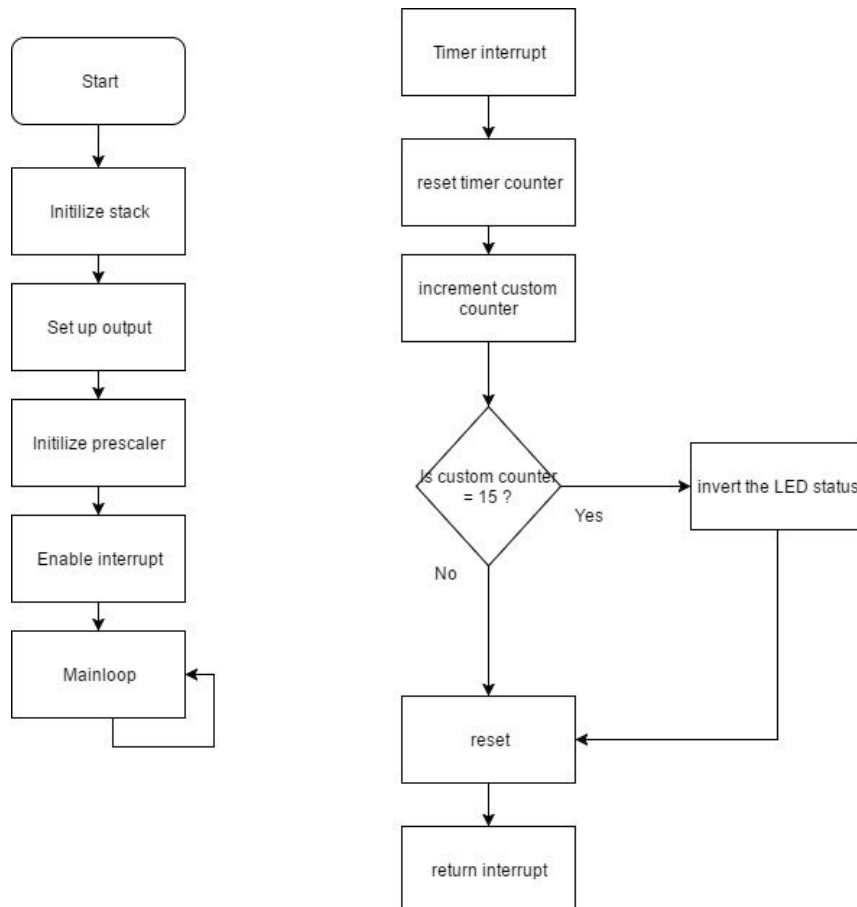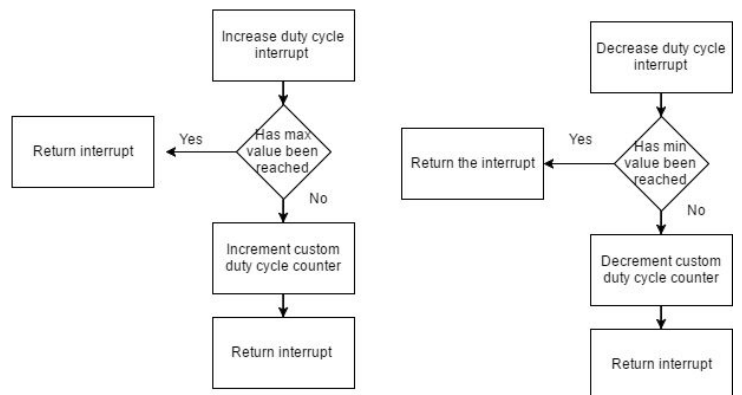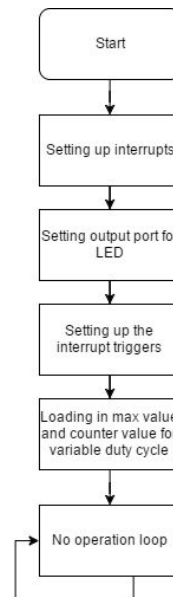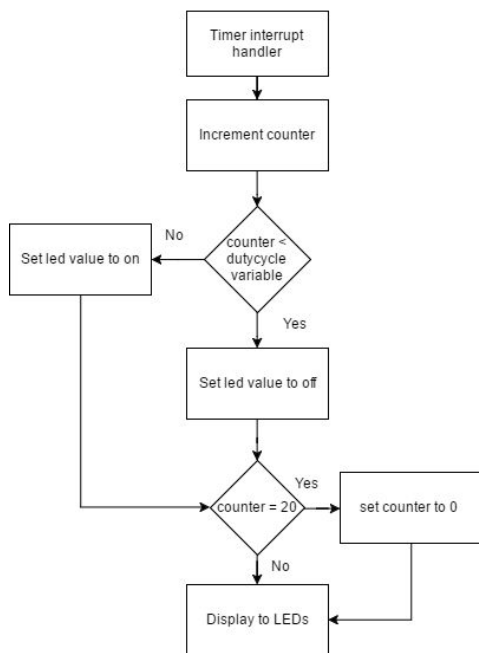# Linnæus University

