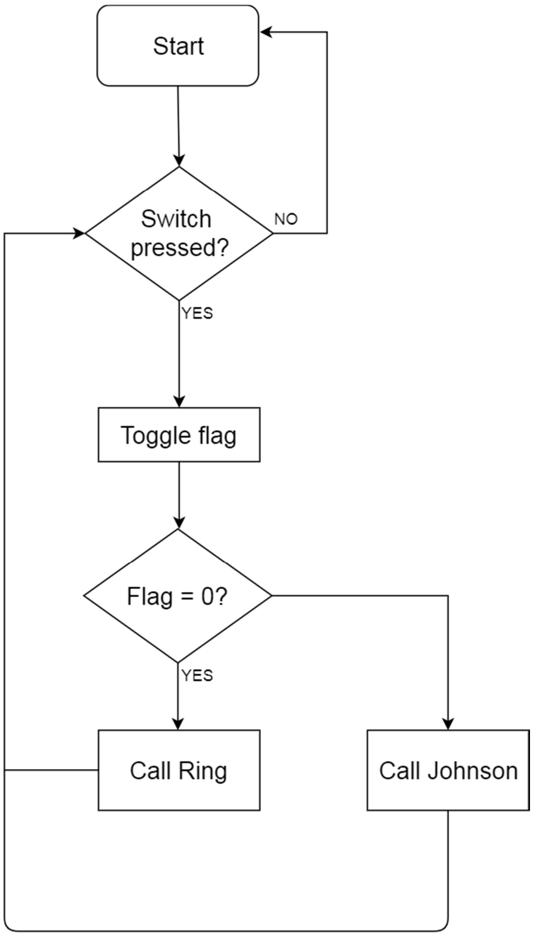
Task 1

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2017-09-12
; Author:
; Student name 1: Ruth Dirnfeld
; Student name 2: Alexandra Bjäremo
;
; Lab number: 2
; Title: Subroutines
;
; Hardware: STK600, CPU ATmega2560
;
; Function: Switching between the Ring and the Johnson counters when pressing SW0
;
; Input ports: On-board switches connected to PORTA.
;
; Output ports: On-board LEDs connected to PORTB.
;
; Subroutines: Delay of approximately 0,5 sec in between each count.
; Included files: m2560def.inc
;
; Other information: Clock set at 1MHz
;
; Changes in program: File updates: 2017-09-16, 2017-09-18, 2017-09-19
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

; Initialize SP, Stack Pointer
ldi r20, HIGH(RAMEND) ; R20 = high part of RAMEND address
out SPH,r20 ; SPH = high part of RAMEND address
ldi R20, low(RAMEND) ; R20 = low part of RAMEND address
out SPL,R20 ; SPL = low part of RAMEND address

ldi r20, 0xFF
out DDRB, r20                   ; All one's to DDRB, outputs

ldi r16, 0xFE                   ; starting with LED0
ldi r17, 0x00                   ; temp register to help with sloop

ldi r23, 0x00
out DDRA, r23
```

```
ldi r22, 0x00                  ; keep track of counters, if 0 then ring, else johnson


main:
    ring:
        ldi r23, 0x00
        cpi r16, 0xFF          ; checking if all LEDs are off
        breq equal
        out PORTB, r16         ; write in PORTB, turning on LEDs
        com r16                ; inverting the bits of r16
        lsl r16                ; pushing a 0 to the left
        com r16                ; inverting the bits of r16 again
        rcall delay
        rjmp ring

     equal:
        ldi r16, 0xFE
     rjmp main

; Generated by delay loop calculator
; at http://www.bretmulvey.com/avrdelay.html
;
; Delay 500 000 cycles
; 500ms at 1 MHz

delay:
    ldi  r18, 3
    ldi  r19, 138
    ldi  r21, 86
L1: dec  r21
    brne L1
    dec  r19
    brne L1
    dec  r18
    brne L1
    in r23, PINA               ; read PINA
    cpi r23, 0xFE              ; check if button is pressed
    breq pressed
    rjmp PC+1
 ret

