Laboration 1

How to use PORTs. Digitial I/O & subroutine calls.

1DT301

Christoffer Lundström

Marcus Johansson

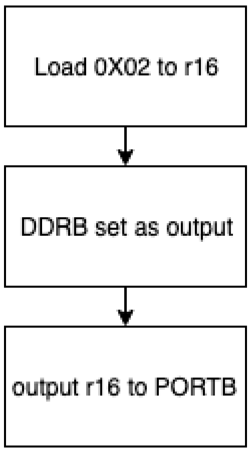
September 18, 2019

Task 1:

A program which lights LED number two.

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2019-09-13
; Author:
; Christoffer Lundström
; Marcus Johansson
;
; Lab number: 1
; Title: Task 1.
;
; Hardware: Arduino UNO Rev 3, CPU ATmega328P
;
; Function: This program lights LED 2.
;
; Input ports: No input in this task.
;
; Output ports: PORTB used to light led.
;
; Subroutines: Start. Main entry point of application.
;
; Other information:
;
; Changes in program: 13/9 2019
;
; Results: Leds are lit by first setting data direction and then setting PORTB to output with bitmask.
; The minimum amount of instructions to light the leds are 3 instructions. One ldi register, one set ddr and one output.
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
start:
    ldi r16, 0b0000_0010 ; Load Bit to Register 16
    out DDRB, r16; Set data direction from register 16
    out PORTB, r16; Output bitmask to portb
```

Notes: To find the minimum amount of instructions needed to light the LED we use the same register to both set DDRB and output PORTB.

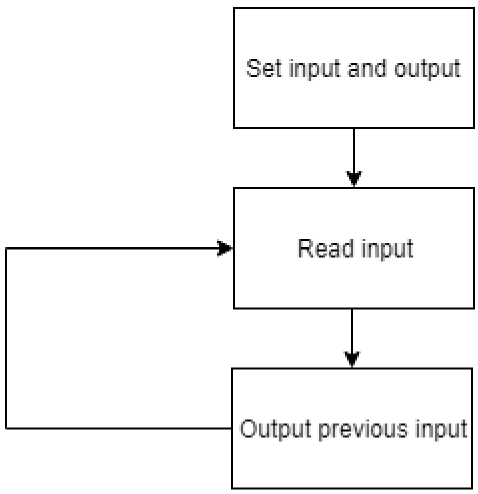


Task 2:

Reads a switch which lights corresponding LED.

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2019-09-13
; Author:
; Christoffer Lundström
; Marcus Johansson
;
; Lab number: 1
; Title: Task 2.
;
; Hardware: Arduino UNO Rev 3, CPU ATmega328P
;
; Function: Lights LED using corresponding button.
;
; Input ports: PIND
;
; Output ports: PORTB
;
; Subroutines: init, loop
;
; Other information: We use the input bit as bitmask for output.
;
; Changes in program: 13/9 2019
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
init:
    ldi r16, 0x00 ; set r16 to 0
    ldi r17, 0xFF ; set r17 to 1
    out DDRD, r16 ; set DDRD to input
    out DDRB, r17 ; set DDRB to output

    loop:
    in r16, PIND ; read pin value to r16
    out PORTB, r16; output r16
    rjmp loop
```

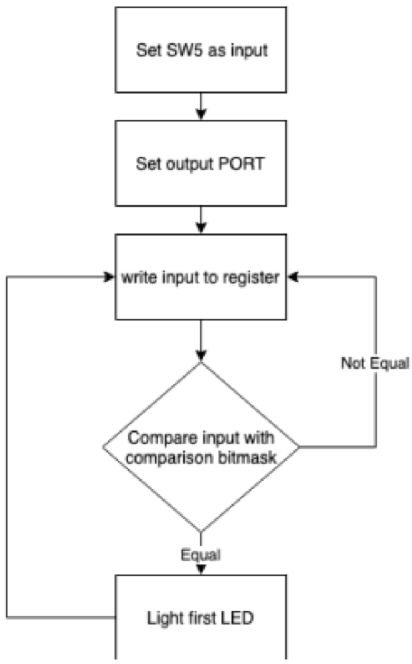
Notes: By using the same register for input and output we save some instructions.



