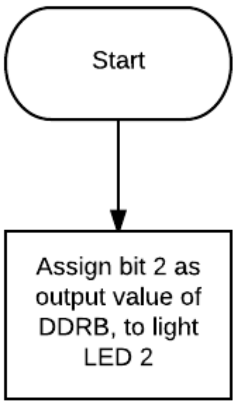
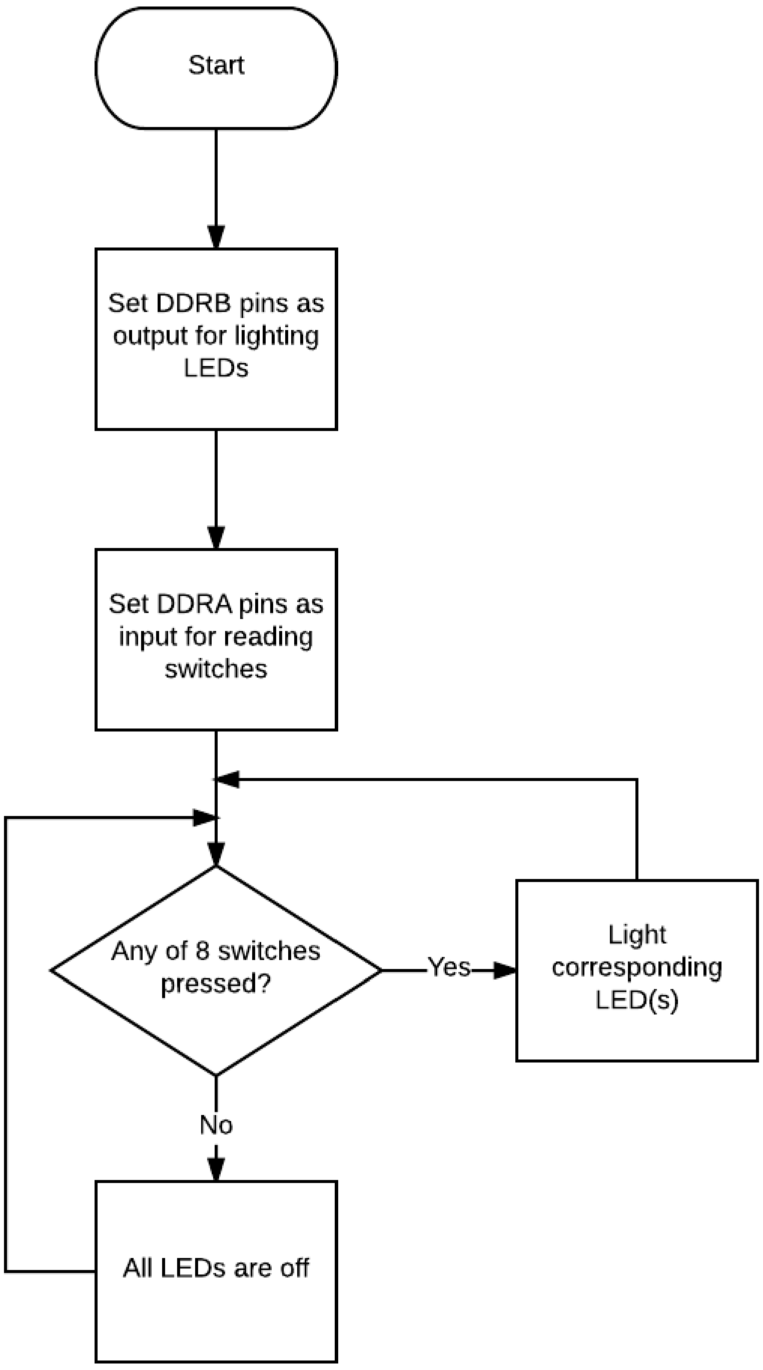
Task 1

