HW#4

Name: Mrinmoy Sarkar
Dept.:ECE
Course: ECEN-621

C code for sequential programming approach :

```c
#include "msp.h"


/**
 * main.c
 */

#define LED_GREEN BIT1
#define S1 BIT1
#define DELAY 500

void main(void)
{
    int i;
      WDT_A->CTL = WDT_A_CTL_PW | WDT_A_CTL_HOLD;           // stop watchdog timer
      P1->DIR &= ~S1;
      P1->REN = S1;
      P1->OUT = S1;
      P2->DIR = LED_GREEN;
      P2->OUT = 0x00;

      while(1)
      {
          if((P1->IN & S1) == 0x00)
          {
              for(i=0;i<DELAY;i++);
              if((P1->IN & S1) == 0x00)
              {
                  P2->OUT ^= LED_GREEN;
                  while((P1->IN & S1) == 0x00);
              }
          }
      }

}
```

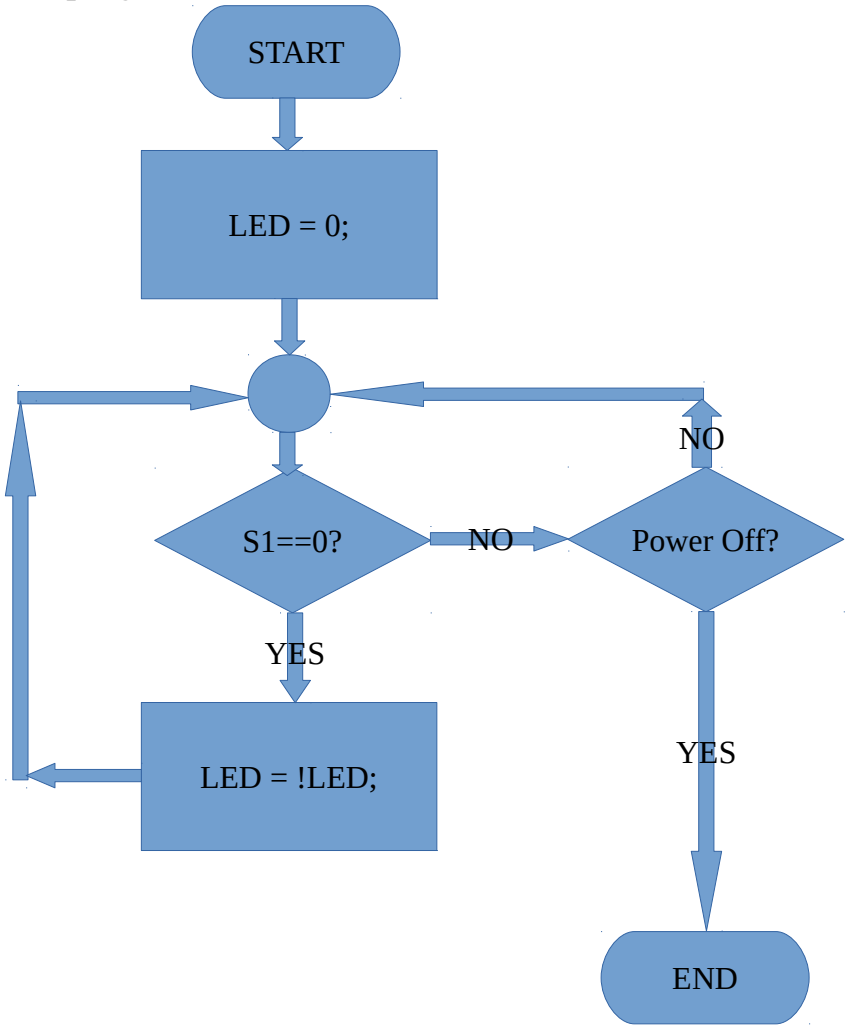
C code for state machine programming approach :

```c
#include "msp.h"


/**
 * main.c
 */

#define LED_GREEN BIT1
#define S1 BIT1
#define DELAY 500
int i;
enum Toggle_states {init_state, Unlit1_state, Unlit2_state, Lit1_state,
Lit2_state} toggle_states;

void TicFct_Toggle_Led()
{
    switch(toggle_states)
    {
```

