# Opdracht 5.2

ET Phone Home

Deze opdracht bestaat uit twee kleinere opdrachten:

1. Elke IMTHE-Box bevat unieke componenten. Zoek naar de datasheets van deze componenten en maak een kort verslag per component over de protocollen die gebruikt worden en hoe je dit moet gebruiken vanuit de AVR code.
2. **Zoek naar een library op internet om het HD44780 display te kunnen besturen en maak een programma dat jouw naam en studentnummer op het display laat zien.**

## Aanpak en Uitvoering

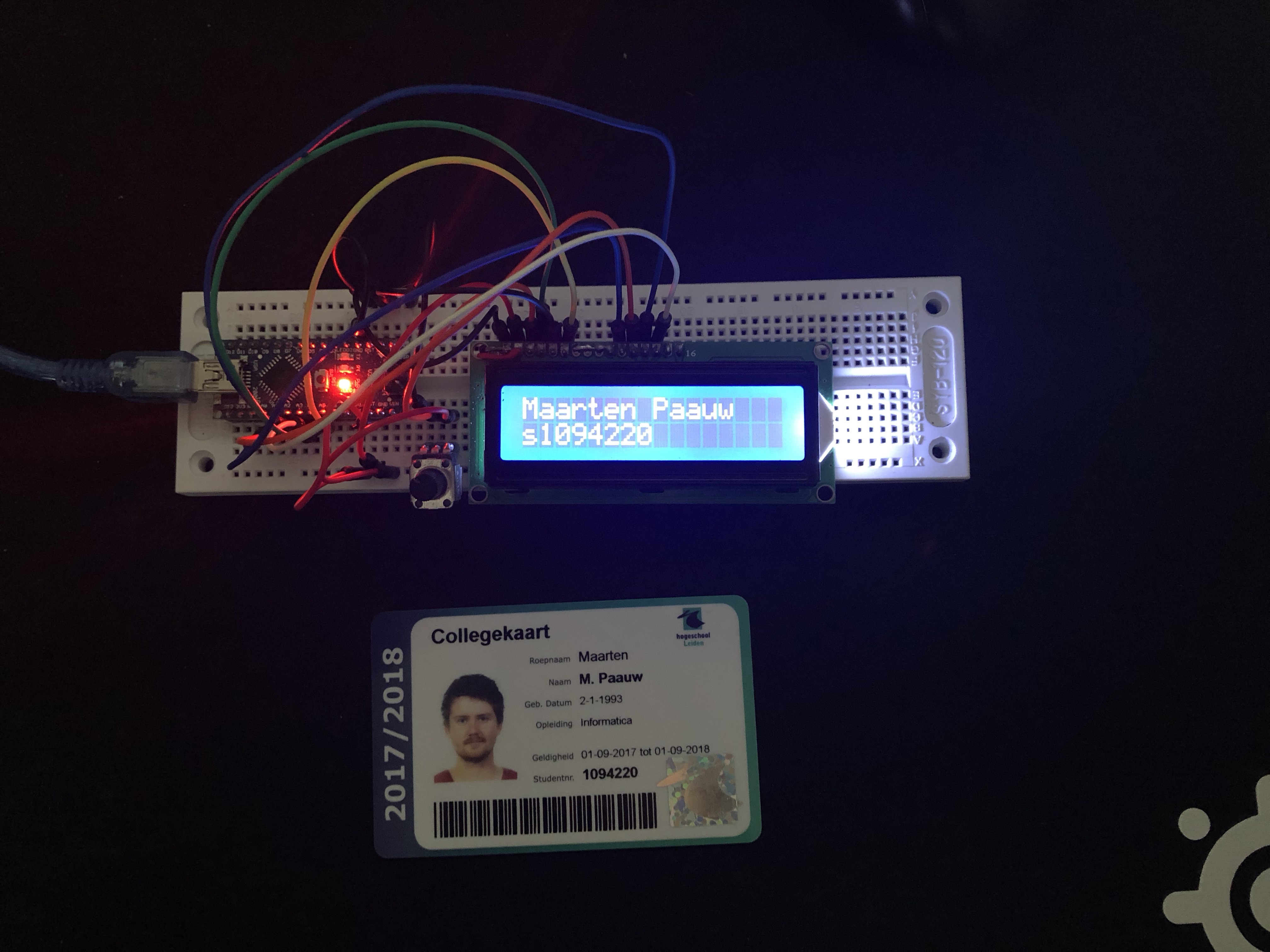
Om het display werkend te krijgen mocht ik gebruik maken van een library. Na een tijdje Googelen kwam ik terrecht bij een post op het forum van Atmel. De post ging over een library die gemaakt was voor dit display. De library had ook een settings bestand. Hier moesten aan aantal variabelen aangepast worden. Dit waren pin nummers, CPU snelheid en het aantal bits. Dit instellingen zijn opgenomen in het hoofdstuk code. Via een andere blogpost kwam ik erachter hoe ik de pinnen moest aansluiten en dat er een potentiometer nodig was.

