



---

# **BACHELORARBEIT**

---

Herrr  
**Jian, Li**

**Entwicklung und Umsetzung eines  
Systems zur Bewertung von  
Zündkerzen von Kraftfahrzeugen**

Mittweida, 2022



# BACHELORARBEIT

---

## Entwicklung und Umsetzung eines Systems zur Bewertung von Zündkerzen von Kraftfahrzeugen

Autor:  
Herr  
**Jian, Li**

Studiengang:  
**Elektro- und Informationstechnik**

Seminargruppe:  
**EI18sA-BC**

Erstprüfer:  
**Prof. Dr. -Ing. Michael Kuhl**

Zweitprüfer:  
**M. Sc. Kevin**

Einreichung:  
**Mittweida, 13.01.2022**

Verteidigung/  
Bewertung: **Mittweida,**  
**2022**

# BACHELORTHESIS

---

**Development and implementation of a  
system for evaluating spark plugs in  
motor vehicles**

author:

Herr **Jian, Li**

Course of studies:

**Electro- and Informationtechnology**

seminar group:

**EI18sA-BC**

first examiner:

**Prof. Dr. -Ing. Michael Kuhl**

second examiner:

**M. Sc. Kevin**

submission:

**Mittweida, 13.01.2022**

defence/

evaluation : **Mittweida,  
2022**

# Bibliografische Angaben

Jian, Li

Entwicklung und Umsetzung eines Systems zur Bewertung von Zündkerzen von Kraftfahrzeugen-2022

Development and implementation of a system for evaluating spark plugs in motor vehicles-2022

Seiten 39, Mittweida, Hochschule Mittweida, Fakultät Ingenieurwissenschaften, Bachelorarbeit-2022

## Referat:

Die Bachelorarbeit entstand in der Zeit der Corona-Pandemie, hauptsächlich im Home-Office.

Ziel der Arbeit ist die prototypische Aufbau eines Systems zur Bewertung von Zündkerzen über eine vergleichende Messung der Zündfunken (Zündkerze im Vergleich zu einer unbekannten Zündkerze). Hierzu sollen ein automatischer Testzyklus gestartet werden, die Intensität der Funken bestimmt und über einen vorgegebenen Mikrocontroller bewertet und ausgegeben werden. Die Werte sollen ebenfalls als Reihen mit entsprechend sinnvollen Angaben an einen PC übergeben, abgespeichert und ausgegeben werden.

# Inhaltverzeichnis

Inhaltverzeichnis .....	i
Abbildungsverzeichnis .....	iii
Tabellenverzeichnis .....	v
Abkürzungsverzeichnis .....	vi
1 Einleitung .....	1
2 Aufgabenstellung.....	1
3 These .....	2
4 Stand der Technik.....	2
4.1 Helligkeitssensor .....	2
4.1.1 Innerer photoelektrischer Effekt.....	2
Photoleitung.....	2
4.1.2 Photowiderstand GL5506 .....	3
5 Umsetzung.....	5
5.1 Gesamtkonzept .....	5
5.2 Hardware .....	8
5.2.1 Anforderungen.....	8
5.2.2 Systementwurf.....	12
5.3 Software.....	13
5.3.1 Softwarekonzept.....	13
5.3.2 Erklärung des entwickelten Codes .....	16
6 Inbetriebnahme / Systemtest.....	18
6.1 Aufbau von Hard- und Software .....	18
6.2 Test .....	18
7 Versuchsdurchführung .....	19
7.1 Versuchsaufbau .....	19

7.2 Helligkeitsmessverfahren .....	20
7.2.1 Daten in einem Excel-Arbeitsblatt speichern .....	20
7.2.2 Test .....	21
8 Zusammenfassung .....	27
9 Ausblick .....	28
Literatur .....	29
Anlagen .....	30
Eigenständigkeitserklärung .....	39

# Abbildungsverzeichnis

Abbildung 1:Photoelektrischer Effekt .....	2
Abbildung 2:Bewegung von Elektronen .....	3
Abbildung 3:Photowiderstand GL5506[3] .....	4
Abbildung 4:Flussdiagramm des Gesamtkonzepts.....	5
Abbildung 5:Messschaltung.....	5
Abbildung 6:analoges Spannungssignal im Pin A0 .....	7
Abbildung 7: Hardware Konzept.....	8
Abbildung 8:Arduino Mega 2560[4] .....	8
Abbildung 9:Beispiel von LCD 1602 und I2C/SPI LCD .....	10
Abbildung 10: DS 1302 Modul[6] .....	11
Abbildung 11:Block Diagramm von DS 1302[6].....	12
Abbildung 12:Schaltplan .....	13
Abbildung 13: Ablaufdiagramm für Messverfahren .....	14
Abbildung 14:Ablaufdiagramm für Zeitaufnahme.....	15
Abbildung 15: Programm von Umschaltung.....	16
Abbildung 16: Programm zum Aufzeichnen von Zeitpunkten.....	17
Abbildung 17: Programm von Messverfahren .....	17
Abbildung 18: Aufbau von System .....	19
Abbildung 19: Aufbau von System .....	19
Abbildung 20: PLX-DAQ[7].....	21
Abbildung 21:Spannungsverläufer der intakten LED (U1) und defekte LED (U2) beim Fall 1 .....	22
Abbildung 22:Anfangszeit und Laufzeit beim Fall 1 .....	22
Abbildung 23: Lebensdauer der bekannten Zündkerze beim Fall 1 .....	23
Abbildung 24: Spannungsverläufer der intakte LED (U1) und defekte LED (U2) beim Fall 2.....	23
Abbildung 25:Anfangszeit und Laufzeit beim Fall 2 .....	24



Abbildung 26: Lebensdauer der bekannten Zündkerze beim Fall 2.....	24
Abbildung 27:Spannungsverläufer der intakte LED (U1) und defekte LED (U2) beim Fall 3.....	25
Abbildung 28:Anfangszeit und Laufzeit beim Fall 3 .....	25
Abbildung 29:Abbildung 22: Lebensdauer der bekannten Zündkerze beim Fall 3.....	26
Abbildung 30: Schaltungsplan1 .....	30
Abbildung 31 : Schaltungsplan 2 .....	30
Abbildung 32: Schaltungsplan 3 .....	31
Abbildung 33: Abbildung 30: PCB(Vorderseite).....	32
Abbildung 34: Abbildung 31: PCB(Rückseite).....	32

# Tabellenverzeichnis

Tabelle 1: Elektrische Parameter von GL5506[3] .....	4
Tabelle 2 : Vorteile und Nachteile von GL5506 .....	4
Tabelle 3: Spezifikation von LCD 1602 [5] .....	10
Tabelle 4: Testdaten von Fall 1 .....	22
Tabelle 5: Testdaten von Fall 2 .....	23
Tabelle 6: Testdaten von Fall 3 .....	24

# Abkürzungsverzeichnis

EEPROM .....	Electrically Erasable Programmable Read-Only Memory
FMCW .....	Frequency-Modulated Continuous-Wave
I <sup>2</sup> C .....	Inter-integrated circuit
ICSP .....	In-Circuit Serial Programming
IDE .....	Integrated Development Environment
LCD .....	liquid crystal display
PWM .....	Pulse Width Modulation
RX .....	Receive
SCL .....	serial clock
SDA .....	serial data
SRAM .....	static random access memory
TX .....	Transmit
UART .....	Universal synchronous / asynchronous receiver / transmitter
VCC .....	Voltage of circuit

# 1 Einleitung

Stellen Sie sich bitte dieses Szenario vor: Sie haben Vormittag um 8 Uhr eine Sitzung und Sie drehen den Schlüssel um 7.30 Uhr, um ihr Auto zu starten, aber aufgrund der Qualität der Zündkerze konnten Sie es nicht zünden und Sie sind unvermeidlich zu spät. Mit diesem Projekt können Sie jedoch die Lebensdauer der Zündkerze vorhersagen, ob die Zündkerze im Auto morgen funktionieren wird und so peinliche Situation vermeiden.

## 2 Aufgabenstellung

Die Aufgabe dieses Projekts ist es, ein System zur Bewertung von unbekannten Zündkerzen aufzubauen. Die Qualität der Zündkerze ist ein wichtiger Bestandteil dafür, ob das Auto gestartet werden kann. Das Zündkerzenbewertungssystem dieses Projekts basiert auf dem Vergleich der Helligkeit einer bekannten funktionierenden Zündkerze und der Helligkeit einer unbekannten Zündkerze.

