LAB 07 - Requirement Description

- **Introduction to Interrupt & Timer**

  Video(Interupt): https://youtu.be/9QEVDbKbzCU

  Slide1

  Video(Timer): https://youtu.be/0HzVE9Wbb74

  Slide2

  Hackmd

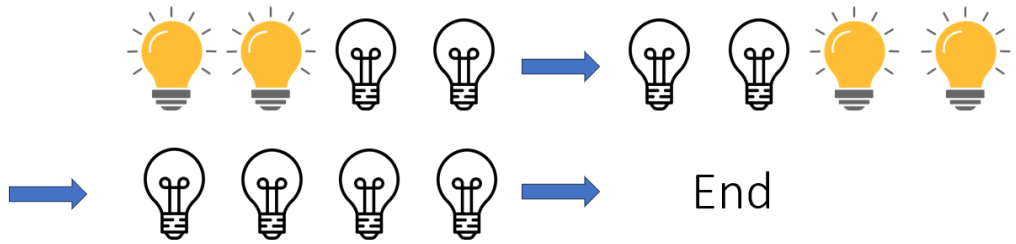
- **Basic (70%) :**

➢ **Description** :

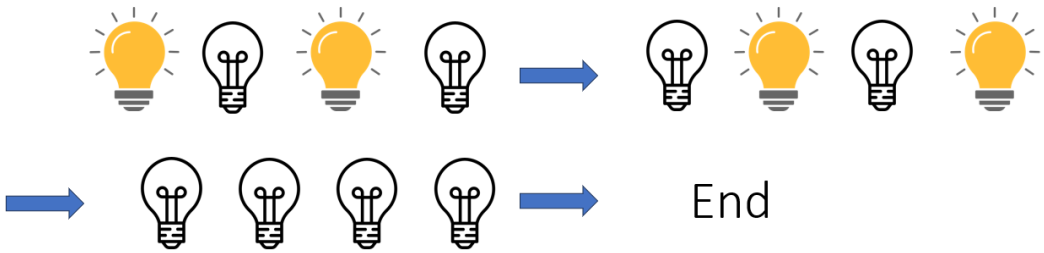
You will need four LED lights and one button, with the LEDs connected to the pins RA0, RA1, RA2, and RA3 in sequence from left to right. Additionally, a button is connected to the pin RB0. When the button is pressed, the state of the LEDs should be able to toggle.

➢ **Example** :

State 1:



State 2:



-click -> State 1
-click -> State 2
-click -> State 1 ……

➢ **Criteria** :

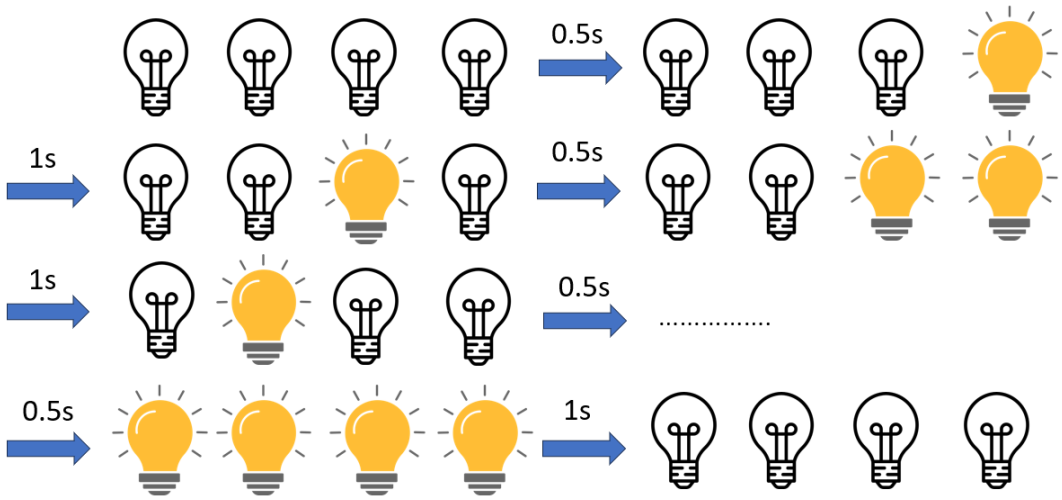
1. Use ISR to handle the button event.
2. Button: RB0, 4 LEDs: RA0~RA3.
3. Write in Assembly.

● **Advanced (30%) :**

➢ **Description** :

Create a cyclic counter ranging from 0x00 to 0x0F with the pins RA0, RA1, RA2, and RA3 representing four different bits. This setup should control LED lights to count with alternating intervals, starting with a 0.5-second delay, followed by a 1-second delay, and continuing in this alternating pattern. You must use TIMER2 to create the proper delay interval. You are not allowed to use DELAY macro.

