick:0
Time slot:112   LED_STATE:      0000000011111111,Flick:0
Time slot:114   LED_STATE:      0000000111111111,Flick:0
Time slot:116   LED_STATE:      0000001111111111,Flick:0
Time slot:118   LED_STATE:      0000011111111111,Flick:0
Time slot:120   LED_STATE:      0000001111111111,Flick:0
Time slot:122   LED_STATE:      0000000111111111,Flick:0
Time slot:124   LED_STATE:      0000000011111111,Flick:0
Time slot:126   LED_STATE:      0000000001111111,Flick:0
Time slot:128   LED_STATE:      0000000000111111,Flick:0
Time slot:130   LED_STATE:      0000000000011111,Flick:0
Time slot:132   LED_STATE:      0000000000001111,Flick:0
Time slot:134   LED_STATE:      0000000000000111,Flick:0
Time slot:136   LED_STATE:      0000000000000011,Flick:0
Time slot:138   LED_STATE:      0000000000000001,Flick:0
Time slot:140   LED_STATE:      0000000000000000,Flick:0
Time slot:142   LED_STATE:      0000000000000001,Flick:0
Time slot:144   LED_STATE:      0000000000000011,Flick:0
Time slot:146   LED_STATE:      0000000000000111,Flick:0
Time slot:148   LED_STATE:      0000000000001111,Flick:0
Time slot:150   LED_STATE:      0000000000011111,Flick:0
Time slot:152   LED_STATE:      0000000000111111,Flick:0
Time slot:154   LED_STATE:      0000000000011111,Flick:0
Time slot:156   LED_STATE:      0000000000001111,Flick:0
Time slot:158   LED_STATE:      0000000000000111,Flick:0
Time slot:160   LED_STATE:      0000000000000011,Flick:0
Time slot:162   LED_STATE:      0000000000000001,Flick:0
Time slot:164   LED_STATE:      0000000000000000,Flick:0
Time slot:166   LED_STATE:      0000000000000000,Flick:0
Time slot:168   LED_STATE:      0000000000000000,Flick:0
Time slot:170   LED_STATE:      0000000000000000,Flick:0
```

Output of system_verification_1_tb.v

3.2.4. Explanation:

Description: Check the state transition timings

- INIT_STATE to ONLED0_15 at Time slot 52 (when flick signal is asserted)
- ONLED0_15 t0 OFFLED15_5 at Time slot 84 (after 16 cycles)
- OFFLED15_5 to ONLED5_10 at Time slot 106 (after 11 cycles)
- ONLED5_10 to OFFLED10_0 at Time slot 118 (after 6 cycles)
- OFFLED10_0 to ONLED0_5 at Time slot 140 (after 11 cycles)
- ONLED0_5 to OFFLED5_0 at Time slot 152 (after 6 cycles)
- OFFLED5_0 tp INIT_STATE at Time slot 164 (after 6 cycles)
- Completed at time slot 164

3.3. System verification 2:

(Source: system_verification_2_tb.v)

3.2.1. Description:

The testbench **system_verification_2_tb** module is used to verify behavior of kickback cases. In this testbench, we only verify the case when the flick signal is equal to 1 in the OFFLED15_5 state.

3.2.2. Expected result:

At LED[5], if the flick signal is 1, the system state will revert to the first behavior of the current state, contrary to LED[5] to LED[15] immediately turning on and gradually turning off afterwards.