De code zelf is vrij kort. Er zijn maar een aantal functies die aangeroepen hoeven worden:

* Een functie die het display initialiseert.
* Een functie die het display leeg maakt.
* Een functie die mijn naam op de eerste regel zet.
* Een functie die er voor zorgt dat er geschreven kan worden op de tweede regel.
* Een functie die mijn studentnummer op de tweede regel zet.

Al deze code staat buiten de loop omdat het maar één keer op het scherm gezet hoeft te worden. Deze code kan terug gevonden worden in het hoofdstuk code.

## Afbeelding



Opdracht 5.2 - ET Phone Home - Afbeelding

De afbeelding van de setup kan ook gedownload worden via de volgende link:

<https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O5_2/assets/setup.jpg>

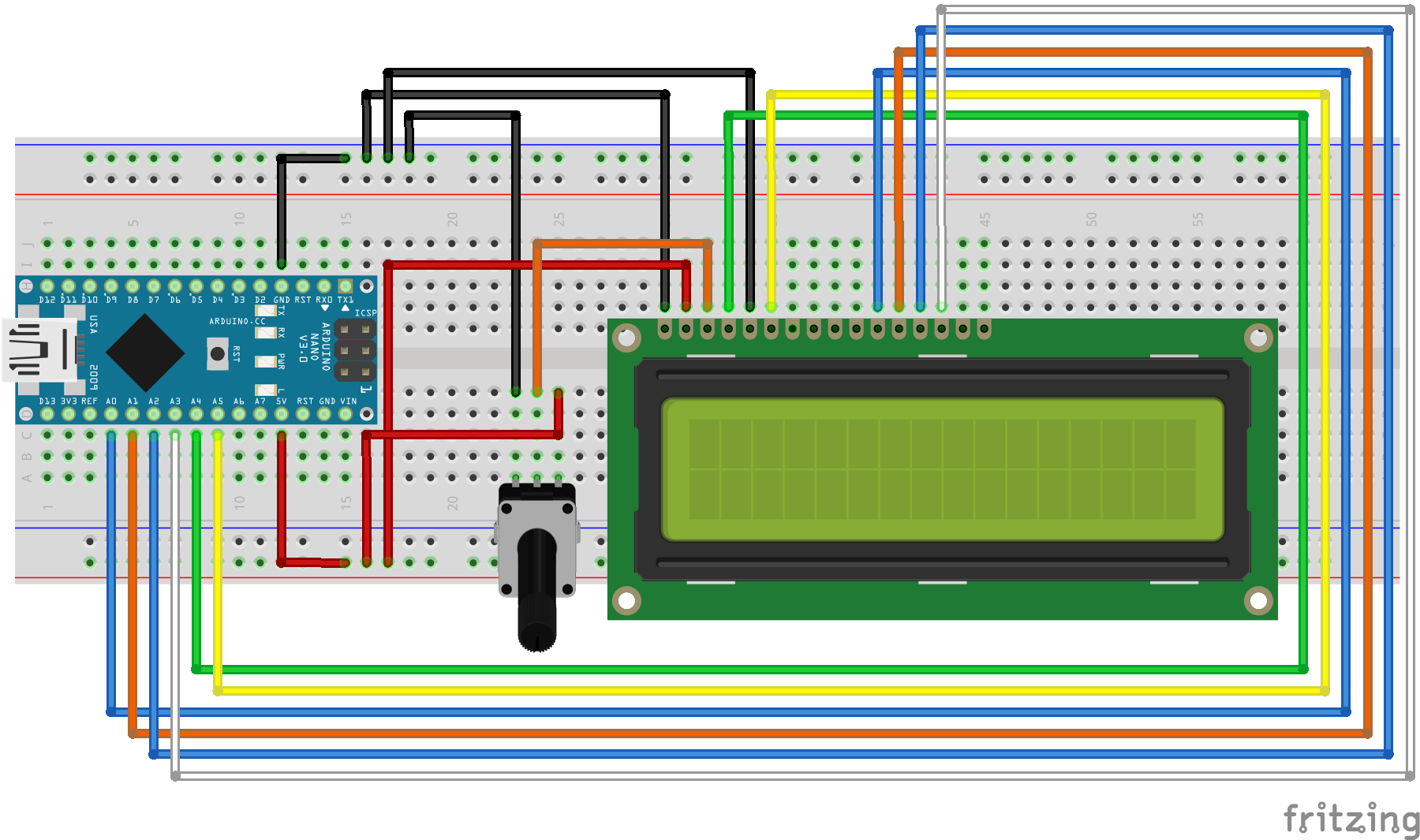
## Video

[](https://www.youtube.com/watch?v=F5fXDvtFF8w)

Deze video is ook te vinden op **Youtube**:

<https://www.youtube.com/watch?v=F5fXDvtFF8w>

## Breadboard Schema

[](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O5_2/assets/fritzing/schema.png)

Het **Fritzing** schema kan ook gedownload worden via de volgende link:

<https://github.com/maartenpaauw/IMTHE1/raw/master/O5_2/assets/fritzing/schema.fzz>

### Hardware

|  |
| --- |