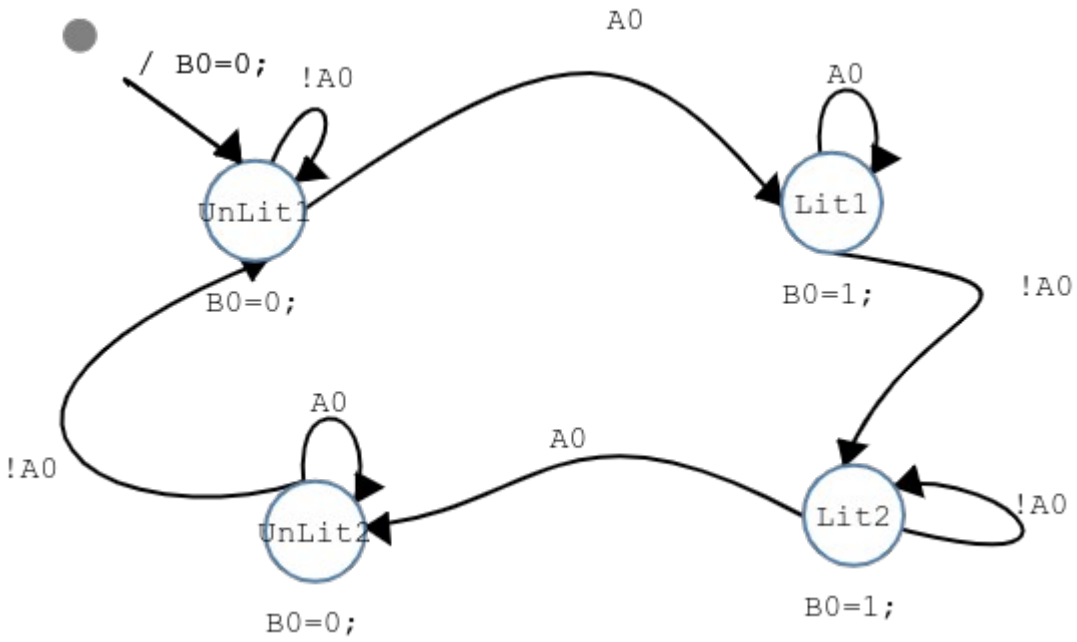
```
    case init_state:
        toggle_states = Unlit1_state;
        break;
    case Unlit1_state:
        if((P1->IN & S1) == 0x00)
        {
            for(i=0;i<DELAY;i++);
            if((P1->IN & S1) == 0x00)
            {
                toggle_states = Lit1_state;
            }
        }
        break;
    case Lit1_state:
        if((P1->IN & S1) != 0x00)
        {
            for(i=0;i<DELAY;i++);
            if((P1->IN & S1) != 0x00)
            {
                toggle_states = Lit2_state;
            }
        }
        break;
    case Lit2_state:
        if((P1->IN & S1) == 0x00)
        {
            for(i=0;i<DELAY;i++);
            if((P1->IN & S1) == 0x00)
            {
                toggle_states = Unlit2_state;
            }
        }
        break;
    case Unlit2_state:
        if((P1->IN & S1) != 0x00)
        {
            for(i=0;i<DELAY;i++);
            if((P1->IN & S1) != 0x00)
            {
                toggle_states = Unlit1_state;
            }
        }
        break;
    }
    switch(toggle_states)
    {
    case Unlit1_state:
    case Unlit2_state:
        P2->OUT &= ~LED_GREEN;
        break;
    case Lit1_state:
    case Lit2_state:
        P2->OUT |= LED_GREEN;
        break;
    }
}
```

```
void main(void)
{
    toggle_states = init_state;

      WDT_A->CTL = WDT_A_CTL_PW | WDT_A_CTL_HOLD;           // stop watchdog timer
      P1->DIR &= ~S1;
      P1->REN = S1;
      P1->OUT = S1;
      P2->DIR = LED_GREEN;
      P2->OUT = 0x00;

      while(1)
      {
          TicFct_Toggle_Led();
      }
}
```

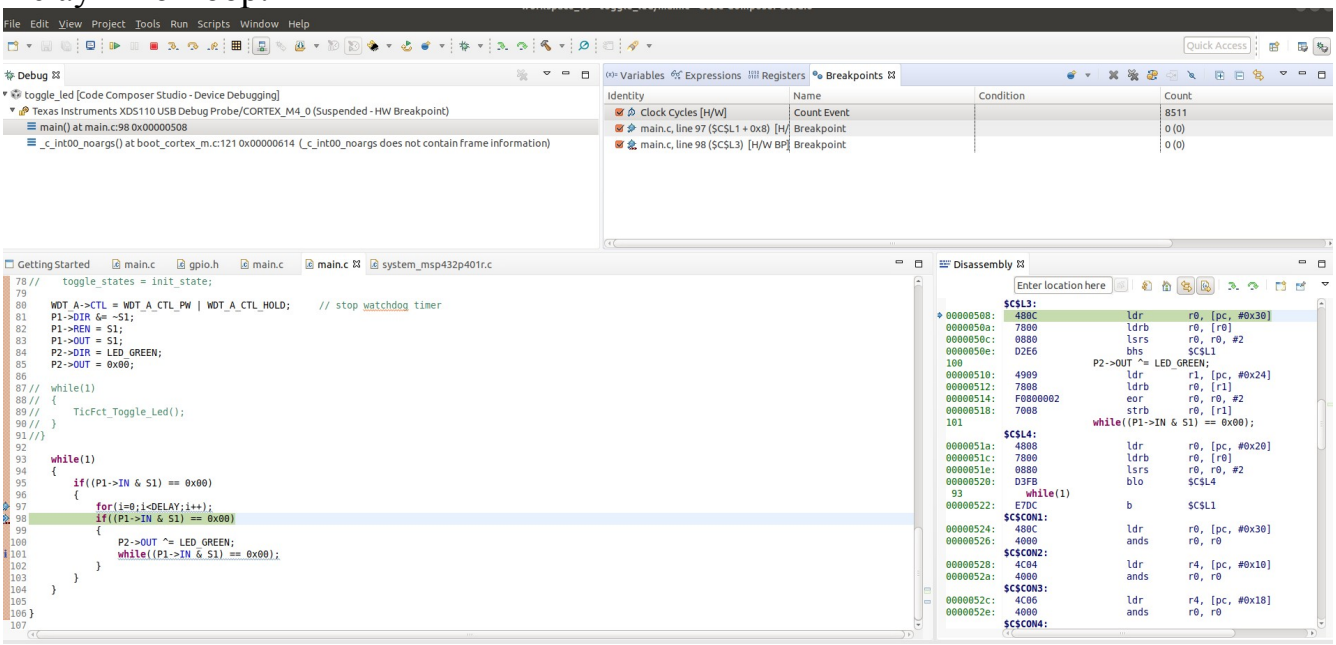
Flowchart of Sequential program:



RIBS model for the SM program:



Delay in for loop:




```
#define   __SYSTEM_CLOCK    3000000
```

The actual delay = 8511/3000000 = 0.002837 sec = 2.837 ms