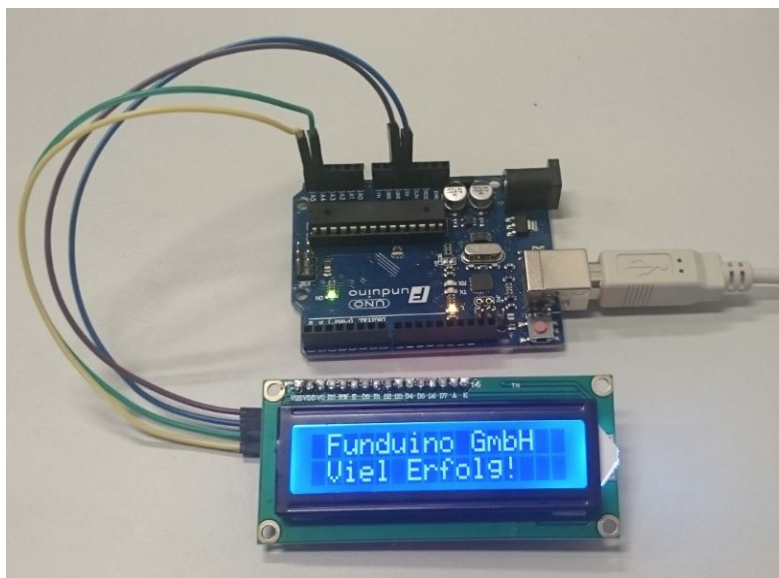


I²C Display

Anleitung zum LCD Display mit I²C Anschluss

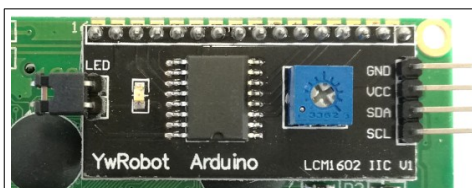
Mit einem LCD-Display kann man Buchstaben und Ziffern darstellen. Dies ist in vielen Anwendungen nützlich, zum Beispiel um Messwerte oder auch Menüs darzustellen. Mit Hilfe des LCD lassen sich aber auch Daten darstellen, wenn kein Computer am Arduino-Mikrocontroller angeschlossen ist.

Das LCD Modul mit angelötetem I²C Bus ermöglicht die Verwendung eines LCD Moduls mit einer einfachen Verkabelung. Dies ist bei komplexeren Projekten besonders vorteilhaft. Ein weiterer Unterschied zum normalen LCD Display besteht darin, dass sich auf der Rückseite des Displays ein Drehregler befindet, mit dem die Leuchtstärke des LCD reguliert werden kann.

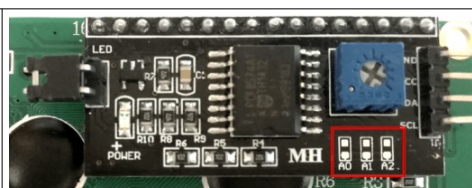


Hinweis:

Es gibt je nach Display unterschiedliche I²C-Module, die auf der Rückseite des LCD angelötet sind. Diese Anleitung funktioniert grundsätzlich mit verschiedenen I²C-Displays, jedoch kann die I²C-Adresse (eine HEX-Zahl) unterschiedlich sein, und muss im Sketch entsprechend angegeben werden. Die typischen I²C-HEX Adressen sind „0x27“ und „0x3F“. Sollte das Display während der Durchführung dieser Anleitung nicht funktionieren, muss eine andere HEX-Adresse verwendet werden.