Dieses Zündkerzenbewertungssystem kann durch Vergleich der Helligkeit der bekannter Zündkerzen und der unbekannten Zündkerzen beurteilen, ob die unbekannte Zündkerzen noch für Autos geeignet sind, die Messdaten werden an den Computer übertragen und gleichzeitig gespeichert. Um die Intensität der Helligkeit einer unbekannten Zündkerze direkter anzuzeigen, gibt es unter verschiedenen Bedingungen unterschiedliche Farben der LEDs. Die genauere Informationen werden durch Umschaltung auf dem Display geliefert und in Echtzeit an Microsoft Excel übergeben. Aufgrund der elektromagnetischen Verträglichkeit von Zündkerze werden die echten Zündkerzen im Test durch 2 LEDs ersetzt. Das Experiment besteht aus vier Teile: Programmierung, Erstellung des Aufbaus, Test und Ausblick von ganzem System.

Als Hardware werden vier Module (Arduino Mega 2560, RTC DS1302, LCD1602, Messschaltung) verbunden. Zusätzlich braucht es noch die Programmiersoftware "Arduino". Durch die vier Module werden die Intensität der Helligkeit der Zündkerze in dunkler

Umgebung gemessen.

## 3 These

Annahme: Die Helligkeit der LEDs kann durch Fotowiderstand gemessen und verglichen werden.

## 4 Stand der Technik

### 4.1 Helligkeitssensor

#### 4.1.1 Innerer photoelektrischer Effekt

#### Photoleitung

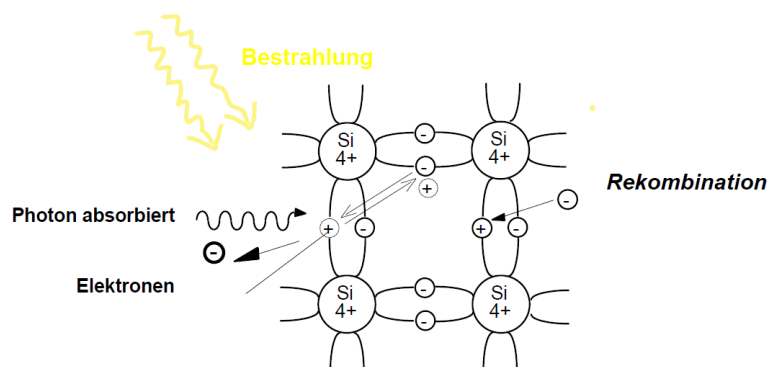


Abbildung 1: Photoelektrischer Effekt[1]

Die Photoleitung bedeutet die Erhöhung der elektrischen Leitfähigkeit von Halbleitermaterialien wegen der Kombination von ungebundenen Elektron-Loch-Paaren bei Bestrahlung. Die Energie des Photons hebt das Elektron aus dem Valenzband in das höherenergetische Leitungsband, daher muss die Energie eines einzelnen Photons mindestens der Bandlücke des leuchtenden Halbleiters entsprechen. Weil die Größe der Bandlücke vom Material abhängt, variiert die maximale Wellenlänge des Lichts, bei der Photoleitung auftritt,

je nach Halbleiter (Galliumarsenid: 0,85  $\mu\text{m}$ , Germanium: 1,8  $\mu\text{m}$ , Silizium: 1,1  $\mu\text{m}$ ). [2]

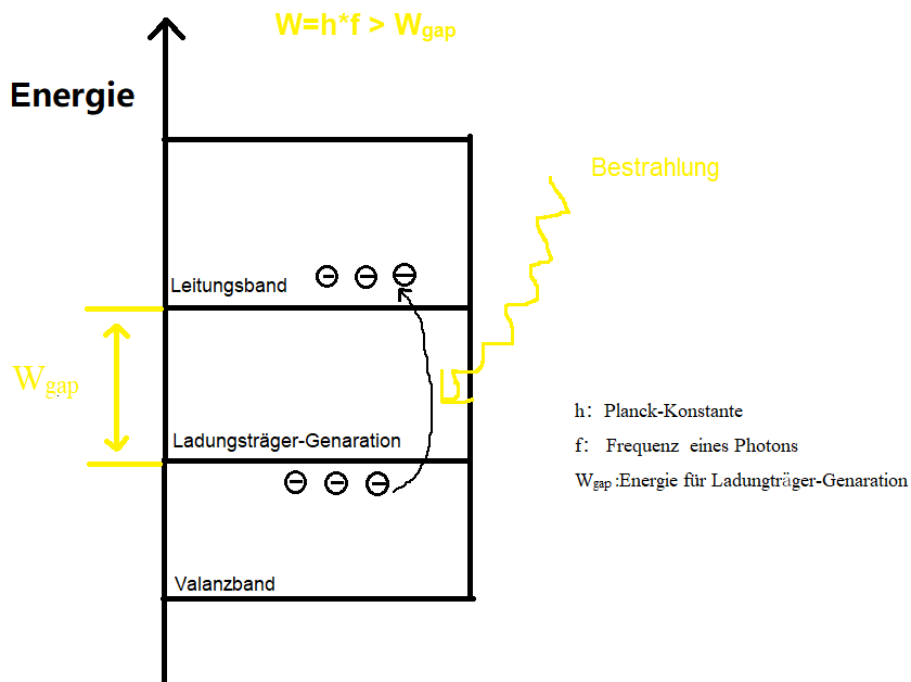


Abbildung 2: Bewegung von Elektronen

Die Photoleitung wird in Photowiderständen, Phototransistoren, Photodioden und CCD-Sensoren (siehe auch pin-Diode und Avalanche-Photodiode) ausgenutzt, welche bei der Herstellung einer Vielzahl von Lichtsensoren Verwendung finden. In diesem Projekt werden zwei Photowiderstände als Sensoren verwendet.

#### 4.1.2 Photowiderstand GL5506

In diesem Projekt wird Photowiderstandsmodell GL5506 verwendet, dies ist ein gebräuchliches Modell von grundlegenden Photowiderständen. Gleichzeitig kann es gut mit Mikrocontrollern wie Arduino UNO, Arduino Mega 2560 verbunden werden. In der Tabelle geht es um detaillierte Informationen von GL5506.



Abbildung 3: Photowiderstand GL5506[3]

Lichtwiderstandsbereich	2~20K $\Omega$
Dunkelwiderstand bei 0 Lux	1M $\Omega$ Min.
Spitzenspektralantwort	540 nm.
Typische Abfallzeit	55ms
Typische Anstiegszeit	45ms
Betriebsspannung $V_{\max}$	100V
Leistung $P_{\max}$	100mW
Widerstandsmaterial	CdS (Cadmiumsulfid).
Nummer von Pins	2-pin Through Hole.
Packungstyp	TO-18 Epoxidharz

Tabelle 1:Elektrische Parameter von GL5506[3]

### Vorteile und Nachteile des Photowiderstands

<b>Vorteil:</b>
1 Sehr breiter Widerstandsbereich 2 Billig, leicht zu bekommen 3 Für hohe Spannungen erhältlich
<b>Nachteil</b>
1 Im Vergleich zu anderen Lichtsensoren reagieren Photowiderstände sehr langsam 2 Temperaturempfindlichkeit.

Tabelle 2 : Vorteile und Nachteile von GL5506

# 5 Umsetzung

## 5.1 Gesamtkonzept

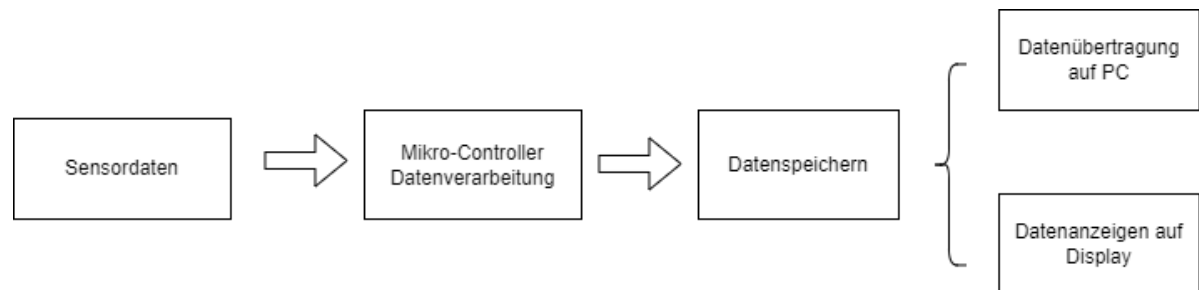


Abbildung 4: Flussdiagramm des Gesamtkonzepts

In dem Projekt werden die Helligkeit auf Basis der Helligkeitssensor (Photowiderstand) gemessen. Weil die Größe des Photowiderstands umgekehrt proportional zur Lichtintensität ist, um den Zusammenhang zwischen Helligkeit und digitalem Signal intuitiver darzustellen, wird ein Photowiderstand in Reihe mit einem Widerstand (470k $\Omega$ ) verbunden, das bedeutet, dass die vom Mikrocontroller empfangene Spannung proportional zur Helligkeit ist.