;;;;;;;;;;;;;;;;;;;;;;;;;;TASK6
johnsonLeft:
ldi r23, 0x00
```

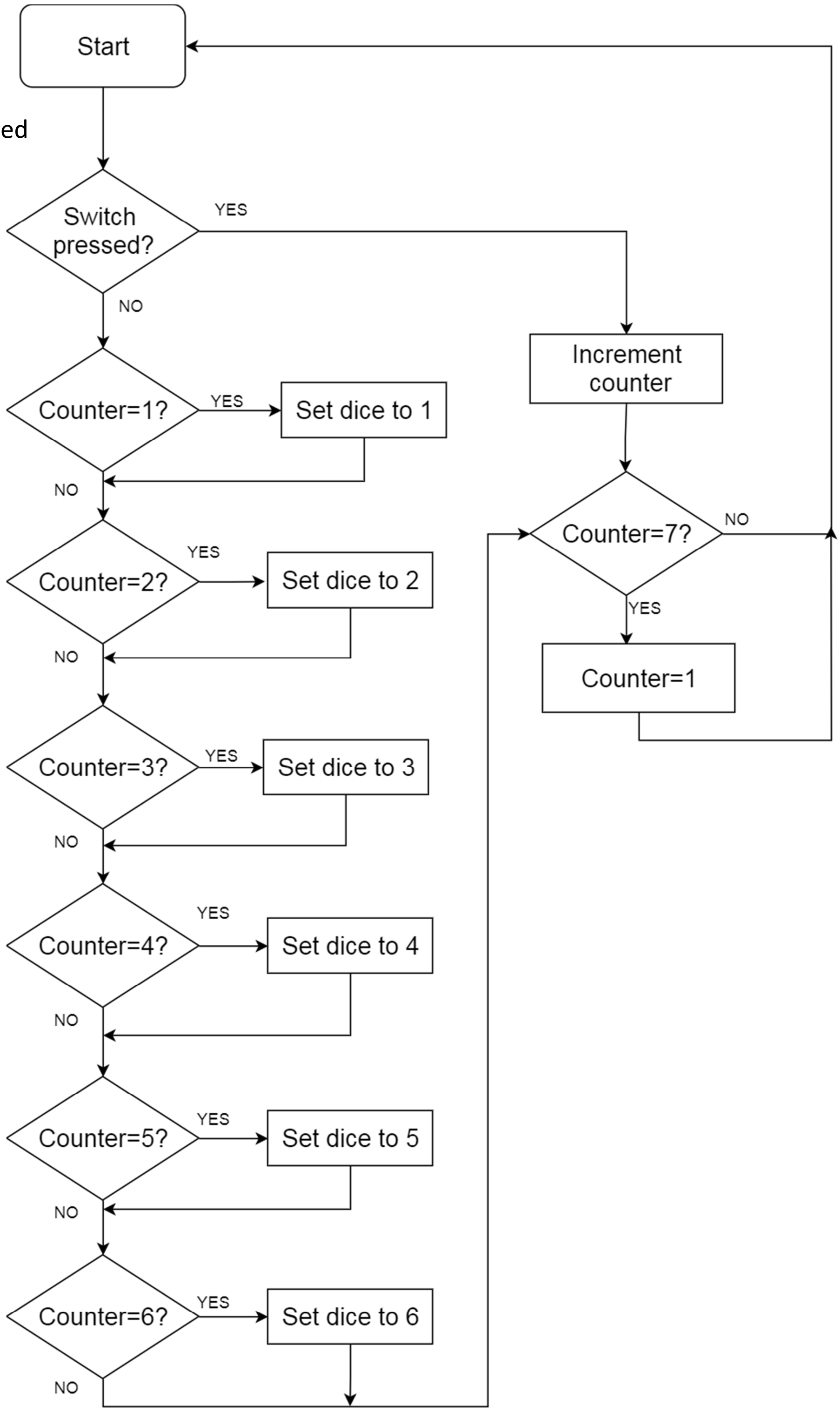


```
cpi r16, 0x00               ; check if all LEDs are on
breq johnsonRight
out PORTB, r16              ; write to PORTB
lsl r16                     ; pushing 0 to the left to turn on next light aswell
rcall delay
rjmp johnsonLeft

johnsonRight:
ldi r23, 0x00
out PORTB, r16              ; write to PORTB
cpi r16, 0xFF               ; check if all LEDs are off
breq johnsonLeft
mov r17, r16                ; move r16's bits to r17
com r17                     ; invert r17's bits
lsr r17                     ; pushing 0 to the right
com r17                     ; invert r17's bits again
mov r16, r17                ; move r17's bits to r16
rcall delay
rjmp johnsonRight

pressed:
com r22                     ; invert r22 to change counter
cpi r22,0x00                ; check if counter is set to Ring counter
breq jumpToRing
rjmp jumpToJohnsonLeft

jumpToRing:
ldi r16, 0xFF
out PORTB, r16              ; turn off all LEDs
rjmp ring

jumpToJohnsonLeft:
ldi r16, 0xFF
out PORTB, r16              ; turn off all LEDs
rjmp johnsonLeft