I²C-Modul ohne Lötstellen mit fester HEX-Adresse 0x27

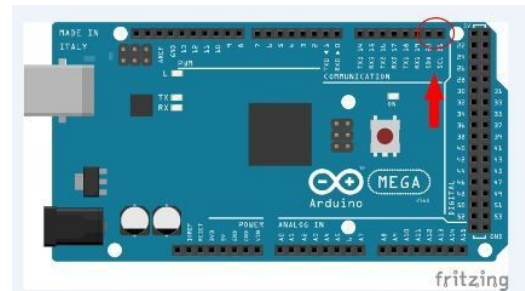


I²C-Modul mit rot markierten Lötstellen. Standard-HEX-Adresse 0x3F, veränderbar durch die Lötstellen A1-A3)

Material: Mikrocontroller (in diesem Beispiel UNO R3), LCD mit I2C Modul, Kabel

Verkabelung: Die Verkabelung ist sehr simpel. Am I2C LCD Modul sind nur vier Kontakte vorhanden. GND wird mit dem GND Kontakt am Mikrocontroller verbunden. VCC mit dem 5V Kontakt am Microcontroller, SDA mit dem analogen Eingang A4 und SCL mit dem analogen Eingang A5.

Achtung! Bei dem MEGA2560 R3 Microcontroller gibt es für die SDA – und SCL- Kontakte eigene Eingänge auf dem Board unter 20 und 21.



Programmieren:

Um mit dem I²C LCD Modul zu arbeiten, benötigt man eine Library welche noch nicht im Arduino Programm vorinstalliert ist. Wir verwenden in dieser Anleitung die „*Arduino-LiquidCrystal-I2C-library-master*“ Library. Die Library kann über die Bibliothekenverwaltung der Arduino-Software hinzugefügt werden. Als Suchbegriff direkt den Namen „*Arduino-LiquidCrystal-I2C-library-master*“ eingeben. Eine detaillierte Beschreibung, wie Bibliotheken eingefügt werden findet man im Theorieteil zu dieser Anleitung im Unterpunkt „Bibliotheken zur Arduino Software hinzufügen“.

Code:

```
#include <Wire.h> // Wire Bibliothek einbinden
//Vorher hinzugefügte LiquidCrystal_I2C Bibliothek einbinden
#include <LiquidCrystal_I2C.h>

//Hier wird festgelegt um was für einen Display es sich handelt. In diesem Fall
//einer mit 16 Zeichen in 2 Zeilen. Für ein vierzeiliges I2C-LCD verwendet man
//den Code "LiquidCrystal_I2C lcd(0x27, 16, 4)"
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
    lcd.begin(); //Im Setup wird der LCD gestartet
    //Hintergrundbeleuchtung einschalten (0 schaltet die
    //Beleuchtung aus).
    lcd.setBacklight(1);
}

void loop()
{
    //Hier wird die Position des ersten Zeichens
    //festgelegt. In diesem Fall bedeutet
    //(0,0) das erste Zeichen in der ersten Zeile.
    lcd.setCursor(0, 0);
    lcd.print("Hallo Welt!");
    // In diesem Fall bedeutet (0,1) das erste Zeichen
    //in der zweiten Zeile.
    lcd.setCursor(0, 1);
    lcd.print("Viel Erfolg!");
}
```

Anwendungsbeispiel:

Mit dem I2C LCD Modul können wie mit dem einfachen LCD Modul, auch Messwerte angezeigt werden.

Hier ein Beispielcode, bei dem ein Feuchtigkeitssensor an Pin A0 angeschlossen wurde :

```
#include <Wire.h> // Wire Bibliothek hochladen
// Vorher hinzugefügte LiquidCrystal_I2C Bibliothek hochladen
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16,2);

int messwert=0;

void setup()
{
    lcd.begin(); //Im Setup wird der LCD gestartet
    lcd.setBacklight(1);
}

void loop()
{
    // Der Messwert vom Analogen Eingang A0 soll ausgelesen,
    //und unter der Variablen „messwert“ gespeichert werden.
    messwert=analogRead(A0);
    // In der ersten Zeile soll der Text „Messwert:“ angezeigt werden.
    lcd.setCursor(0, 0);
    lcd.print("Messwert:");
    // In der zweiten Zeile soll der Messwert, der vom
    Feuchtigkeitssensor bestimmt wurde, angezeigt werden.
    lcd.setCursor(0, 1);
    lcd.print(messwert);
    delay(500); // Die Pause von 500ms ist die Zeit zwischen den Messungen.
}
```