3.2.3. Result:

```
--------------------------------------
Time slot:40    LED_STATE:     0000000000000000,Flick:x
Time slot:42    LED_STATE:     0000000000000000,Flick:x
Time slot:44    LED_STATE:     0000000000000000,Flick:x
Time slot:46    LED_STATE:     0000000000000000,Flick:x
Time slot:48    LED_STATE:     0000000000000000,Flick:x
Time slot:50    LED_STATE:     0000000000000000,Flick:x
Time slot:52    LED_STATE:     0000000000000000,Flick:1
Time slot:54    LED_STATE:     0000000000000001,Flick:1
Time slot:56    LED_STATE:     0000000000000011,Flick:1
Time slot:58    LED_STATE:     0000000000000111,Flick:0
Time slot:60    LED_STATE:     0000000000001111,Flick:0
Time slot:62    LED_STATE:     0000000000011111,Flick:0
Time slot:64    LED_STATE:     0000000000111111,Flick:1
Time slot:66    LED_STATE:     0000000001111111,Flick:1
Time slot:68    LED_STATE:     0000000011111111,Flick:1
Time slot:70    LED_STATE:     0000000111111111,Flick:1
Time slot:72    LED_STATE:     0000001111111111,Flick:1
Time slot:74    LED_STATE:     0000011111111111,Flick:1
Time slot:76    LED_STATE:     0000111111111111,Flick:1
Time slot:78    LED_STATE:     0001111111111111,Flick:1
Time slot:80    LED_STATE:     0011111111111111,Flick:1
Time slot:82    LED_STATE:     0111111111111111,Flick:1
Time slot:84    LED_STATE:     1111111111111111,Flick:1
Time slot:86    LED_STATE:     0111111111111111,Flick:1
Time slot:88    LED_STATE:     0011111111111111,Flick:1
Time slot:90    LED_STATE:     0001111111111111,Flick:1
Time slot:92    LED_STATE:     0000111111111111,Flick:1
Time slot:94    LED_STATE:     0000011111111111,Flick:1
Time slot:96    LED_STATE:     0000001111111111,Flick:1
Time slot:98    LED_STATE:     0000000111111111,Flick:1
Time slot:100   LED_STATE:     0000000011111111,Flick:1
Time slot:102   LED_STATE:     0000000001111111,Flick:1
Time slot:104   LED_STATE:     0000000000111111,Flick:1
Time slot:106   LED_STATE:     0000000000011111,Flick:1
Time slot:108   LED_STATE:     1111111111111111,Flick:1
Time slot:110   LED_STATE:     0111111111111111,Flick:1
Time slot:112   LED_STATE:     0011111111111111,Flick:1
Time slot:114   LED_STATE:     0001111111111111,Flick:1
Time slot:116   LED_STATE:     0000111111111111,Flick:1
Time slot:118   LED_STATE:     0000011111111111,Flick:1
Time slot:120   LED_STATE:     0000001111111111,Flick:1
Time slot:122   LED_STATE:     0000000111111111,Flick:1
Time slot:124   LED_STATE:     0000000011111111,Flick:1
Time slot:126   LED_STATE:     0000000001111111,Flick:1
Time slot:128   LED_STATE:     0000000000111111,Flick:1
Time slot:130   LED_STATE:     0000000000011111,Flick:1
Time slot:132   LED_STATE:     1111111111111111,Flick:1
Time slot:134   LED_STATE:     0111111111111111,Flick:1
Time slot:136   LED_STATE:     0011111111111111,Flick:1
Time slot:138   LED_STATE:     0001111111111111,Flick:1
Time slot:140   LED_STATE:     0000111111111111,Flick:1
Time slot:142   LED_STATE:     0000011111111111,Flick:1
```

Output of system_verification_2_tb.v

3.2.4. Explanation:

At time slot 106 and 130, when LED[5] turns off, the flick signal remains high. The kickback_match generator will send a control signal to the state controller to set the state and counter registers to return to the previous state.

3.4. System verification 3:
(Source: system_verification_3_tb.v)

3.3.1. Description:
The testbench **system_verification_3_tb** module is used to verify behavior of kickback cases. In this testbench, we only verify the case when the flick signal is equal to 1 in the OFFLED10_0 state and LED[5] is turned off.

3.3.2. Expected result:
At LED[5], if the flick signal is 1, the system state will revert to the first behavior of the current state, contrary to LED[5] to LED[10] immediately turning on and gradually turning off afterwards.