| Onderdelen |
| Arduino Nano (v3.0) - 1× |
| LCD Scherm - 1× |
| Potentiometer - 1× |
| Dupont Kabel - 14× |

## Code

/\*  
 \* Opdracht Opdracht 5.2 - ET Phone Home  
 \*   
 \* Eigenaar:  
 \* Maarten Paauw <s1094220@student.hsleiden.nl>  
 \* s1094220  
 \* INF3C  
 \*   
 \* Versie: 1  
 \* Aangemaakt: 12 maart 2018  
 \* Gewijzigd: 29 maart 2018  
 \*/  
  
#include "hd44780.h"  
  
// Mijn naam.  
#define NAAM "Maarten Paauw"  
  
// Mijn studentnummer.  
#define STUDENTNUMMER "s1094220"  
  
// De main functie.  
int main(void)  
{  
 // Initialiseer het LCD display.  
 lcd\_init();  
  
 // Maak het LCD display leeg.  
 lcd\_clrscr();  
   
 // Schrijf mijn naam op de eerste regel.  
 lcd\_puts(NAAM);  
  
 // Ga naar de tweede regel.  
 lcd\_goto(0x40);  
  
 // Schrijf mijn studentnummer op de tweede regel.  
 lcd\_puts(STUDENTNUMMER);  
  
 // Loop voor altijd.  
 while (1) { }  
  
 // Geef 0 terug.  
 return 0;  
}

De code kan ook gevonden worden in mijn **GitHub** *repository* via de volgende link:

<https://github.com/maartenpaauw/IMTHE1/blob/master/O5_2/src/main.c>

### Library

De library die ik gebruikt heb heeft ook een aantal instellingen nodig. De instellingen heb ik hieronder opgenomen.