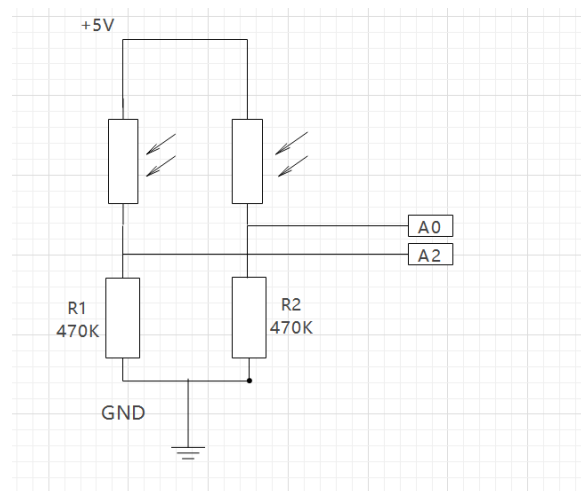


Abbildung 5: Messschaltung

Die Abbildung 4 zeigt die Messschaltung, je stärker das Licht ist, desto kleiner ist der Widerstand des Photowiderstands, die Spannung von Photowiderstand wird auch kleiner. Daher erhöht sich die Spannung des Reihenwiderstands R mit der Zunahme des Lichts. Durch Messen der Spannung von Widerstand R1, R2 kann das optische Signal in ein Spannungssignal von 0 ~ 5v bis 0 ~ 1023 umgewandelt werden. Danach programmiert man



den Arduino so, dass ein Vergleich der Helligkeit der beiden Zündkerzen anhand der Sensordaten erfolgt. Bei der Umsetzung der Hardware wird jeder Photowiderstand mit zwei Leitungen verbunden und zusammen mit der analogen Zündkerze in einer dunklen Umgebung platziert. Die Anschlüsse A0 und A2 werden mit dem Mikrocontroller verbunden.

Entsprechend der Helligkeit der bekannten Zündkerze gibt es drei Ergebnisse für die zu bewertende Zündkerze:

1. Die unbekannte Zündkerze funktioniert normal (Helligkeit 2 ist gleich Helligkeit 1)
2. Die unbekannte Zündkerze ist kaputt (Helligkeit 2 ist kleiner als Helligkeit 1)
3. Die unbekannte Zündkerzenlichtintensität ist zu hoch (Helligkeit 2 ist größer als Helligkeit 1)

In jedem Fall wird eine entsprechende farbige LED als Hinweis beleuchtet.

Sobald der Test gestartet wurde, zeichnet der Mikrocontroller automatisch die Laufzeit jedes Tests und den Echtzeitpunkt der Zündung der bekannten Zündkerze auf, dann wird die Laufzeit im ROM gespeichert, um die Lebensdauer der bekannten Zündkerzen zu bewerten, schließlich werden das Vergleichsergebnis und die aufgezeichnete Zeit durch Umschaltung auf dem Display angezeigt.

Wie in der Abbildung 5 gezeigt, wird das Spannungssignal vom Helligkeitssensoren ausgegeben. Dieses Spannungssignal wird an den analogen Anschluss A0, A2 gesandt. Die nächste Abbildung zeigt die Daten des Verlaufs des Sensors 1 von der Dunkelheit bis Zündung der Zündkerze.

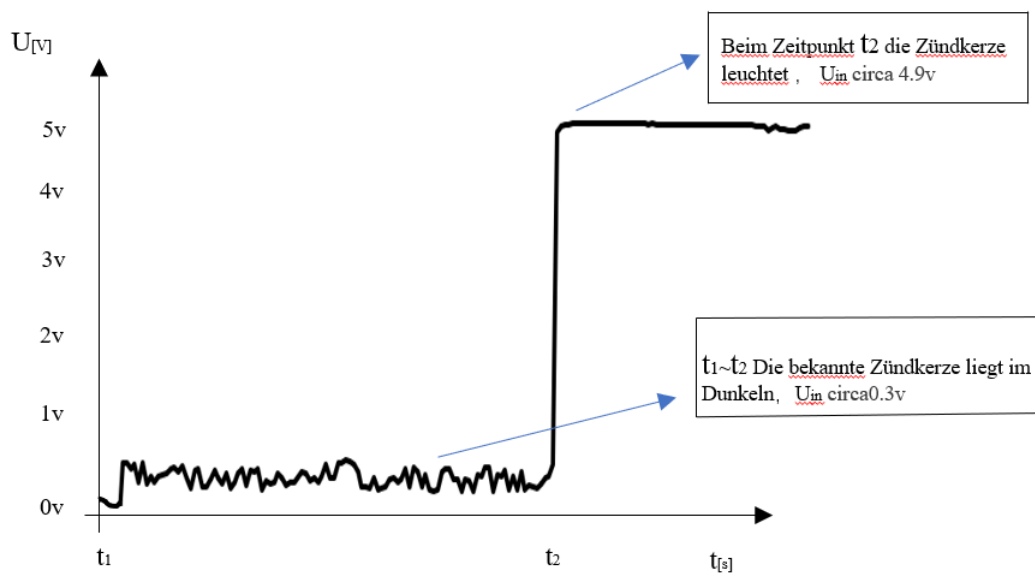


Abbildung 6:analoges Spannungssignal im Pin A0

Die analogen Anschlüsse A0 und A2 entsprechen der Helligkeit der bekannten Zündkerzen und der unbekannten Zündkerze. Die beiden Spannungen werden im Programm in Echtzeit verglichen. Die digitalen Anschlüsse 2, 3 und 4 dient zur Ansteuerung von dreifarbig LEDs. Anschluss 5 wird für die Umschaltung verwendet, normalerweise ist der Anschluss 5 mit einem Pull-Down-Widerstand verbunden, wenn der Schalter gedrückt wird, empfängt der Anschluss 5 einen hohen Pegel und schaltet der Inhalt des Displays um.

## 5.2 Hardware

### 5.2.1 Anforderungen

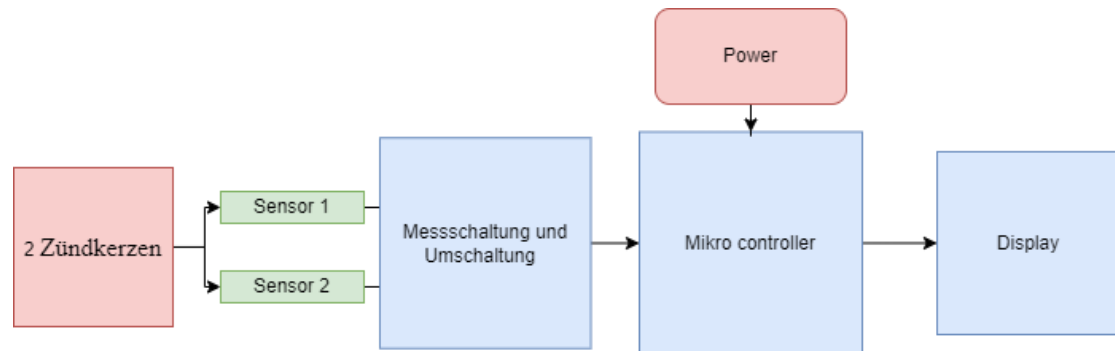


Abbildung 7: Hardware Konzept

Die obige Hardwarebeschreibung sind im Folgenden.