➢ **Example** :



➢ **Criteria** :

1. Don't use DELAY macro int this program.
2. 4 LEDs: RA0~RA3.
3. Write in Assembly.

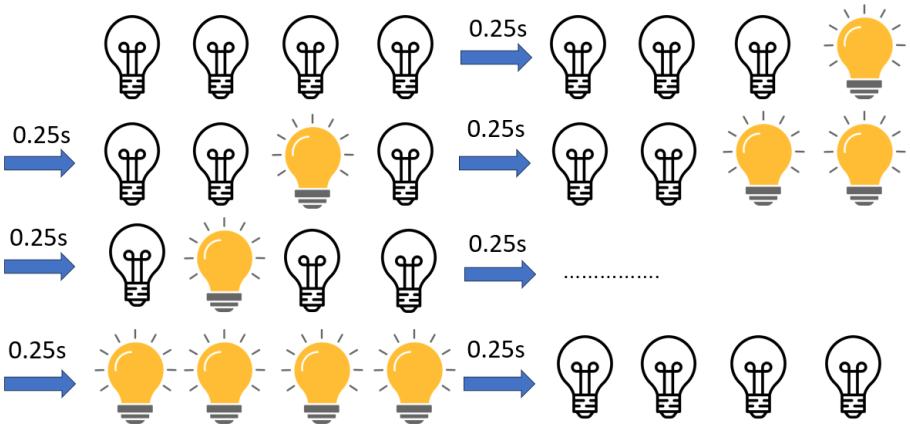
● **Bonus (20%) :**

➢ **Description** :

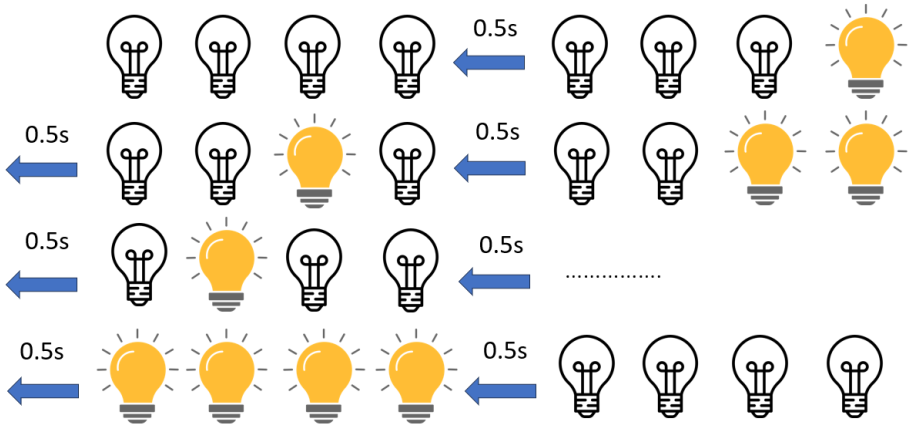
Implement a cyclic counter ranging from 0x00 to 0x0F that can count both up and down. The timer should have three interval states: 0.25 seconds, 0.5 seconds, and 1 second. Upon pressing a button, the timer should be able to switch both the counting direction and the interval state. The initial state is set to count up with a 0.25-second interval. The counting direction should cycle through the following pattern: counting up, then down, and back to up. The interval timing should progress: 0.25 second -> 0.5 second -> 1 second, and then back to 0.25 second. The state change should occur immediately upon button press.

➢ **Example** :

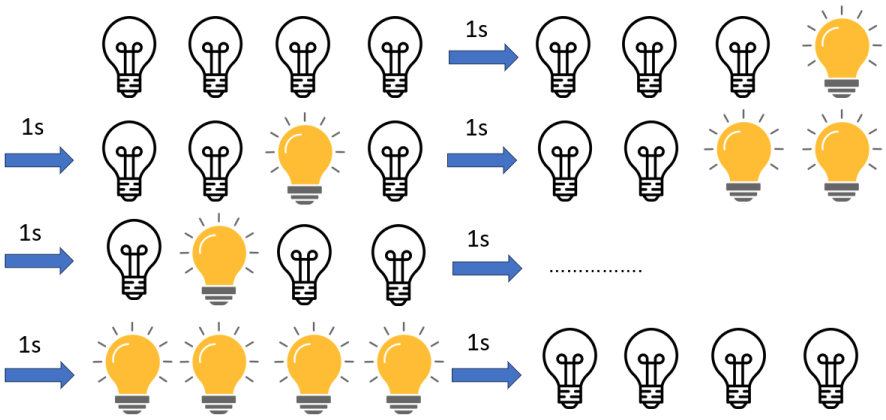
State 1:



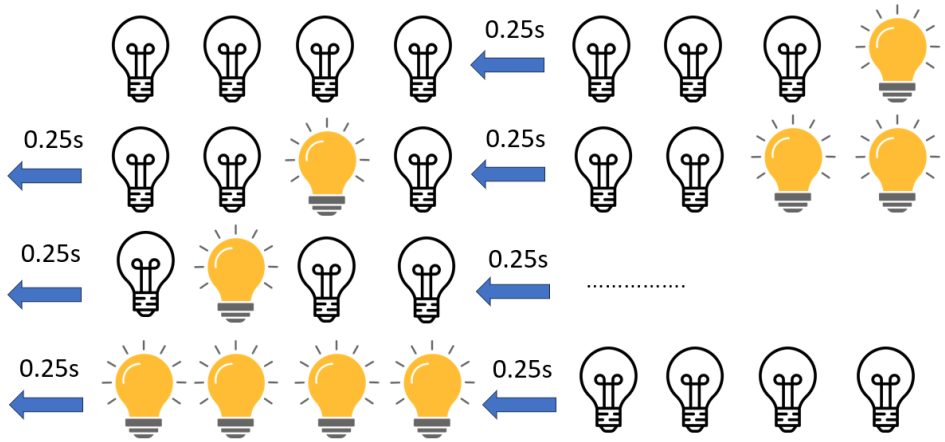
State 2:



State 3:



State 4:



Initial: State 1

-click -> State 2

-click -> State 3

-click -> State 4…….

➢ **Criteria** :

1. Use ISR to handle the button event.
2. Button: RB0, 4 LEDs: RA0~RA3.
3. Write in Assembly.

➢ **Hint** :

1. There are more than four states.