Task 3:

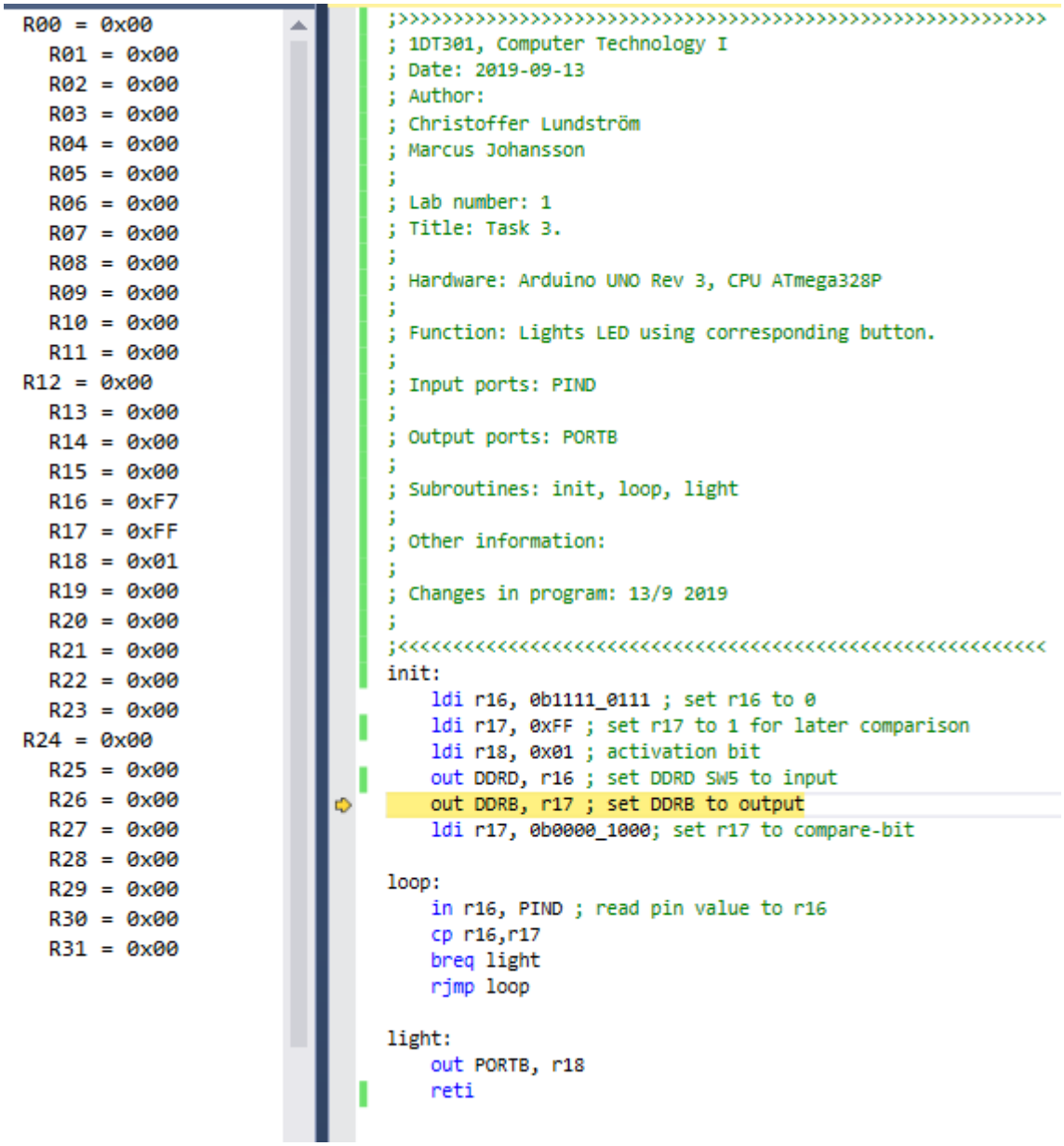
Lights LED 0 when switch 5 is pressed.

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2019-09-13
; Author:
; Christoffer Lundström
; Marcus Johansson
;
; Lab number: 1
; Title: Task 3.
;
; Hardware: Arduino UNO Rev 3, CPU ATmega328P
;
; Function: Lights LED using corresponding button.
;
; Input ports: PIND
;
; Output ports: PORTB
;
; Subroutines: init, loop, light
;
; Other information:
;
; Changes in program: 13/9 2019
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
init:
    ldi r16, 0b1111_0111 ; set r16 to 0
    ldi r17, 0xFF ; set r17 to 1 for later comparison
    ldi r18, 0x01 ; activation bit
    out DDRD, r16 ; set DDRD SW5 to input
    out DDRB, r17 ; set DDRB to output
    ldi r17, 0b0000_1000; set r17 to compare-bit