### 1. Arduino Mega 2560 R3

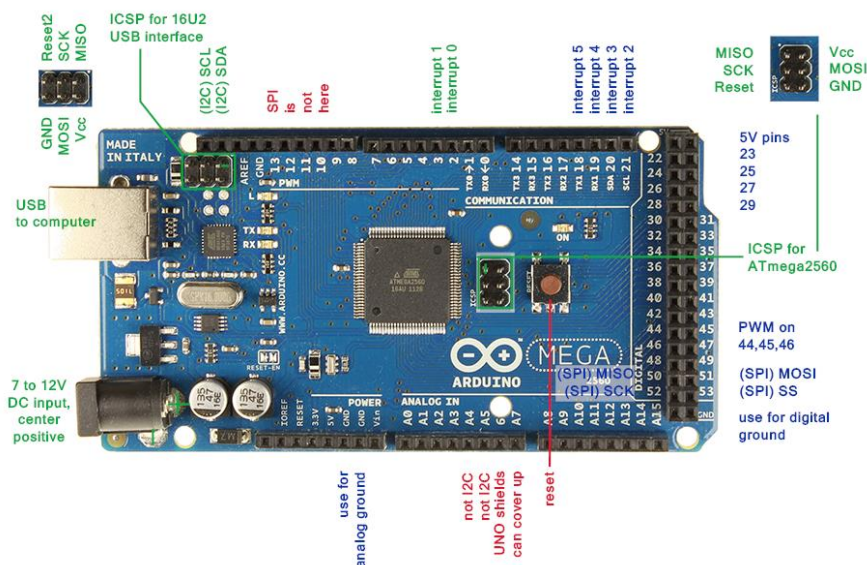


Abbildung 8:Arduino Mega 2560[4]

Das Arduino Mega 2560 ist eine Mikrocontroller-Karte, die auf dem ATmega2560 (Datenblatt) basiert. Es verfügt über 54 digitale Ein- / Ausgangspins (von denen 14 als PWM-Ausgänge verwendet werden können), 16 analoge Eingänge, 4 UARTs (serielle Hardware-Ports), einen 16-MHz-Quarzoszillator, eine USB-Verbindung, eine Stromanschlussbuchse,

einen ICSP-Header, und eine Reset-Taste. Es enthält alles, was zur Unterstützung des Mikrocontrollers erforderlich ist. Schließen Sie es einfach mit einem USB-Kabel an einen Computer an oder versorgen Sie es mit einem AC-DC-Adapter oder einer Batterie, um loszulegen. Es wird den Mikrocontroller Arduino verwendet, um das analoge Eingangssignal zu verarbeiten. Außerdem kann der Radarsensor auch mit 5 V versorgt werden. [5]

## Produktspezifikationen:

- Controller-Kern: ATmega2560
- Steuerspannung: 5V
- Empfohlene Eingangsleistung (empfohlen): 7-12 V.
- Maximale Eingangsspannung (Grenzwerte): 6-20 V.
- Digitale E / A-Pins: 54 (von denen 14 einen PWM-Ausgang bieten)
- Analogeingangspins: 6 Gruppen
- Gleichstrom pro E / A-Pin: 40 mA
- Gleichstrom für 3,3 V Pin: 50 mA
- Flash-Speicher: 256 KB, davon 8 KB vom Bootloader
- SRAM: 8 KB
- EEPROM: 4 KB
- Taktrate: 16 MHz

## 2. LCD1602

Das I<sup>2</sup>C 1602 LCD-Modul ist eine 2-Zeilen-16-Zeichen-Anzeige, die an eine I<sup>2</sup>C-Tochterplatine angeschlossen ist. Die I<sup>2</sup>C-Schnittstelle benötigt nur 2 Datenverbindungen, +5 VDC und GND.

I <sup>2</sup> C Address Range	0x20 to 0x27 (Default=0x27, addressable)
Operating Voltage	5 Vdc
Backlight	white
Size	80 mm x 36mm x 20 mm

Contrast	adjustable by potentiometer on I <sup>2</sup> c interface
GND	ground
VCC	power supply (5V / 3.3V)
SDA	I <sup>2</sup> C data cable
SCL	I <sup>2</sup> C clock line

Tabelle 3:Spezifikation von LCD 1602 [5]

Natürlich wird es in der nachfolgenden Praxis gemäß dem Design des Arduino Mega 2560 mit SCL- und SDA-Kommunikationsports geliefert. Es gibt eine einfache Verbindungsmethode für den I2C-Bus. Es muss nur vier Ports angeschlossen werden (SCL; SDA; VCC; GND). Durch Laden der neuen Bibliothek "LiquidCrystal\_I2C.h" kann es einfach angewendet werden.

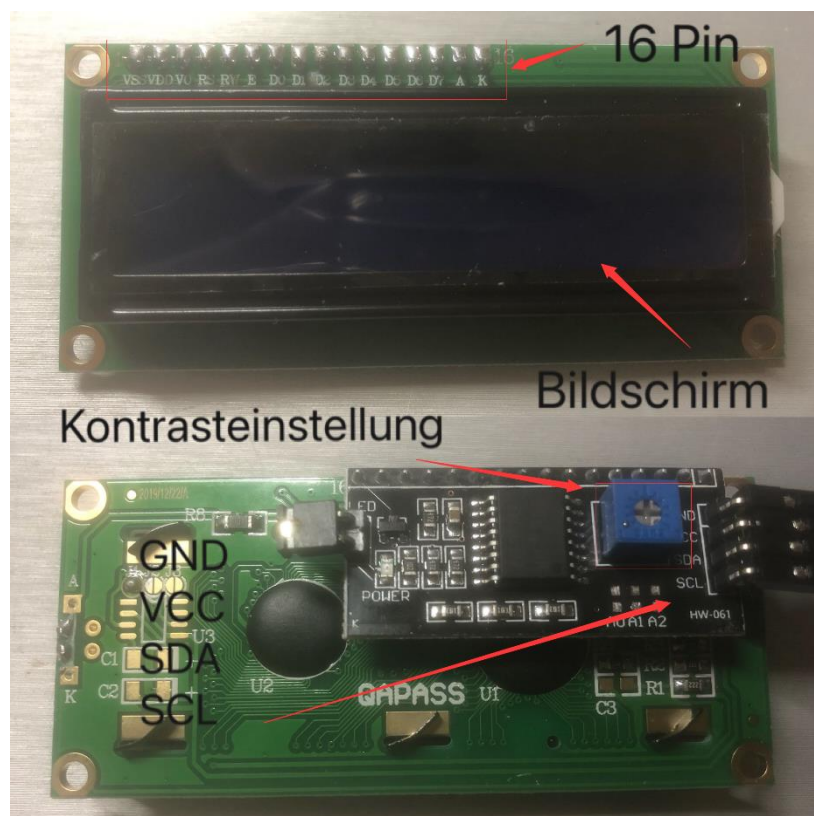


Abbildung 9:Beispiel von LCD 1602 und I2C/SPI LCD

### 3. RTC DS 1302 Modul (Real Time Clock )

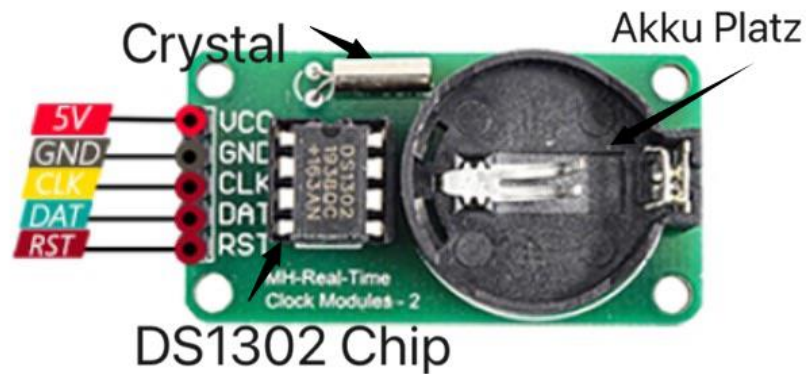


Abbildung 10: DS 1302 Modul[6]

Der DS1302-Chip verfügt über eine Echtzeituhr/einen Kalender und 31 Byte statischen RAM. Durch die serielle Schnittstelle kommuniziert es mit dem Mikrocontroller. Die Echtzeituhr liefert Daten zur Sekunden, Minuten, Stunden, Wochentag, Datum, Monat und Jahr. Nur drei Leitungen sind notwendig, um mit der Uhr/dem RAM zu kommunizieren: CE, I/O (Datenleitung) und SCLK (serielle Uhr). Der DS1302 arbeitet mit sehr geringem Stromverbrauch und hält Daten- und Taktinformationen unter  $1 \mu\text{W}$ . Der DS1302 hat Dual-Power-Pins, einen für die Hauptstromversorgung und den anderen für die Notstromversorgung..[6]

In diesem System wird das RTC-Modul verwendet, um die Echtzeit der Aktivierung der Zündkerzen aufzuzeichnen.