3.3.3. Result:

```
Time slot:90    LED_STATE:      0001111111111111,Flick:0
Time slot:92    LED_STATE:      0000111111111111,Flick:0
Time slot:94    LED_STATE:      0000011111111111,Flick:0
Time slot:96    LED_STATE:      0000001111111111,Flick:0
Time slot:98    LED_STATE:      0000000111111111,Flick:0
Time slot:100   LED_STATE:      0000000011111111,Flick:0
Time slot:102   LED_STATE:      0000000001111111,Flick:0
Time slot:104   LED_STATE:      0000000000111111,Flick:0
Time slot:106   LED_STATE:      0000000000011111,Flick:0
Time slot:108   LED_STATE:      0000000000111111,Flick:0
Time slot:110   LED_STATE:      0000000001111111,Flick:0
Time slot:112   LED_STATE:      0000000011111111,Flick:0
Time slot:114   LED_STATE:      0000000111111111,Flick:0
Time slot:116   LED_STATE:      0000001111111111,Flick:0
Time slot:118   LED_STATE:      0000011111111111,Flick:0
Time slot:120   LED_STATE:      0000001111111111,Flick:1
Time slot:122   LED_STATE:      0000000111111111,Flick:1
Time slot:124   LED_STATE:      0000000011111111,Flick:1
Time slot:126   LED_STATE:      0000000001111111,Flick:1
Time slot:128   LED_STATE:      0000000000111111,Flick:1
Time slot:130   LED_STATE:      0000000000011111,Flick:1
Time slot:132   LED_STATE:      0000011111111111,Flick:1
Time slot:134   LED_STATE:      0000001111111111,Flick:1
Time slot:136   LED_STATE:      0000000111111111,Flick:1
Time slot:138   LED_STATE:      0000000011111111,Flick:1
Time slot:140   LED_STATE:      0000000001111111,Flick:1
Time slot:142   LED_STATE:      0000000000111111,Flick:1
Time slot:144   LED_STATE:      0000000000011111,Flick:1
Time slot:146   LED_STATE:      0000011111111111,Flick:1
Time slot:148   LED_STATE:      0000001111111111,Flick:1
Time slot:150   LED_STATE:      0000000111111111,Flick:1
Time slot:152   LED_STATE:      0000000011111111,Flick:1
Time slot:154   LED_STATE:      0000000001111111,Flick:1
Time slot:156   LED_STATE:      0000000000111111,Flick:1
Time slot:158   LED_STATE:      0000000000011111,Flick:1
Time slot:160   LED_STATE:      0000011111111111,Flick:1
Time slot:162   LED_STATE:      0000001111111111,Flick:1
Time slot:164   LED_STATE:      0000000111111111,Flick:1
Time slot:166   LED_STATE:      0000000011111111,Flick:1
Time slot:168   LED_STATE:      0000000001111111,Flick:1
Time slot:170   LED_STATE:      0000000000111111,Flick:1
Time slot:172   LED_STATE:      0000000000011111,Flick:1
Time slot:174   LED_STATE:      0000011111111111,Flick:1
Time slot:176   LED_STATE:      0000001111111111,Flick:1
Time slot:178   LED_STATE:      0000000111111111,Flick:1
Time slot:180   LED_STATE:      0000000011111111,Flick:1
Time slot:182   LED_STATE:      0000000001111111,Flick:1
Time slot:184   LED_STATE:      0000000000111111,Flick:1
Time slot:186   LED_STATE:      0000000000011111,Flick:1
Time slot:188   LED_STATE:      0000011111111111,Flick:1
Time slot:190   LED_STATE:      0000001111111111,Flick:1
Time slot:192   LED_STATE:      0000000111111111,Flick:1
Time slot:194   LED_STATE:      0000000011111111,Flick:1
Time slot:196   LED_STATE:      0000000001111111,Flick:1
```

Output of system_verification_3_tb.v

3.3.4. Explanation:

At time slot 130/144/158/172/186, when LED[5] turns off, the flick signal remains high. The kickback_match generator will send a control signal to the state controller to set the state and counter registers to return to the previous state.

3.5. System verification 4:

(Source: system_verification_4_tb.v)

3.4.1. Description:

The testbench **system_verification_4_tb** module is used to verify behavior of kickback cases. In this testbench, we only verify the case when the flick signal is equal to 1 in the OFFLED10_0 state and LED[0] is turned off.