Lab 4

Matus Maruna & John Charo

Task 1

In Task 1 we set up a timer and a counter in the timer interrupt to reach 0.5 second LED cycle where the LED would be turned on or off only if the counter was satisfied.

Task 2

Task 2 is similar to task 1 except a custom duty cycle counter was made that is increased or decreased with button presses using interrupts. The result is a variable duty cycle that is increased or decreased by 5%

```
Timer interrupt
    handler
        |
        v
Increment counter
        |
        v
   counter <          No
   dutycycle    ----------> Set led value to on
   variable
        |
       Yes
        |
        v
Set led value to off
        |
        v
                    Yes
   counter = 20  --------> set counter to 0
        |
        No
        |
        v
  Display to LEDs  <------
```

```
        Start
          |
          v
Setting up interrupts
          |
          v
Setting output port for
        LED
          |
          v
Setting up the
interrupt triggers
          |
          v
Loading in max value
and counter value for
variable duty cycle
          |
          v
No operation loop  <--+
          |           |
          +-----------+
```

```
Increase duty cycle
     interrupt
          |
          v
   Has max          Yes
   value been  --------> Return interrupt
   reached
          |
         No
          |
          v
Increment custom
duty cycle counter
          |
          v
Return interrupt
```

```
Decrease duty cycle
     interrupt
          |
          v
   Has min          Yes
   value been  --------> Return the interrupt
   reached
          |
         No
          |
          v
Decrement custom
duty cycle counter
          |
          v
Return interrupt
```