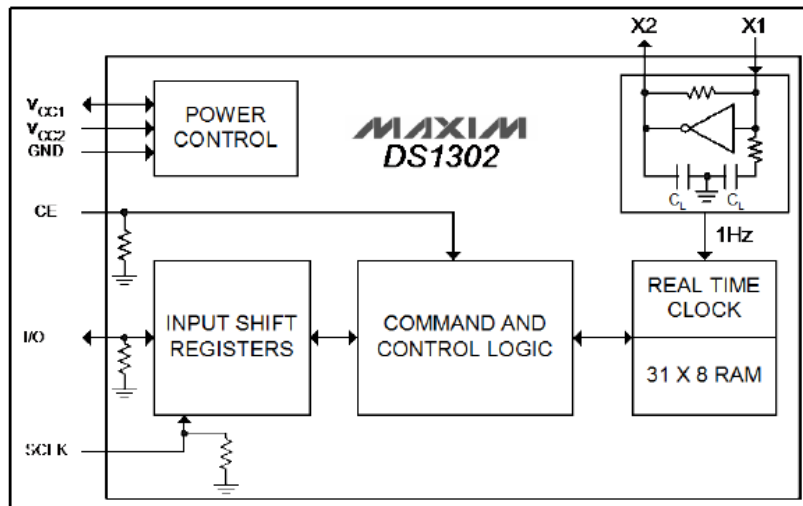


Abbildung 11:Block Diagramm von DS 1302[6]

#### Technische Spezifikationen

Versorgungsspannung (typisch): 3.3V

Spannungsbereich an jedem Pin (bezogen auf Masse): -0,5V bis 7,0°C

Betriebstemperatur (kommerziell): 0 °C bis +70 °C

Betriebstemperatur (industriell): -40°C bis +85°C

Lagertemperaturbereich: -55°C bis +125°C

Löttemperatur: 260°C

In diesem Projekt wird das DS1302 Modul verwendet, um den Echtzeit-Zeitpunkt aufzuzeichnen, wenn die bekannte Zündkerze gezündet wird. Es kann einfach auf IDE Arduino verwendet werden, indem die Bibliotheksdatei DS1302 geladen wird.

## 5.2.2 Systementwurf

### Erforderliche Komponenten:

----- 6 Widerstände ( $\Omega$ ): 470k, 470k, 1k, 1k, 1k, 10k.

----- 3 LED: rot, grün, gelb.

----- 1 Switch.

----- 2 Fotowiderstände

## Schaltplan

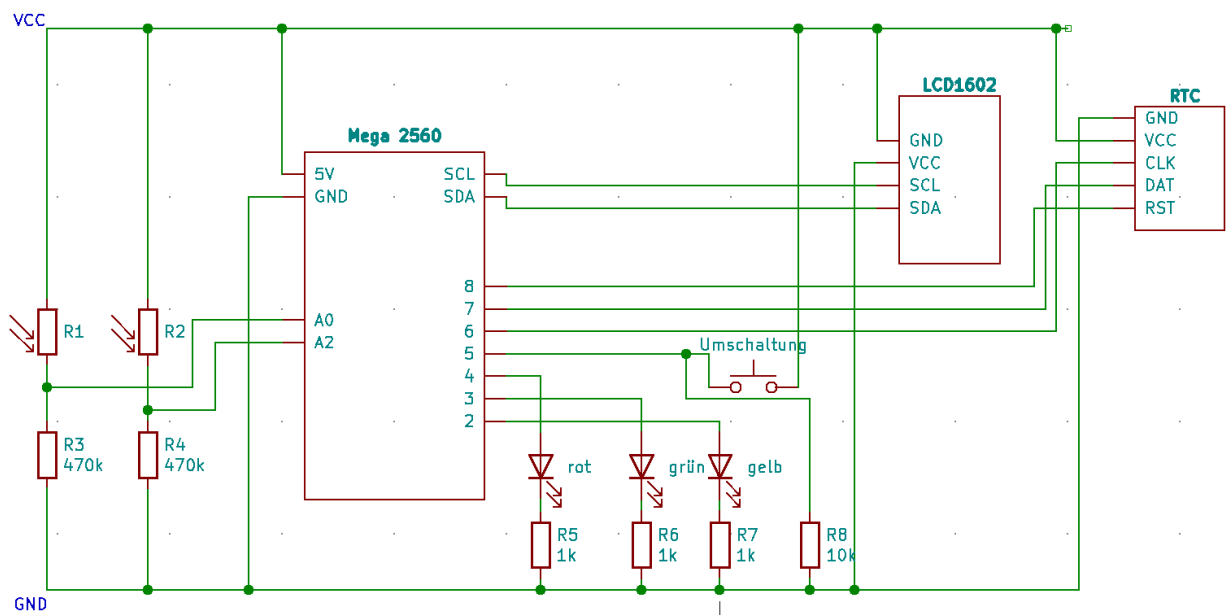


Abbildung 12:Schaltplan

Das ganze Projekt begann mit einem Breadboard. Da im Test zu viele Leitungen verwendet wurden, wird eine passende Leiterplatte für die Größe des Arduino Mega 2560 angefertigt. Mehr PCB-Layout Informationen sind in den Anlagen zu finden. Die für die Simulation verwendeten Zündkerzen werden durch LEDs ersetzt und in einen dunklen Karton gelegt. Die LED-Helligkeit kann durch ein Potentiometer eingestellt werden.