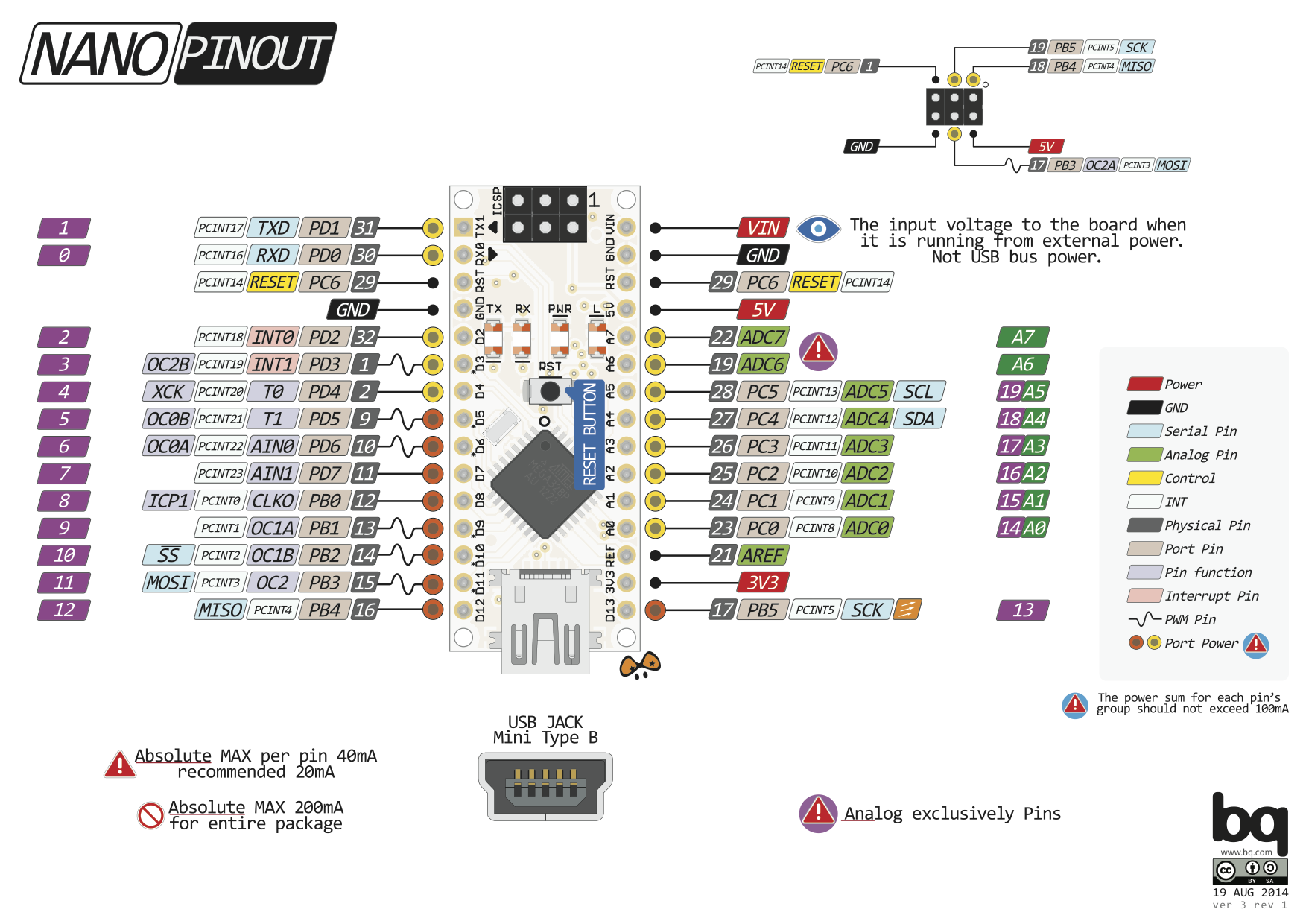
#ifndef HD44780\_SETTINGS\_H  
  
 #define HD44780\_SETTINGS\_H  
  
 #define F\_CPU 16000000 // Set Clock Frequency  
  
 #define USE\_ADELAY\_LIBRARY 0 // Set to 1 to use my ADELAY library, 0 to use internal delay functions  
 #define LCD\_BITS 4 // 4 for 4 Bit I/O Mode, 8 for 8 Bit I/O Mode  
 #define RW\_LINE\_IMPLEMENTED 0 // 0 for no RW line (RW on LCD tied to ground), 1 for RW line present  
 #define WAIT\_MODE 0 // 0=Use Delay Method (Faster if running <10Mhz)  
 // 1=Use Check Busy Flag (Faster if running >10Mhz) \*\*\*Requires RW Line\*\*\*  
 #define DELAY\_RESET 15 // in mS  
  