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2016-09-05
; Authors:
; Mehdi Hmidi
; Jorian Wielink
;
; Lab number: 1
; Title: Task 1
;
; Hardware: STK600, CPU ATmega2560
;
; Function: To light LED 2 using PORTB as output
;
; Input ports: None
;
; Output ports: on-board LEDs connected to PORTB.
;
; Subroutines: If applicable.
; Included files: m2560def.inc
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ldi r16, 0x04       ; put decimal value 4 in register
out 0x04, r16       ; output value of r16 to address 0x04 (DDRB)
```



Task 2

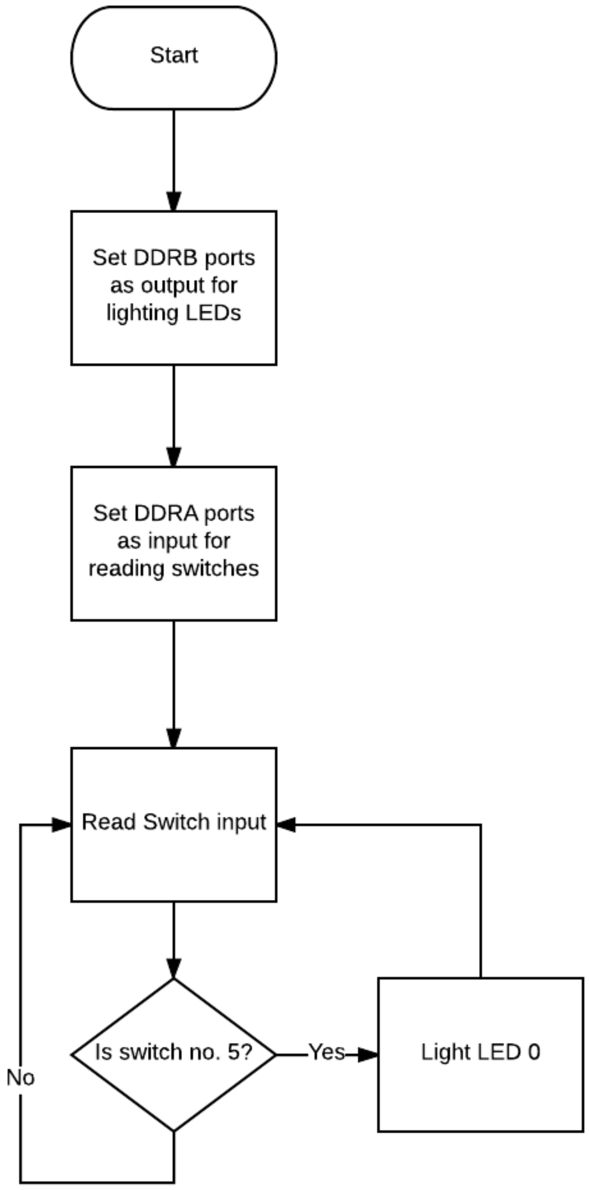
```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2016-09-05
; Authors:
; Mehdi Hmidi
; Jorian Wielink
;
; Lab number: 1
; Title: Task 2
;
; Hardware: STK600, CPU ATmega2560
;
; Function: To read the switches and light the corresponding LED
;
; Input ports: Switches connected to port A
;
; Output ports: Leds connected to port B.
;
; Included files: m2560def.inc
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

ldi r16, 0xFF
out DDRB, r16                ; set DDRB as output for lighting LEDs


ldi r16, 0x00                ; set DDRA as input for reading switches
out DDRA, r16

loop:                        ; Keep a loop running indefinitely
   in r15, PINA              ; Read switch info from PIN A
   out PORTB, r15            ; Send this info to light LED on Port B.