Task 3

In task 3 we had connected the board to the computer using the serial port and run putty at baud rate of 4800 corresponding to the table provided in the lectures. The code would constantly listen for any incoming characters. When character was recieved the code would branch and output the character to LEDs, afterwards returning to main loop.

```
                    ┌─────────────┐
                    │    Start    │
                    └─────────────┘
                           │
                    ┌─────────────┐
                    │ Setting output │
                    └─────────────┘
                           │
                    ┌─────────────┐
                    │ Enable flags in UCR │
                    │  for TXEN1 and  │
                    │     RXEN1       │
                    └─────────────┘
                           │
                           ▼          No
                    ◇ Incoming ◇ ──────
                    ◇  char ?  ◇
             Yes           │
                    ┌─────────────┐
                    │ Save to char │
                    └─────────────┘
                           │
                    ┌─────────────┐
                    │ com char and output │
                    │    to LEDs      │
                    └─────────────┘
```
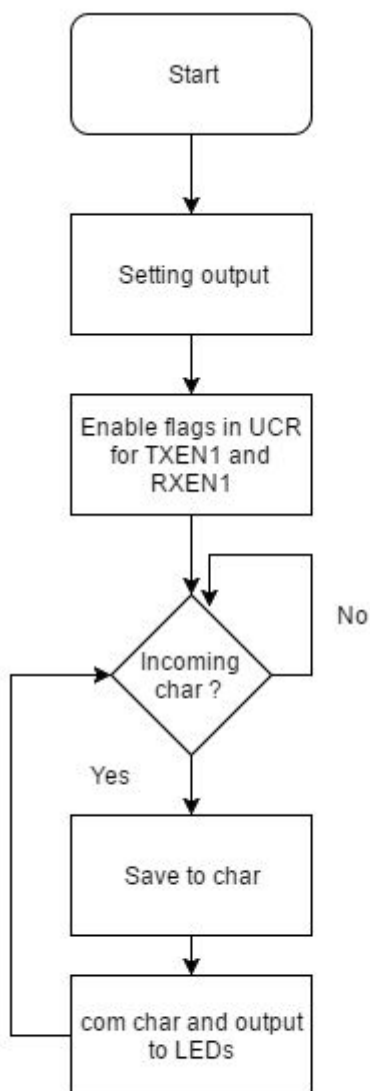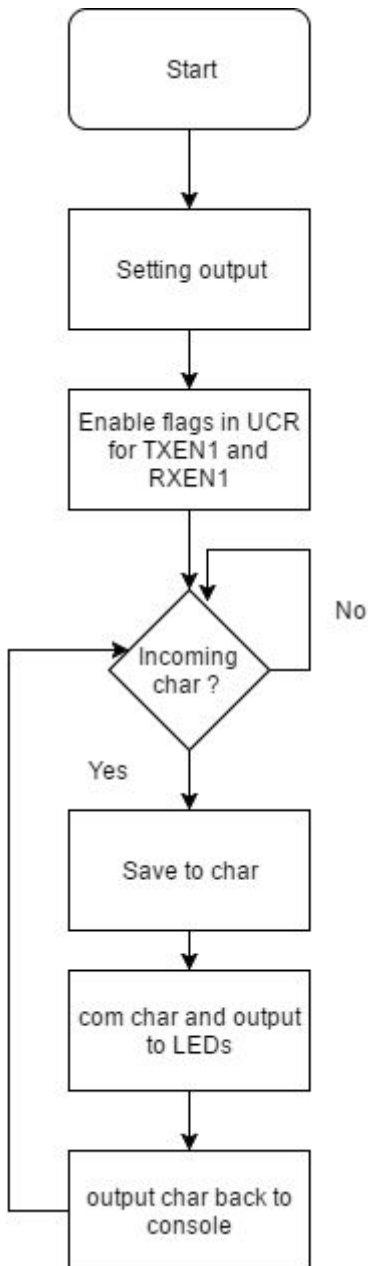
Task 4

Task 4 is similar to task 3 with the main difference being that after the character was output the character was sent back to the console for the user (us) to see.

Task 5

Instead of having to constantly listen for a character like in task 4 we had used and interrupt that would then retrieve the character, send it to the LEDs and send it back to the console.

```
                    ┌─────────────┐
                    │    Start    │
                    └─────────────┘
                           │
                           ▼
                    ┌─────────────┐
                    │ Setting up  │
                    │    the      │
                    │  interrupts │
                    └─────────────┘
                           │
                           ▼
                    ┌─────────────┐
                    │ Interrupts  │
                    │   enable    │
                    └─────────────┘
                           │
                           ▼
             ┌──────▶┌─────────────┐
             │       │    Loop     │
             │       └─────────────┘
             │              │
             │              ▼
             │           ◇ Has the
             │      No   interrupt
             └──────── been triggered ◇
                            │
                           Yes
                            │
                            ▼
                    ┌─────────────┐
                    │  Read and   │
                    │    store    │
                    │  character  │
                    └─────────────┘
                           │
                           ▼
                    ┌─────────────┐
                    │ Output the  │
                    │ character   │
                    │  to LEDs    │
                    └─────────────┘
                           │
                           ▼
                    ┌─────────────┐
                    │ Send the    │
                    │ character   │
                    │ back to     │
                    │ console     │
                    └─────────────┘
```