 #if (LCD\_BITS==8) // If using 8 bit mode, you must configure DB0-DB7  
 #define LCD\_DB0\_PORT PORTC  
 #define LCD\_DB0\_PIN 0  
 #define LCD\_DB1\_PORT PORTC  
 #define LCD\_DB1\_PIN 1  
 #define LCD\_DB2\_PORT PORTC  
 #define LCD\_DB2\_PIN 2  
 #define LCD\_DB3\_PORT PORTC  
 #define LCD\_DB3\_PIN 3  
 #endif  
  
 #define LCD\_DB4\_PORT PORTC // If using 4 bit omde, yo umust configure DB4-DB7  
 #define LCD\_DB4\_PIN 0  
 #define LCD\_DB5\_PORT PORTC  
 #define LCD\_DB5\_PIN 1  
 #define LCD\_DB6\_PORT PORTC  
 #define LCD\_DB6\_PIN 2  
 #define LCD\_DB7\_PORT PORTC  
 #define LCD\_DB7\_PIN 3  
  
 #define LCD\_RS\_PORT PORTC // Port for RS line  
 #define LCD\_RS\_PIN 4 // Pin for RS line  
  
 #define LCD\_RW\_PORT PORTC // Port for RW line (ONLY used if RW\_LINE\_IMPLEMENTED=1)  
 #define LCD\_RW\_PIN 6 // Pin for RW line (ONLY used if RW\_LINE\_IMPLEMENTED=1)  
  
 #define LCD\_DISPLAYS 1 // Up to 4 LCD displays can be used at one time  
 // All pins are shared between displays except for the E  
 // pin which each display will have its own  
  
 // Display 1 Settings - if you only have 1 display, YOU MUST SET THESE  
 #define LCD\_DISPLAY\_LINES 2 // Number of Lines, Only Used for Set I/O Mode Command  
 #define LCD\_E\_PORT PORTC // Port for E line  
 #define LCD\_E\_PIN 5 // Pin for E line  
  
 #if (LCD\_DISPLAYS>=2) // If you have 2 displays, set these and change LCD\_DISPLAYS=2  
 #define LCD\_DISPLAY2\_LINES 2 // Number of Lines, Only Used for Set I/O Mode Command  
 #define LCD\_E2\_PORT PORTC // Port for E line  
 #define LCD\_E2\_PIN 5 // Pin for E line  
 #endif  
  
 #if (LCD\_DISPLAYS>=3) // If you have 3 displays, set these and change LCD\_DISPLAYS=3  
 #define LCD\_DISPLAY3\_LINES 2 // Number of Lines, Only Used for Set I/O Mode Command  
 #define LCD\_E3\_PORT PORTC // Port for E line  
 #define LCD\_E3\_PIN 5 // Pin for E line  
 #endif  
  
 #if (LCD\_DISPLAYS>=4) // If you have 4 displays, set these and change LCD\_DISPLAYS=4  
 #define LCD\_DISPLAY4\_LINES 2 // Number of Lines, Only Used for Set I/O Mode Command  
 #define LCD\_E4\_PORT PORTC // Port for E line  
 #define LCD\_E4\_PIN 5 // Pin for E line  
 #endif  
  
#endif

De code kan ook gevonden worden in mijn **GitHub** *repository* via de volgende link:

<https://github.com/maartenpaauw/IMTHE1/blob/master/O5_2/lib/hd44780/hd44780_settings.h>