rjmp loop
```


Start
Set DDRB pins as output for lighting LEDs
Set DDRA pins as input for reading switches
Any of 8 switches pressed?
Yes
Light corresponding LED(s)
No
All LEDs are off

Task 3 + 4

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2016-09-05
; Authors:
; Mehdi Hmidi
; Jorian Wielink
;
; Lab number: 1
; Title: Task 3
;
; Hardware: STK600, CPU ATmega2560
;
; Function: To make a program to light LED0 when Switch 5 is pressed.
;
; Input ports: Switches connected to port A
;
; Output ports: Leds connected to port B.
;
; Included files: m2560def.inc
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

ldi r16, 0xFF
out DDRB, r16               ; set DDRB as output for lighting LEDs

ldi r16, 0x00
out DDRA, r16               ; set DDRA as input for reading switches switches

ldi r16, 0xFF
out portB, r16

ldi r17, 0b11011111         ; values set in register to compare to in loop
ldi r18, 0b11111110

loop:
   in r16, PINA             ; Read switch info from PIN A
   cp r16, r17              ; Compare with r17, to check if SW5 is pressed
   breq equal               ; if so, branch to ‘equal’

rjmp loop
```

```
equal: out portB, r18       ; when run, this lights led 0
```

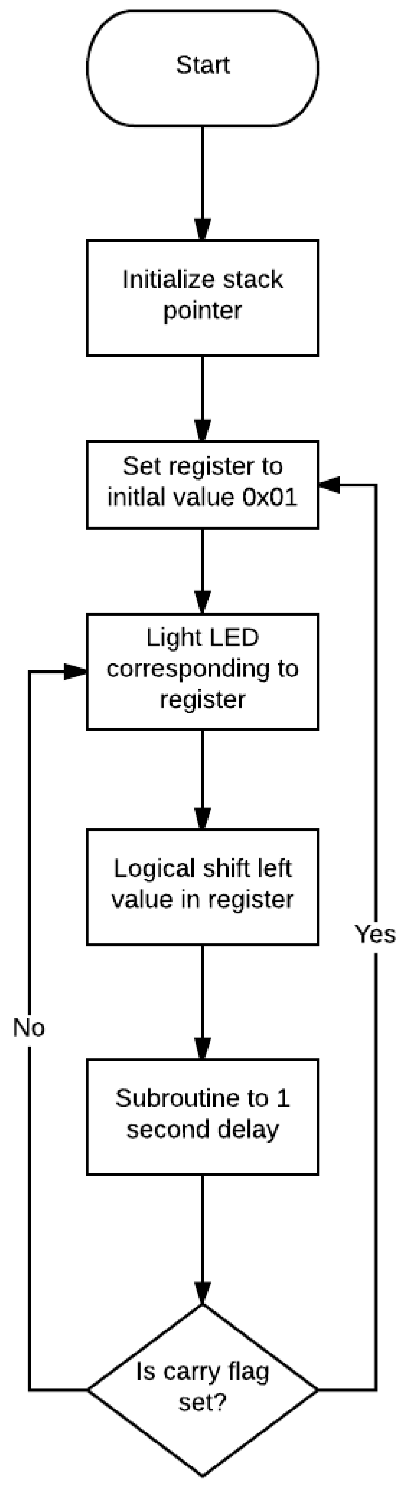

Start
Set DDRB ports as output for lighting LEDs
Set DDRA ports as input for reading switches
Read Switch input
Is switch no. 5?
Yes
Light LED 0
No

Notes regarding task 4: To light a LED, the bits representing the pins need to be inverted, because the pins are connected to ground on one side and will pull the port pin to ground (0) once they're pressed. This will result in a zero for a pressed button and a 1 for a released one. Once the port pin is low, it will sink current and the LED is lit.
Simplified: because of current and voltage specifications of the board, we have to invert the bits on the port pins to light the LED.
This was a little ‘gotcha’ while running the code in the simulator.