## 5.3 Software

### 5.3.1 Softwarekonzept

----- **Komplette Ablaufdiagramme**



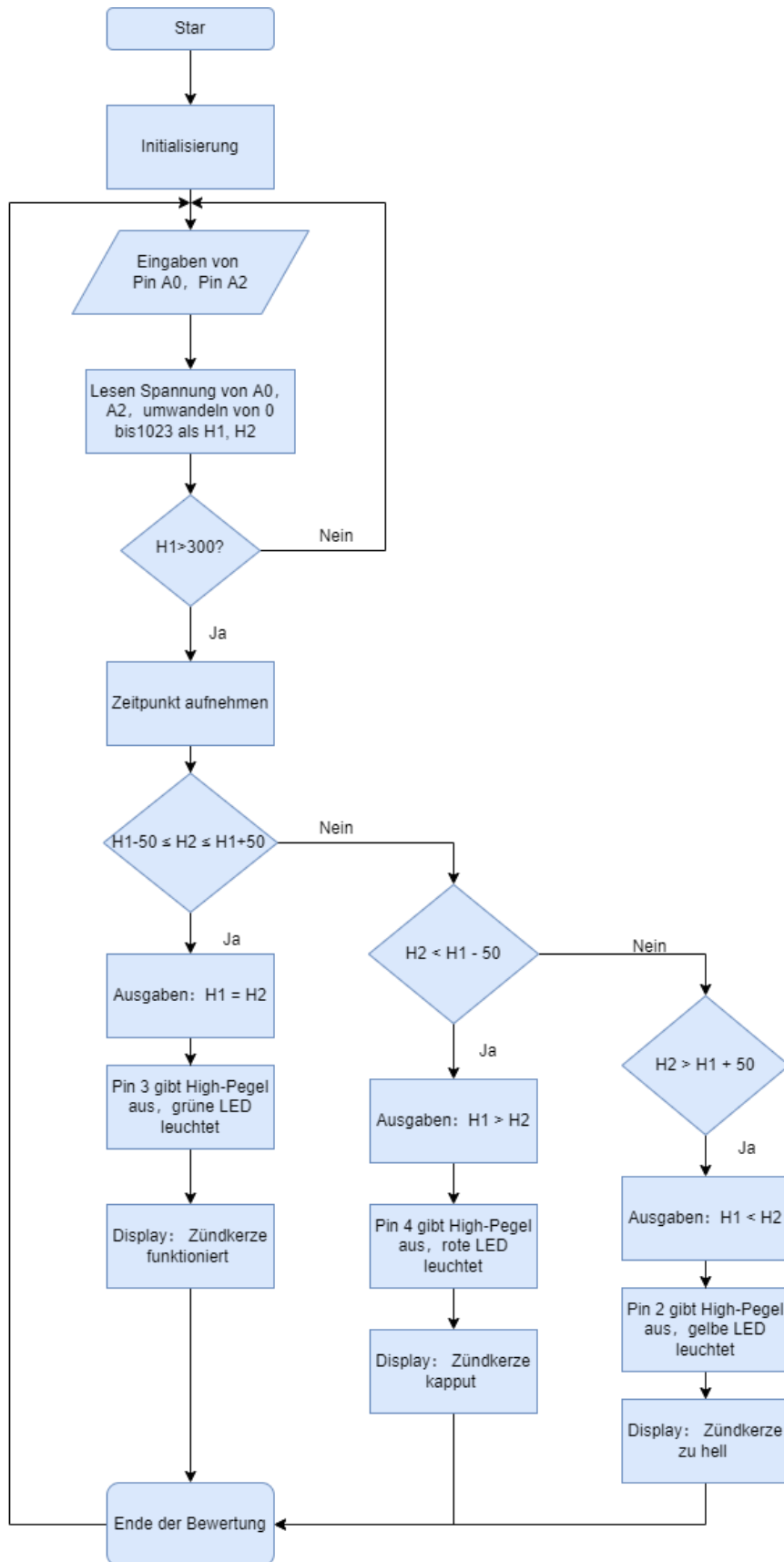


Abbildung 13: Ablaufdiagramm für Messverfahren

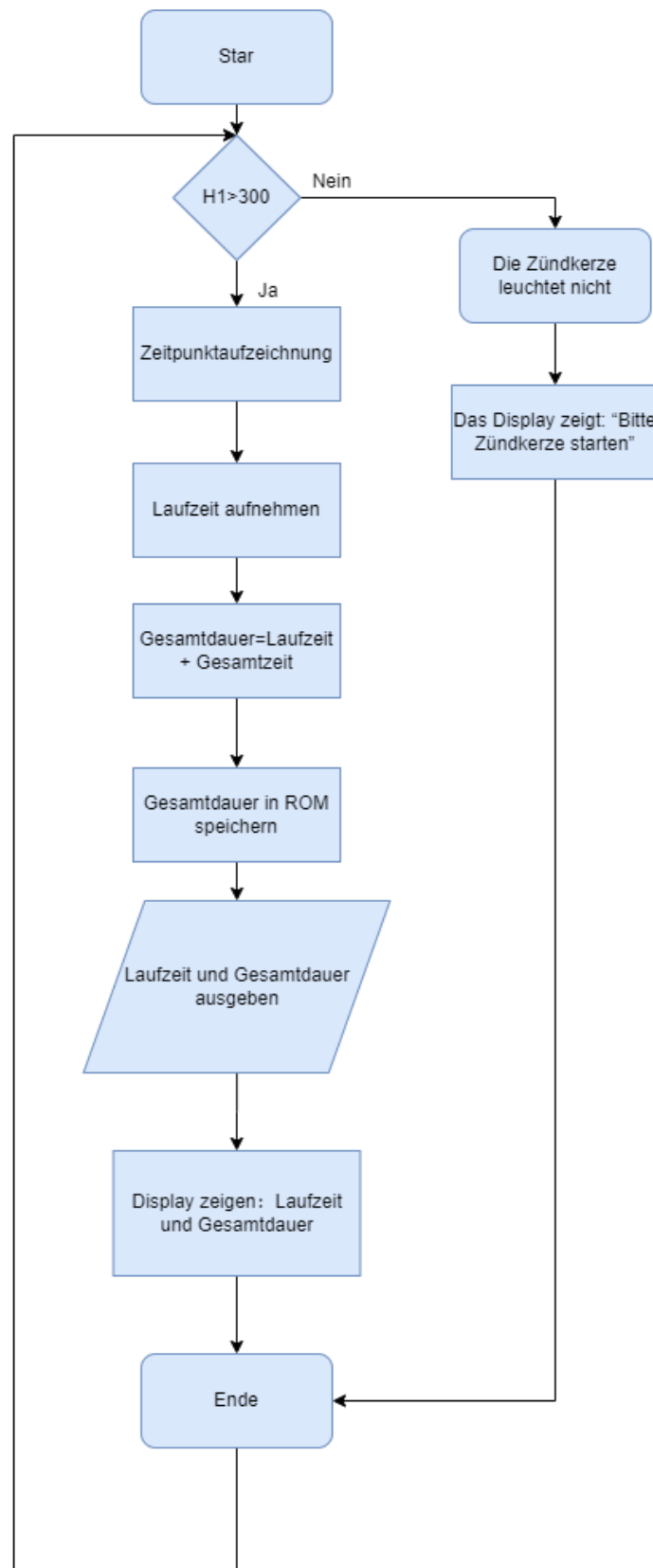


Abbildung 14:Ablaufdiagramm für Zeitaufnahme

### ----- Anmerkung

1. H1=Helligkeit der bekannten Zündkerze, H2=Helligkeit der unbekannten Zündkerze.
2. Wenn  $H1 > 300$  ist, die Zündkerze leuchtet, das System arbeitet.
3. Laufzeit: Die Laufzeit dieses Tests vom Start der Zündkerze.
4. Gesamtzeit: Letzte in ROM gespeicherte Gesamtdauer.
5. Gesamtdauer: Die Gesamtzeit vom ersten Start bis zu diesem Start der bekannten Zündkerze in ihrer Lebensdauer

## 5.3.2 Erklärung des entwickelten Codes

```
X = analogRead(A0);           //Daten von Sensor 1 = X
volt1= X * (5.0 / 1024.0);    //Die Spannung des gemessenen Widerstands 1
Y = analogRead(A2);           //Daten von Sensor 2 = Y
volt2= Y * (5.0 / 1024.0);    //Die Spannung des gemessenen Widerstands 2
buttonstate=digitalRead(5);   //buttonstate mit digitalpin 5 zugeordnet
if (buttonstate == 1) {
    delay(200); Flag++;        //Die vier Zustände der Umschaltung. Jedes Status-LCD zeigt unterschiedliche Inhalte an
    if (Flag>=4) {
        Flag = 1;             //Ein Zyklus: Die Taste wird dreimal gedrückt
    }
}
```

Abbildung 15: Programm von Umschaltung

Das obige Programm steht am Anfang der Schleife, die Daten der beiden Sensoren werden Y und X zugeordnet, dann in die Spannung (volt1 und volt2) umgerechnet, während des Tests werden Spannung und Zeit zur Datenanalyse an Excel übertragen. Wenn die Taste gedrückt wird, wird an Pin5 ein hoher Pegel eingegeben, der Mikrocontroller akzeptiert das Signal und wechselt dann den Zustand, um den Inhalt des Displays zu ändern.

```

if (Flag==1) {                                     //Umstand 1

if (analogRead(A0)>300 && started == false) //Wenn die Zündkerze eingeschaltet ist und boolsche Umstand false ist
{
    starttime=rtc.getTimeStr();                     //Echtzeit durch RTC aufzeichnen
    starttime = millis();                           //Gesamtlaufzeit des Mikrocontrollers
    started = true;                                 //boolsche Umstand true
}
if(started=true&&analogRead(A0)>300)               //Wenn die Zündkerze eingeschaltet ist und boolsche Umstand true ist
{
    timenow=(millis()-starttime)/1000;              //timenow: Laufzeit der Zündkerze bei diesem Test
    timeTotal=timenow+total;                         //timeTotal: Die Gesamtlaufzeit der bekannten Zündkerze, jeder Test
    EEPROM.put(addr,timeTotal);                     //timeTotal wird ins ROM gespeichert
    lcd.setCursor(0,0);                             //LCD zeigt Laufzeit und Startzeitpunkt
    lcd.print("Starttime");
    lcd.print(starttime);
    lcd.setCursor(0,1);
    lcd.print("Laufzeit:");
    lcd.print(timenow);
    lcd.print("s      ");
}
}

```

Abbildung 16: Programm zum Aufzeichnen von Zeitpunkten

Das obige Programm ist Zustand 1, der die Startzeit und die Laufzeit der Zündkerze aufzeichnet, die Laufzeit für jeden Test wird im ROM gespeichert, daher werden die Daten nicht gelöscht, wenn Unit das nächste Mal verwendet werden. Der Aufzeichnung der Laufzeit ist für die Bewertung der Lebensdauer bekannter Zündkerzen notwendig.

```

if (Flag==2) {

if (X-50 <= Y && Y <= X+50) {                     //Wenn der Messwert von Helligkeit 1 und die Daten von Helligkeit 2 ähnlich sind

    lcd.setCursor(0,0);                           //LCD zeigt an, dass die unbekannte Zündkerze funktioniert
    lcd.print("Zündkerze  fun");
    lcd.setCursor(0,1);
    lcd.print("ktioniert H1=H2  ");
    digitalWrite(3,HIGH);                           //Pin3 wird auf High-Pegel gesetzt und die grüne LED leuchtet
    digitalWrite(2,LOW);
    digitalWrite(4,LOW);
}
}

```

Abbildung 17: Programm von Messverfahren

Das obige Programm ist Teil von Zustand 2. Hier wird nur der Fall gezeigt, dass Helligkeit 1 gleich Helligkeit 2 ist. Wenn Helligkeit 1 gleich Helligkeit 2 ist, gibt Pin3 High-Pegel aus und die grüne LED leuchtet. "Zündkerze funktioniert " wird auf dem LCD angezeigt. Weitere Programminformationen befinden sich in der Anlagen.

# **6 Inbetriebnahme / Systemtest**

## **6.1 Aufbau von Hard- und Software**

Die Verbindung zwischen dem Hardwareteil und der Software Arduino erfordert ein USB-Datenkabel. Der Computer wird mit dem Arduino - Board verbunden. Der nächste Schritt kann ausgeführt werden. In der Option " Tools/Boards " muss ein richtiger "Board Manager" ausgewählt werden. (Arduino Mega 2560). Dann ist Port: COM4.