## Datasheets

### Arduino Nano

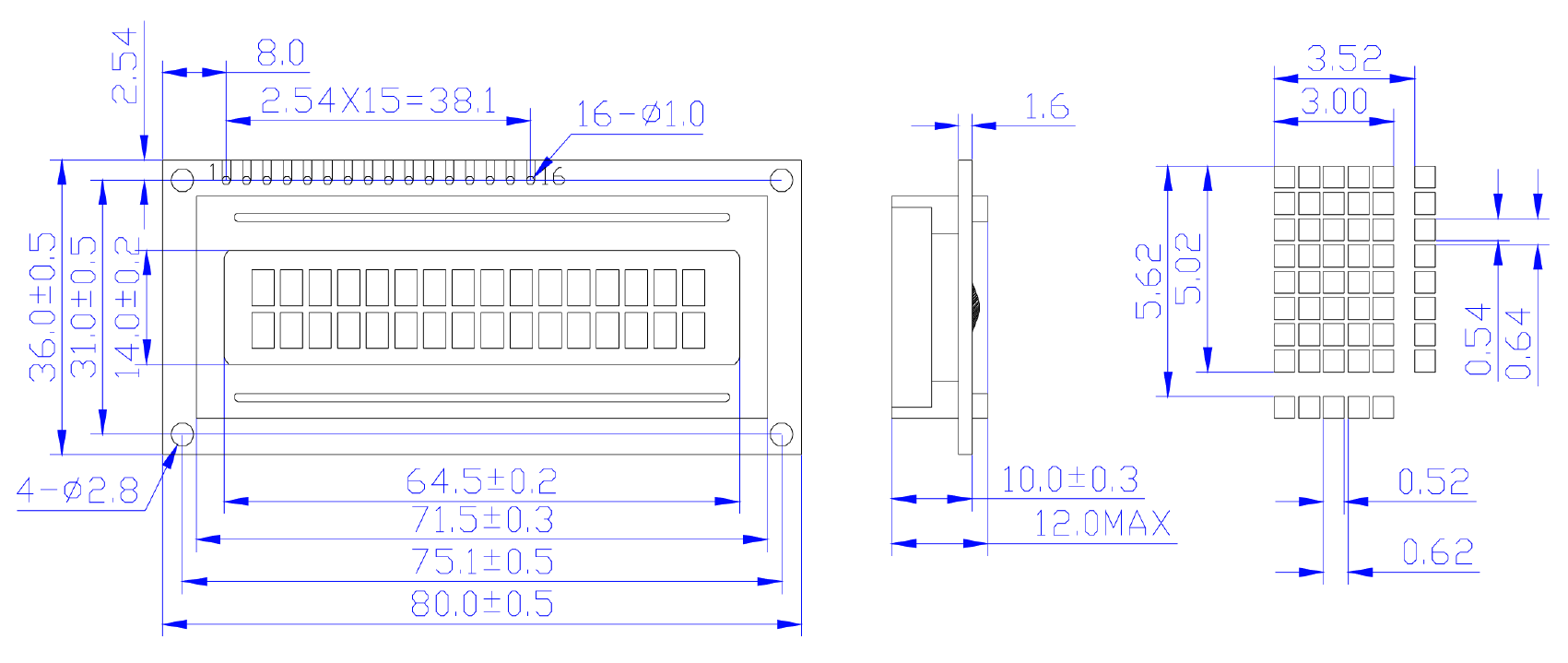


Arduino Nano Pinout

De [Arduino Nano Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O5_2/assets/data_sheets/nano.png) is uitgedeeld tijdens de eerste les en terug te vinden via het volgende forum post:

<https://forum.arduino.cc/index.php?topic=147582.0>

### HD44780

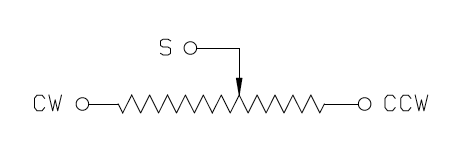


HD44780

De [HD44780 Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O5_2/assets/data_sheets/hd44780.png) heb ik gevonden in de datasheet van het displat zelf. Deze datasheet kan terug gevonden worden via de volgende link:

<https://www.electronicshub.org/wp-content/uploads/2013/09/HD44780_16x2_Character_LCD_Display.pdf>

### Potentiometer



Potentiometer Pinout

De [Potentiometer Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O5_2/assets/data_sheets/potentiometer.png) heb ik gevonden in de datasheet van de Potentiometer. Deze datasheet is terug te vinden via de volgende link:

<https://components101.com/sites/default/files/component_datasheet/potentiometer%20datasheet.pdf>

## Bronvermelding

* <https://forum.arduino.cc/index.php?topic=147582.0> (Arduino Nano Pinout)
* <https://community.atmel.com/projects/hd44780-library> (HD44780 Library)
* <https://www.electronicshub.org/wp-content/uploads/2013/09/HD44780_16x2_Character_LCD_Display.pdf> (HD44780 Datasheet)
* [https://cdn-learn.adafruit.com/assets/assets/000/027/586/original/light\_control-pot.png](https://components101.com/sites/default/files/component_datasheet/potentiometer%20datasheet.pdf) (Potentiometer Datasheet)