**Drivers**

Liquid Crystal Display

# Specifications:

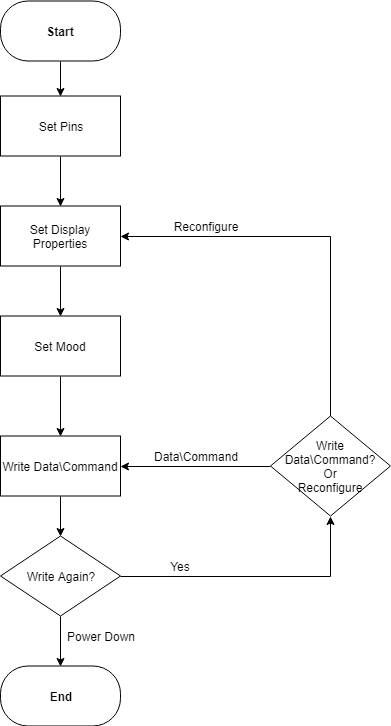
*A Simple driver for most common LCDs with built-in Hitachi HD44780 LCD controller*

We should first explore what we want this driver to do, it should be simple, but not too simple. I don’t want it to deviate too much from the original ***HD44780***chip command set.

Another thing to but in consideration is that this driver will be made for single microcontroller ***ATMEL328p***  but it will also be ease to modify to work with other microcontrollers.

1. Write characters to display.
2. Move cursor.
3. Control contrast.
4. News feed mood.
5. Shift to right\left mood.
6. Fixed cursor mood.

# Design

Let’s graph a flow chart to see how to we should implement the driver.

The best way to go about implementing this driver is through C++ Classes. The main reason for using classes is (in the rare case) if the user want to use more than one LCD.

The Class:

Let’s name the class “LCD”, first we need to set the pins. The logical choice for this task is the constructor seeing that this is the very first thing we need to set if we want to “talk” to any device.

The constructors will take the microcontroller Pins that are connected to the LCD so that the methods within the class will know how to send data\command to the correct pins in the chosen port.

Class LCD{

Constructor(Port,D7,D6,D5,D4,D3,D2,D1,D0,Port2,RS,RW,E);

Constructor(Port,D7,D6,D5,D4,D3,D2,D1,D0, Port2,RS,E);

Constructor(Port,D7,D6,D5,D4,RS,RW,E);

Constructor(Port,D7,D6,D5,D4,RS,E);

}

The second step is to set the display properties before sending any data\command. The properties will be the number of lines “**Lines**” and the “**Resolution**” which defines the organization of dots in the display characters of the LCD “5 \* 7 Dots” or “5 \* 10 Dots”. The “**C**” and “**B**” represent **C**ursor on\off and **B**link on\off respectively, they choose whether the user wants the cursor to be shown or not and if he\she wants current position to keep blinking or not.

the user will choose a predefined a macro for all parameters, or can define a new one if he\she wishes so.

Void init(Lines, C, B, Resolution);

Void init(Lines, C, B);

Void init(Lines);

Void print