## **6.2 Test**

Zuerst wird das Programm kompiliert, dann wird das Programm auf dem Arduino - Board gebrannt. Wenn es gelingt, blinken TX-Led und RX-Led. Das Programm beginnt mit der Arbeit, dann verbindet Arduino mit einer mobile Batterie Arduino-Board empfängt das Signal und sendet verarbeitetes Signal an das Display. Die Daten werden auf dem Display angezeigt. Der ganze Aufbau funktioniert gut.

# 7 Versuchsdurchführung

## 7.1 Versuchsaufbau

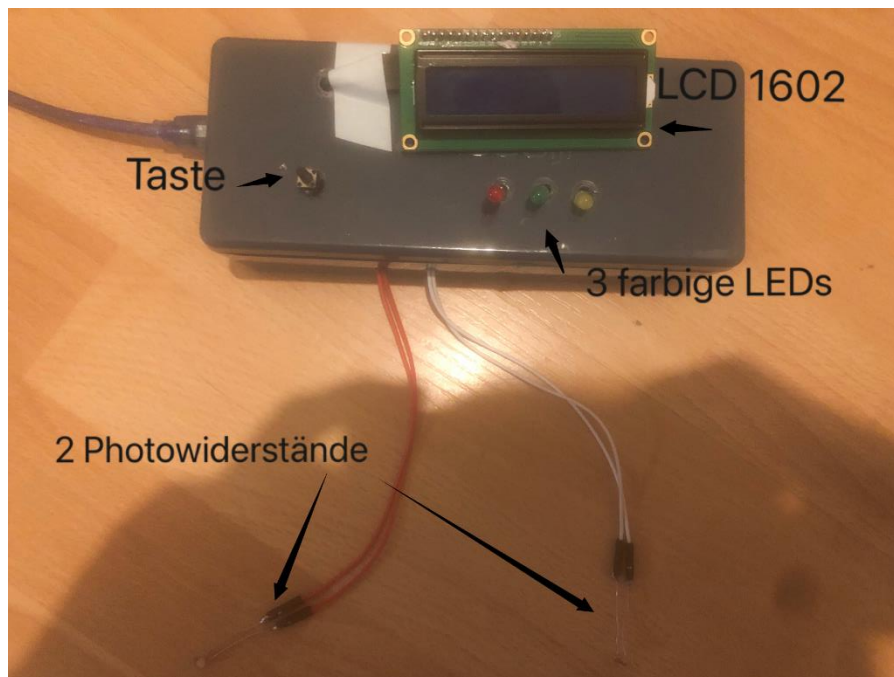


Abbildung 18: Aufbau von System

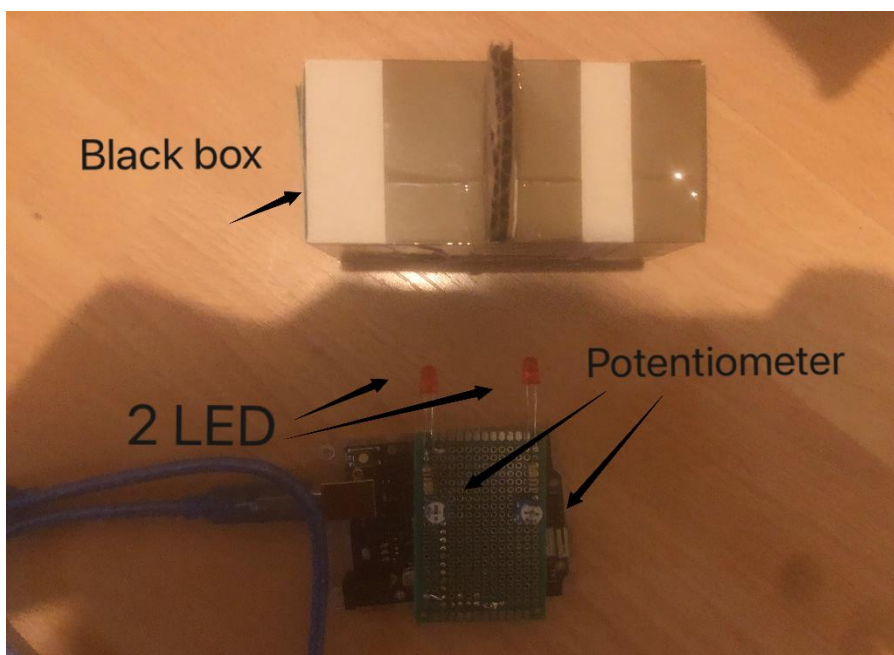


Abbildung 19: Aufbau von LEDs

**Versuchsaufbau:**

Die gelötete Platine kann direkt in den Arduino gesteckt werden, wie in Abbildung 18 gezeigt, wird das gesamte System in eine Kiste gesteckt, Die Oberfläche der Kiste sind eine Taste, ein Display und drei LEDs. Im Test werden beide Sensoren in einen Black Box eingesetzt, um den Einfluss von Fremdlicht zu verhindern.

**Möglichkeiten von Fehlereinflüssen und Lösungen:**

1. Außer dem Licht der Zündkerze gibt es auch Effekte von Fremdlicht, alle Innenbeleuchtungen können während des Tests ausgeschaltet werden oder können das gesamte System beim Testen mit Kleidung abdecken.
2. Weil die Leitfähigkeit des Photowiderstands bei hoher Temperatur zunimmt, werden die Testergebnisse beeinflusst. Bei Verwendung einer echten Zündkerze sollte der Sensor in einem temperaturunabhängigen Bereich platziert werden.

## **7.2 Helligkeitsmessverfahren**

### **7.2.1 Daten in einem Excel-Arbeitsblatt speichern**

**PLX-DAQ**

PLX-DAQ[7] ist eine Excel-Add-In-Software Der serielle Ausgabewert des Arduino kann direkt in das Excel-Blatt auf dem Computer eingegeben werden. Das Bild nach dem Öffnen der Software ist unten zu sehen. Das Format der Ausgabedaten kann durch einfachen Code in Arduino IDE eingestellt werden.

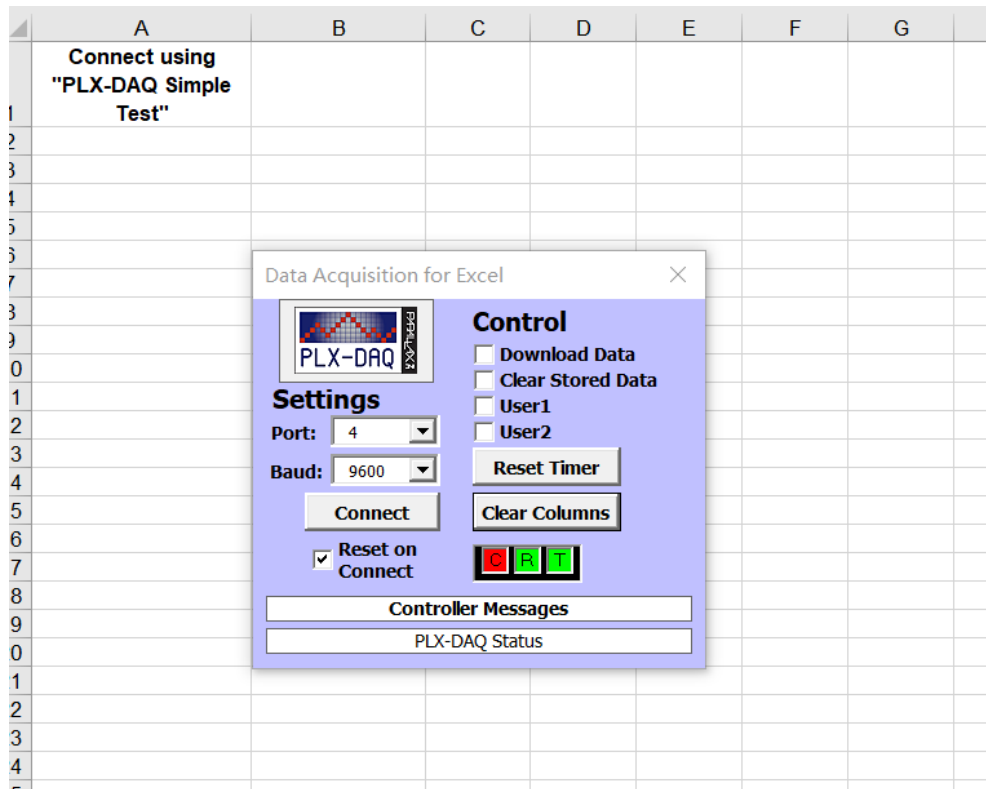


Abbildung 20: PLX-DAQ[7]

## 7.2.2 Test

Die im Test gemessenen Daten sind:

1. Widerstandsspannung, die die LED-Helligkeit darstellt.
2. Ganze Laufzeit.
3. Echtzeit, wenn die bekannte Zündkerze zündet.
4. Gesamtnutzungsdauer der bekannten Zündkerzen

Die gemessene Spannung wird verwendet, um die Helligkeit der Zündkerze zu vergleichen.