loop:
    in r16, PIND ; read pin value to r16
    cp r16,r17
    breq light
    rjmp loop

light:
    out PORTB, r18
    reti
```



Task 4:

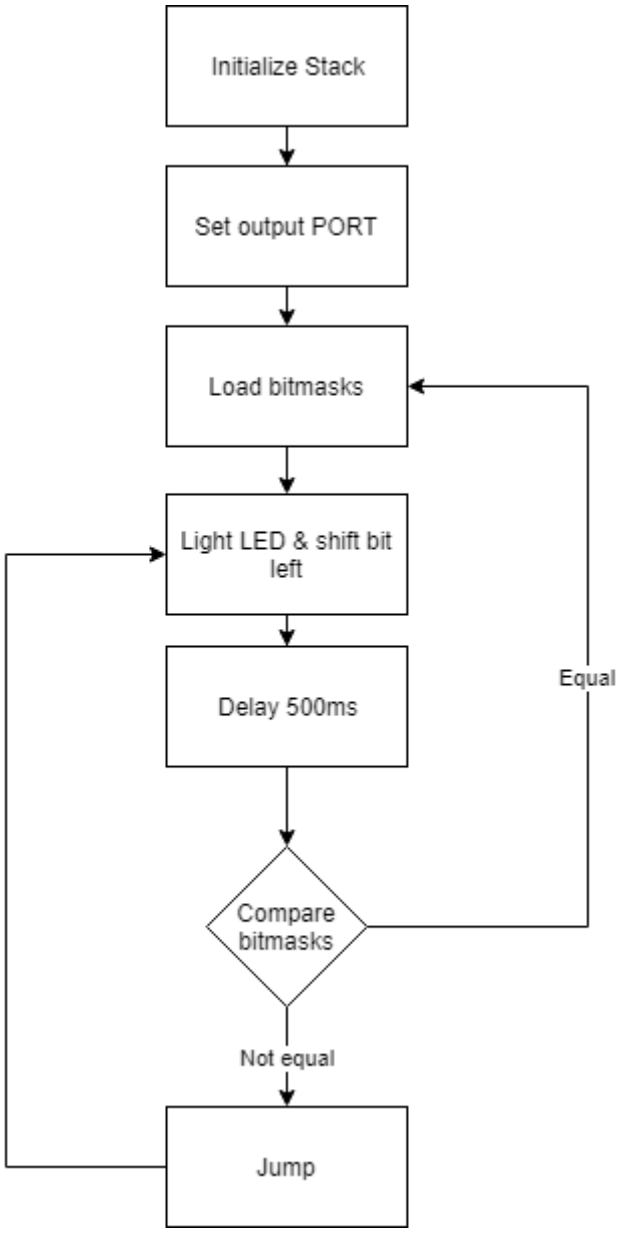
Debugger running.



For flowchart see Task 3.

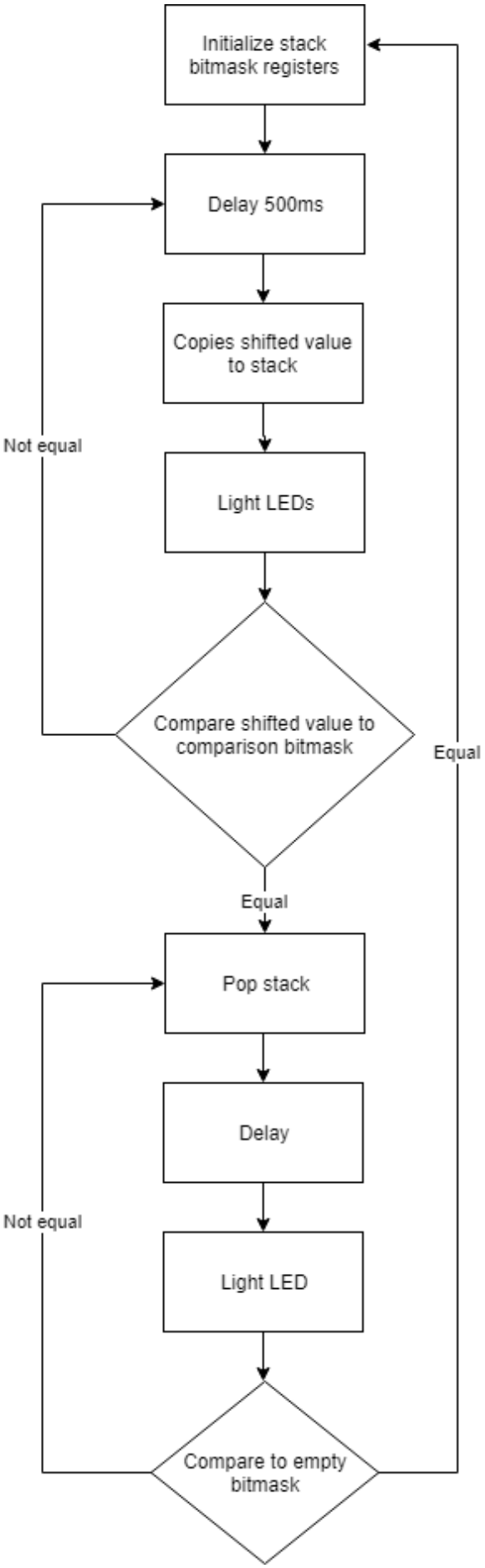
Task 5:

```
;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
; 1DT301, Computer Technology I
; Date: 2019-09-13
; Author:
; Christoffer Lundström
; Marcus Johansson
;
; Lab number: 1
; Title: Task 5.
;
; Hardware: Arduino UNO Rev 3, CPU ATmega328P
;
; Function: This program lights 6 LEDs in using 500ms delay (ring counter).
;
; Input ports: No input in this task.
;
; Output ports: PORTB used to light leds.
;
; Subroutines: init, main, compare, delay
;
; Other information:
;
; Changes in program: 13/9 2019
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
ldi r20, HIGH(RAMEND) ; high part of ram
out SPH,r20
ldi r20, LOW(RAMEND) ; low part of ram
out SPL, r20
;;;;;;
ldi r16, 0xFF; load 11 to r16
out DDRB, r16; set output on

init:
    ldi r16, 0b0000_0001
    ldi r17, 0b0100_0000; RESET BIT

main:
    out PORTB, r16; light led 1
    rcall delay
    lsl r16 ; shift r16 one to the left and output
    rcall compare
    rjmp main

compare:
    cp r16,r17
    breq init ; if registers are equal branch
    reti

delay:; delay routine
    ldi  r21, 41 ; load temp value1
    ldi  r22, 150; load temp value2
    ldi  r23, 128; load temp value3
    L1:
    dec  r23
    brne L1
    dec  r22
    brne L1
    dec  r21
    brne L1
    reti
```


Initialize Stack
Set output PORT
Load bitmasks
Light LED & shift bit left
Delay 500ms
Compare bitmasks
Equal
Not equal
Jump

Task 6:

```
; 1DT301, Computer Technology I
; Date: 2019-09-13
; Author:
; Christoffer Lundström
; Marcus Johansson
;
; Lab number: 1
; Title: Task 6.
;
; Hardware: Arduino UNO Rev 3, CPU ATmega328P
;
; Function: This program lights LED 1-6 with 500ms delay and turns off leds in reverse order.
;
; Input ports: No input in this task.
;
; Output ports: PORTB used to light leds.
;
; Subroutines: init, main_loop, reset and delay
;
; Other information:
; init: Responsible for setting up stack, data directions and initial register values.
; main loop: Shifts bits, adds 0x01 to shifted bit and pushes to stack before outputting to PORTB.
; reset: Reverses actions made by main loop by popping stack and outputting values.
; delay: 500ms delay.
; Changes in program: 13/9 2019
;
;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

init:
    ldi r20, high(ramend)
    out sph, r20
    ldi r20, low(ramend)
    out spl, r20
    ldi r16, 0xFF
    out DDRB, r16 ; Set to output
    ldi r17, 0b0011_1111 ; Full bit
    ldi r18, 0x00
    ldi r19, 0x01
    ldi r21, 0x00
    push r21

main_loop:
    rcall delay
    lsl r18 ; shift 18 left
    add r18, r19; add 19 to 18
    push r18 ;copy to stack
    out PORTB, r18
    cp r18, r17
    breq reset
    rjmp main_loop

reset:
    pop r18
    rcall delay
    out PORTB, r18
    cp r18, r21
    breq init
    rjmp reset


delay:; delay routine
    ldi  r21, 41 ; load temp value1
    ldi  r22, 150; load temp value2
    ldi  r23, 128; load temp value3
    L1:
    dec  r23
    brne L1
    dec  r22
    brne L1
    dec  r21
    brne L1
    reti
```


Initialize stack bitmask registers
Delay 500ms
Copies shifted value to stack
Light LEDs
Compare shifted value to comparison bitmask
Not equal
Equal
Equal
Pop stack
Delay
Light LED
Compare to empty bitmask
Not equal