3.4.2. Expected result:

At LED[0], if the flick signal is 1, the system state will revert to the first behavior of the current state, contrary to LED[0] to LED[10] immediately turning on and gradually turning off afterwards.

3.4.3. Result:

```
Time slot:116   LED_STATE:      0000001111111111,Flick:0
Time slot:118   LED_STATE:      0000011111111111,Flick:0
Time slot:120   LED_STATE:      0000001111111111,Flick:0
Time slot:122   LED_STATE:      0000000111111111,Flick:0
Time slot:124   LED_STATE:      0000000011111111,Flick:0
Time slot:126   LED_STATE:      0000000001111111,Flick:0
Time slot:128   LED_STATE:      0000000000111111,Flick:0
Time slot:130   LED_STATE:      0000000000011111,Flick:0
Time slot:132   LED_STATE:      0000000000001111,Flick:1
Time slot:134   LED_STATE:      0000000000000111,Flick:1
Time slot:136   LED_STATE:      0000000000000011,Flick:1
Time slot:138   LED_STATE:      0000000000000001,Flick:1
Time slot:140   LED_STATE:      0000000000000000,Flick:1
Time slot:142   LED_STATE:      0000011111111111,Flick:0
Time slot:144   LED_STATE:      0000001111111111,Flick:0
Time slot:146   LED_STATE:      0000000111111111,Flick:0
Time slot:148   LED_STATE:      0000000011111111,Flick:0
Time slot:150   LED_STATE:      0000000001111111,Flick:0
Time slot:152   LED_STATE:      0000000000111111,Flick:0
Time slot:154   LED_STATE:      0000000000011111,Flick:0
Time slot:156   LED_STATE:      0000000000001111,Flick:1
Time slot:158   LED_STATE:      0000000000000111,Flick:1
Time slot:160   LED_STATE:      0000000000000011,Flick:1
Time slot:162   LED_STATE:      0000000000000001,Flick:1
Time slot:164   LED_STATE:      0000000000000000,Flick:1
Time slot:166   LED_STATE:      0000011111111111,Flick:0
Time slot:168   LED_STATE:      0000001111111111,Flick:0
Time slot:170   LED_STATE:      0000000111111111,Flick:0
Time slot:172   LED_STATE:      0000000011111111,Flick:0
Time slot:174   LED_STATE:      0000000001111111,Flick:0
Time slot:176   LED_STATE:      0000000000111111,Flick:0
Time slot:178   LED_STATE:      0000000000011111,Flick:0
Time slot:180   LED_STATE:      0000000000001111,Flick:1
Time slot:182   LED_STATE:      0000000000000111,Flick:1
Time slot:184   LED_STATE:      0000000000000011,Flick:1
Time slot:186   LED_STATE:      0000000000000001,Flick:1
Time slot:188   LED_STATE:      0000000000000000,Flick:1
Time slot:190   LED_STATE:      0000011111111111,Flick:0
Time slot:192   LED_STATE:      0000001111111111,Flick:0
Time slot:194   LED_STATE:      0000000111111111,Flick:0
Time slot:196   LED_STATE:      0000000011111111,Flick:0
Time slot:198   LED_STATE:      0000000001111111,Flick:0
Time slot:200   LED_STATE:      0000000000111111,Flick:0
Time slot:202   LED_STATE:      0000000000011111,Flick:0
Time slot:204   LED_STATE:      0000000000001111,Flick:1
Time slot:206   LED_STATE:      0000000000000111,Flick:1
Time slot:208   LED_STATE:      0000000000000011,Flick:1
Time slot:210   LED_STATE:      0000000000000001,Flick:1
Time slot:212   LED_STATE:      0000000000000000,Flick:1
```

Output of system_verificaiton_4_tb.v

3.4.4. Explanation:

At time slot 142/166/180, when LED[0] turns off, the flick signal remains high. The kickback_match generator will send a control signal to the state controller to set the state and counter registers to return to the previous state.

4. History

Date	Author	Modified part	Description
2024/04/05		All	New creation