Task 5

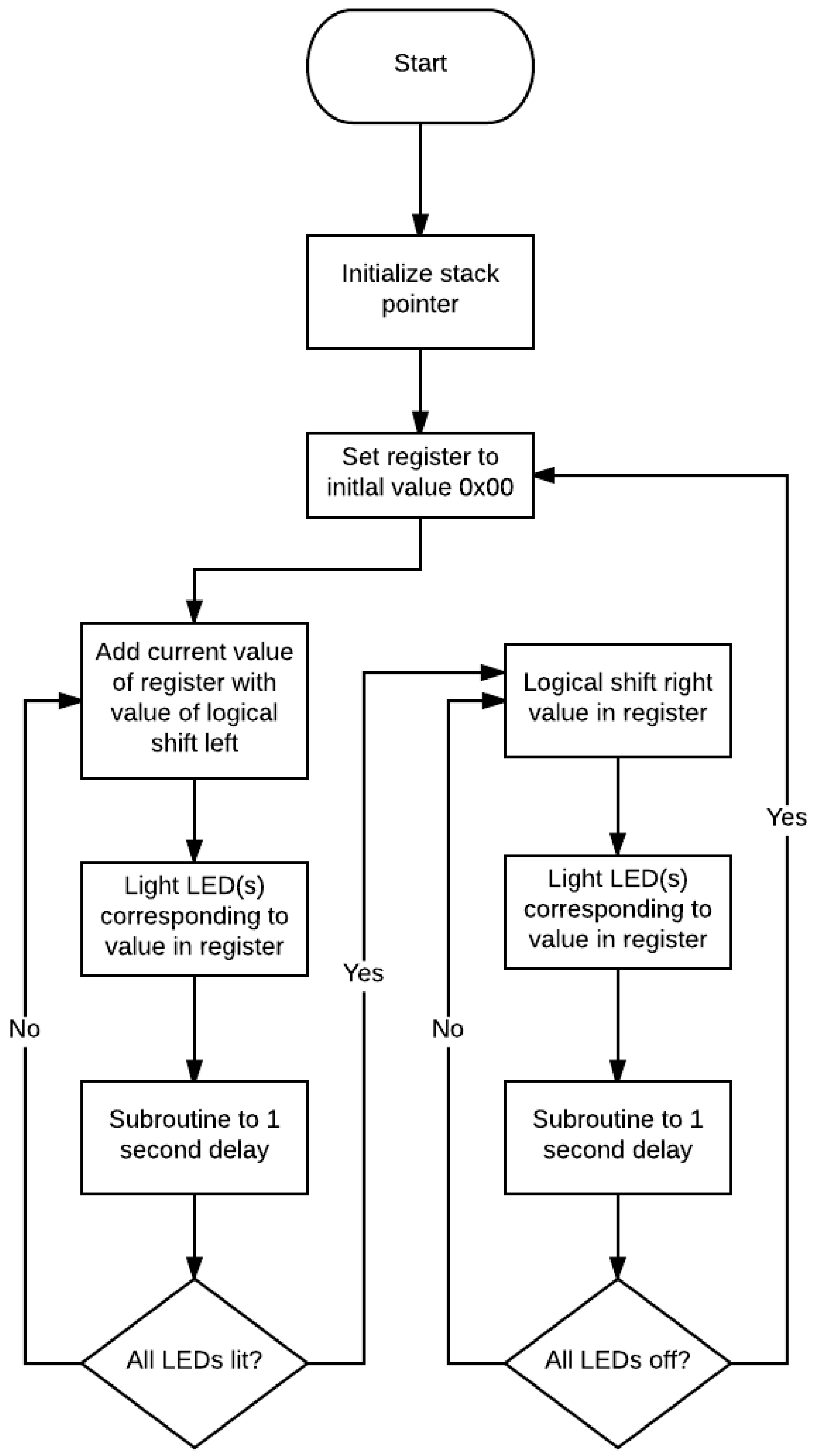
```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2016-09-05
; Authors:
; Mehdi Hmidi
; Jorian Wielink
;
; Lab number: 1
; Title: Task 5
;
; Hardware: STK600, CPU ATmega2560
;
; Function: Run a ring counter on the LEDS
;
; Input ports: None
;
; Output ports: Leds connected to port B.
;
; Subroutines: delay: accounts for 1 second delay between changing LED.
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

; Initialize SP, Stack Pointer
ldi r19, high(ramend)
out sph, r19
ldi r19, low(ramend)
out spl, r19

init:
   ldi r19, 0x01

loop:
   out DDRB, r19
   lsl r19              ; logical shifts value of r19 to the left
   rcall delay          ; the delay is good for 16.000.000 cpu cycles,
                        ; representing 1 second. (since cpu runs on 16Mhz)
   brcs init            ; branch to start at init again if carry flag is set
                        ; (resets LED to first bit in Port B).

rjmp loop
```

```
delay:                  ; 16MHz -> 16000000 cycles = 1s,
                        ; Cycles = 3a + 4ab + 3abc   ->  a(3 + b(4 + 3c))
   ldi r16, 11          ; -> a

delay_1:
   ldi r17, 237         ; -> b

delay_2:
   ldi r18, 255         ; -> c  (a,b,c ~> 0.5s)

delay_3:
   dec r18              ; decrements value of r18 until 0 because
                        ; it will branch to delay_3 again if not zero.
   brne delay_3         ; 3c - 1        -> d = 3c - 1

   dec r17
   nop                  ; delay_2 resets r18 to 255 for 237 times because
                        ; r17 is decremented each time when delay_3 is zero.
   brne delay_2         ; 5b - 1 + bd   -> e = 5b - 1 + 3cb - b

   dec r16              ; the whole process above is repeated 11 times.
   brne delay_1         ; 5a - 1 + ae   -> f = 3a + 5ab + 3abc - ab

   ret                  ; f - 1
```