/*Description:
*Starting the main program with the Ring counter, while in the delay phase the switch is checked. If
*the switch has been pressed, then the program the flag is toggled (from 0 to 1 or 1 to 0) so that the
*next counter will start. However, it will turn off all the LEDs before starting the counter.
*/
```

Task 2

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2017-09-16
; Author:
; Student name 1 Ruth Dirnfeld
; Student name 2 Alexandra Bjäremo
;
; Lab number: 2
; Title: Subroutines
;
; Hardware: STK600, CPU ATmega2560
;
; Function: Rolling the electronical dice by randomly generating the number
; with the switch
;
; Input ports: On-board switches connected to PORTA.
;
; Output ports: On-board LEDs connected to PORTB.
;
; Subroutines: Loops to help with the rolling of the dice and outputting.
; Included files: m2560def.inc
;
; Other information: Clock set at 1MHz
;
; Changes in program: None
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"
ldi r16, 0xFF
out DDRB, r16                   ; All one's to DDRB, output

ldi r16, 0x00
out DDRA, r16                   ;  All zero's to DDRA, input

ldi r17,1                       ; r17 to help with output

loop:
in r18, PINA                    ; read PINA
cpi r18, 0xFF                   ; check if button is pressed
breq output
inc r17                         ; increment r17
cpi r17, 7                      ; check if value is 7
```

```
brne jmpLoop
ldi r17, 1
jmpLoop:
rjmp loop

;Checking which value is registered
output:
cpi r17, 1
breq one
cpi r17, 2
breq two
cpi r17, 3
breq three
cpi r17, 4
breq four
cpi r17, 5
breq five
cpi r17, 6
breq six

; Depending on the value
;turn on corresponding LEDs
one:
ldi r16, 0b0001_0000
out DDRB, r16
rjmp loop
two:
ldi r16, 0b0100_0100
out DDRB, r16
rjmp loop
three:
ldi r16, 0b0101_0100
out DDRB, r16
rjmp loop
four:
ldi r16, 0b1100_0110
out DDRB, r16
rjmp loop
five:
ldi r16, 0b1101_0110
out DDRB, r16
rjmp loop
six:
ldi r16, 0b1110_1110
out DDRB, r16
```

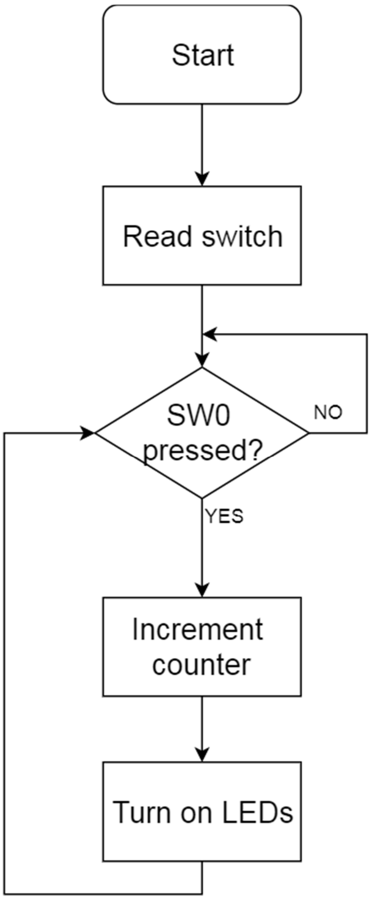


```
rjmp loop
/*Description:
*Checking if the switch is pressed, then incrementing the counter until it reaches 7 then resetting it
*to 1 and turning on the corresponding LEDs. This will generate a visual effect that resembles two
*dice.
*/
```

Task 3

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2017-09-17
; Author:
; Student name 1 Ruth Dirnfeld
; Student name 2 Alexandra Bjäremo
;
; Lab number: 2
; Title: Subroutines
;
; Hardware: STK600, CPU ATmega2560
;
; Function: A program that is able to count the number of changes on a switch.
;
; Input ports: On-board switches connected to PORTA.
;
; Output ports: On-board LEDs connected to PORTB.
;
; Subroutines: None.
;
; Included files: m2560def.inc
;
; Other information: Using a loop to make the counter work infinitely.
;                    Clock set at 1MHz.
;
; Changes in program: File created - 2017-09-18.
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

.def counter = r17

ldi counter, 0x00
rjmp main
```

```
switch:
inc counter                 ; pushed

loop:
in r16, PINA                ; read PINA
cpi r16, 0xFF               ; release all switches
brne loop
inc counter                 ; released

out DDRB, counter

main:
in r16, PINA                ; read PINA
cpi r16, 0xFE               ; switch 0 is pushed
breq switch

rjmp main
```



```
/*Description:
*Reading the switch 0 to see if it is pressed. If it is
*pressed the counter increases by one and is then ported to the LEDs. Otherwise it waits.
*/
```

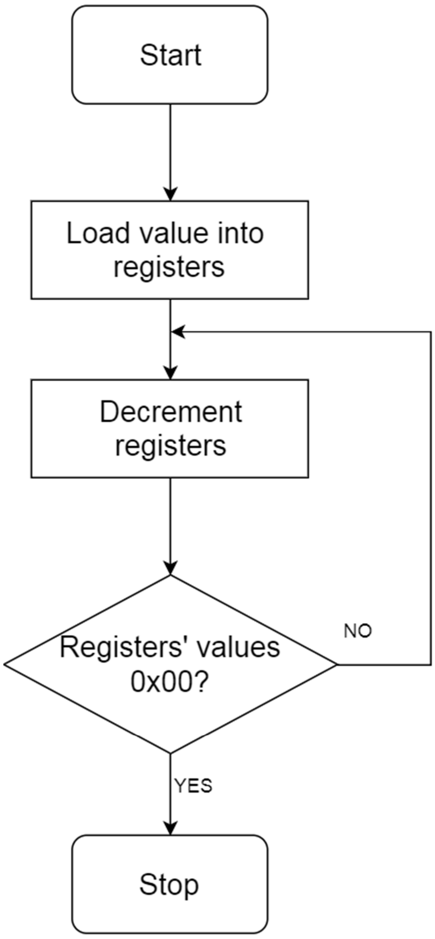
Task 4

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2017-09-18
; Author:
; Student name 1 Ruth Dirnfeld
; Student name 2 Alexandra Bjäremo
;
; Lab number: 2
; Title: Subroutines.
;
; Hardware: STK600, CPU ATmega2560
;
; Function: Adaptation of Ring counter, in which the delay is changeable depending
; on the decided input
;
; Input ports: None.
;
```

```
; Output ports: On-board LEDs connected to PORTB.
;
; Subroutines: Delay that is changeable.
; Included files: m2560def.inc
;
; Other information: Clock set at 1MHz.
;
; Changes in program: None.
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

; Initialize SP, Stack Pointer
ldi r20, HIGH(RAMEND) ; R20 = high part of RAMEND address
out SPH,r20 ; SPH = high part of RAMEND address
ldi R20, low(RAMEND) ; R20 = low part of RAMEND address
out SPL,R20 ; SPL = low part of RAMEND address

ldi r20, 0xFF
out DDRB, r20                 ; All one's to DDRB, outputs

ldi r16, 0xFE                 ; starting with LED0

floop:
cpi r16, 0xFF                 ; checking if all LEDs are off
breq equal
out PORTB, r16                ; write in PORTB, turning on LEDs
com r16                       ; inverting the bits of r16

lsl r16                       ; pushing a 0 to the left
com r16                       ; inverting the bits of r16 again

ldi r24, low(600)             ; Loading integer to register pair r25:r24
ldi r25, high(600)
rcall wait_milliseconds       ;  Call the wait_milliseconds subroutine
rjmp floop

equal:
ldi r16, 0xFE

rjmp floop
```

```
; Depending on the integer value set in the registers the
; delay will either increase or decrease
wait_milliseconds:

L:
ldi r20, low(600)            ; loading part of the integer
ldi r21, high(600)           ; loading part of the integer

L1:
dec r20                      ; decrement r20
nop
brne L1

dec r21                      ; decrement r21
nop
brne L1

sbiw r25:r24, 1              ; substract immediate from word
brne L

ret
```

DELAY FLOWCHART



```
/*Description:
*The program starts a Ring counter that will be turning on one LED at a time, while turning off the
*previous one. The speed with which this will be done depends on the value set in the delay. In the
*delay the value is split into two registers and then the registers are being decremented until their
*values are 0x00 at which point the program returns to the ring counter loop.
*/
```