Start
Initialize stack pointer
Set register to initlal value 0x01
Light LED corresponding to register
Logical shift left value in register
Subroutine to 1 second delay
Is carry flag set?
No
Yes

Task 6

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2016-09-05
; Authors:
; Mehdi Hmidi
; Jorian Wielink
;
; Lab number: 1
; Title: Task 6
;
; Hardware: STK600, CPU ATmega2560
;
; Function: To run a Johnson Counter
;
; Input ports: None
;
; Output ports: Leds connected to port B.
;
; Subroutines: delay: accounts for 1 second delay between changing LED.
;
; Included files: m2560def.inc
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

;Initialize stack pointer
ldi r16, high(ramend)
out sph, r16
ldi r20, low(ramend)
out spl, r20

ldi r19, 0x00
ldi r20, 0x01

_add:
   out DDRB, r19
   add r19, r20         ; adds current value r19 with logical left shifted r20
   lsl r20              ; to keep lighting LEDs until bit 7.
   rcall delay          ; have a delay of 1 second between lighting LEds.

   cpi r19, 0xff
   brne _add            ; run _add until all LEDs are lit.

_sub:

   out DDRB, r19
   lsr r19
   rcall delay

   brcs _sub             ; branch if carry flag is set in status register

   ldi r20, 0x01
   rjmp _add

delay:                  ; 16MHz -> 16000000 cycles = 1s,
                        ; Cycles = 3a + 4ab + 3abc   ->  a(3 + b(4 + 3c))
   ldi r16, 11          ; -> a

delay_1:
   ldi r17, 237         ; -> b

delay_2:
   ldi r18, 255         ; -> c  (a,b,c ~> 0.5s)

delay_3:
   dec r18              ; decrements value of r18 until 0 because
                        ; it will branch to delay_3 again if not zero.
   brne delay_3         ; 3c - 1        -> d = 3c - 1

   dec r17
   nop                  ; delay_2 resets r18 to 255 for 237 times because
                        ; r17 is decremented each time when delay_3 is zero.
   brne delay_2         ; 5b - 1 + bd   -> e = 5b - 1 + 3cb - b

   dec r16              ; the whole process above is repeated 11 times.
   brne delay_1         ; 5a - 1 + ae   -> f = 3a + 5ab + 3abc - ab

   ret                  ; f - 1
```


Start
Initialize stack pointer
Set register to initlal value 0x00
Add current value of register with value of logical shift left
Light LED(s) corresponding to value in register
Subroutine to 1 second delay
All LEDs lit?
No
Yes
Logical shift right value in register
Light LED(s) corresponding to value in register
Subroutine to 1 second delay
All LEDs off?
No
Yes