Die Laufzeit kann die Zeit des gesamten Tests aufzeichnen und kann auch zur Analyse der Daten verwendet werden. Echtzeitpunkt und Gesamtnutzungsdauer dienen zur Bewertung bekannter Zündkerzen.

Fall 1: Beide Zündkerzen haben die gleiche Helligkeit. Helligkeit 1= Helligkeit 2, die



unbekannte Zündkerze funktioniert.

Time (s)	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
U1(V)	0,30	0,26	0,25	0,21	0,14	4,92	4,93	4,92	4,93	4,92	4,92	4,92	4,92	4,92	4,92	4,92
U2(V)	0,42	0,29	0,26	0,24	0,23	0,22	0,22	0,28	0,67	0,74	4,72	4,78	4,78	4,77	4,73	4,76

Tabelle 4: Testdaten von Fall 1

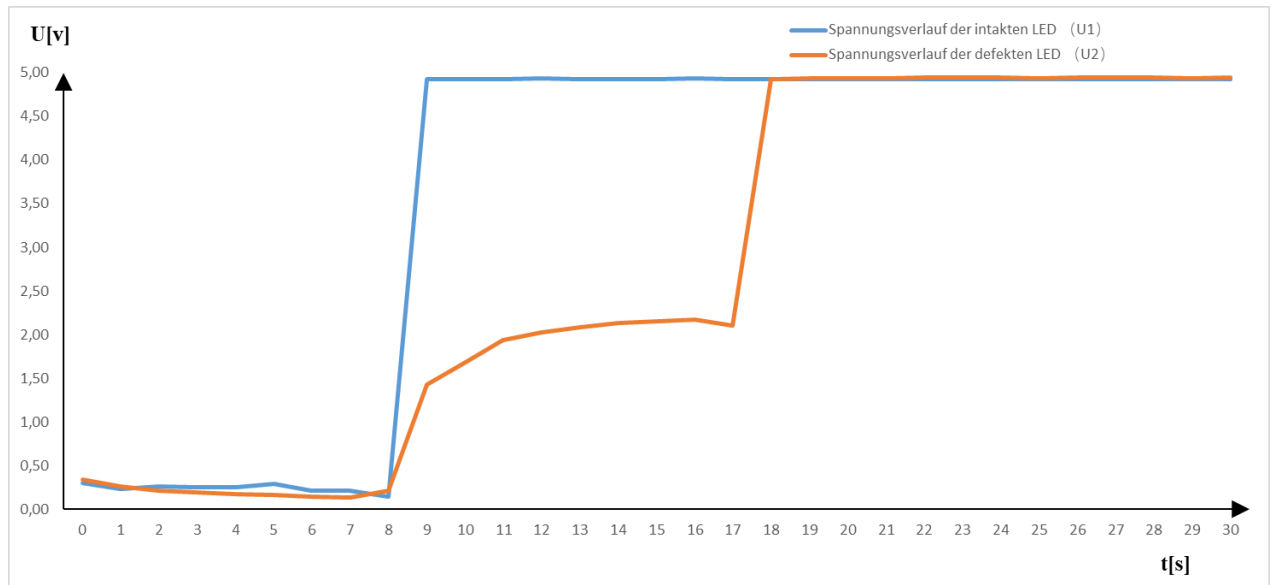


Abbildung 21: Spannungsverläufe der intakten LED (U1) und defekte LED (U2) beim Fall 1

In den ersten acht Sekunden haben die Zündkerzen nicht aktiviert, wenn  $T=8$  s ist, starten die Zündkerzen. Die unbekannte Zündkerzen werden beim Start durch andere Lichter beeinflusst und schließlich bei  $T=18$  s die Helligkeit stabilisieren. Zu diesem Zeitpunkt ist Helligkeit 1 gleich Helligkeit 2 und die grüne LED leuchtet. Dadurch wird angezeigt, dass die unbekannte Zündkerzen weiter verwendet werden kann.



Abbildung 22: Anfangszeit und Laufzeit beim Fall 1



Abbildung 23: Lebensdauer der bekannten Zündkerze beim Fall 1

Fall 2: Beide Zündkerzen haben unterschiedliche Helligkeit. Helligkeit 1 > Helligkeit 2, die unbekannte Zündkerze kaputt.

Time (s)	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
U1(V)	0,10	0,09	0,09	0,26	4,20	4,92	4,92	4,92	4,93	4,92	4,92	4,92	4,92	4,92	4,92	4,92
U2(V)	0,15	0,11	0,1	0,18	1,16	1,42	1,34	2,71	3,49	3,48	3,52	3,53	3,54	3,54	3,66	3,67

Tabelle 5: Testdaten von Fall 2

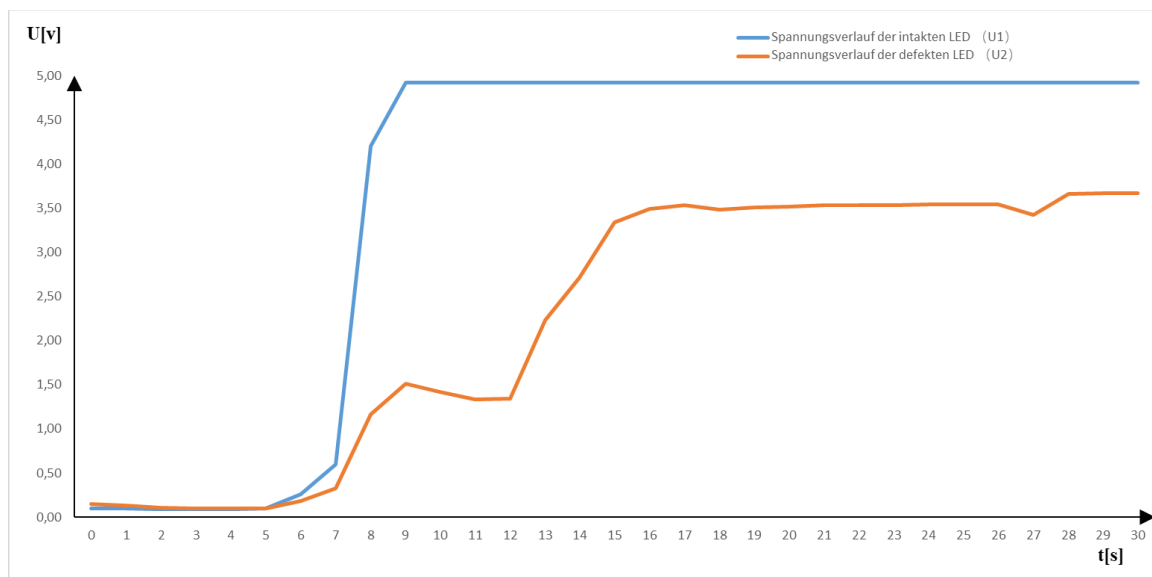


Abbildung 24: Spannungsverläufe der intakte LED (U1) und defekte LED (U2) beim Fall 2

Wenn  $T=8s$  ist, aktiviert die bekannte Zündkerze, die Spannung erreicht 4,92 V. Nachdem die unbekannte Zündkerze gestartet wurde kann die Spannung nur etwa 3,5V erreichen, weil die

Helligkeit sehr gering ist. Dieser Zustand bedeutet, dass die unbekannte Zündkerze beschädigt ist, wenn sie weiter verwendet wird, kann die Zündkerze möglicherweise nicht zünden. Gleichzeitig leuchtet die rote LED.



Abbildung 25:Anfangszeit und Laufzeit beim Fall 2



Abbildung 26: Lebensdauer der bekannten Zündkerze beim Fall 2

Fall 3: Beide Zündkerzen haben unterschiedliche Helligkeit. Helligkeit 1 < Helligkeit 2, die bekannte Zündkerze nähern sich dem Ende ihrer Lebensdauer.

Time (s)	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30
U1(V)	0,10	0,13	0,12	0,10	0,63	0,39	1,94	3,71	3,72	3,73	3,76	4,13	4,11	4,11	4,11	4,11
U2(V)	0,1	0,09	0,08	0,07	0,21	0,19	0,63	1,32	1,22	4,92	4,93	4,92	4,92	4,89	4,89	4,92

Tabelle 6:Testdaten von Fall 3

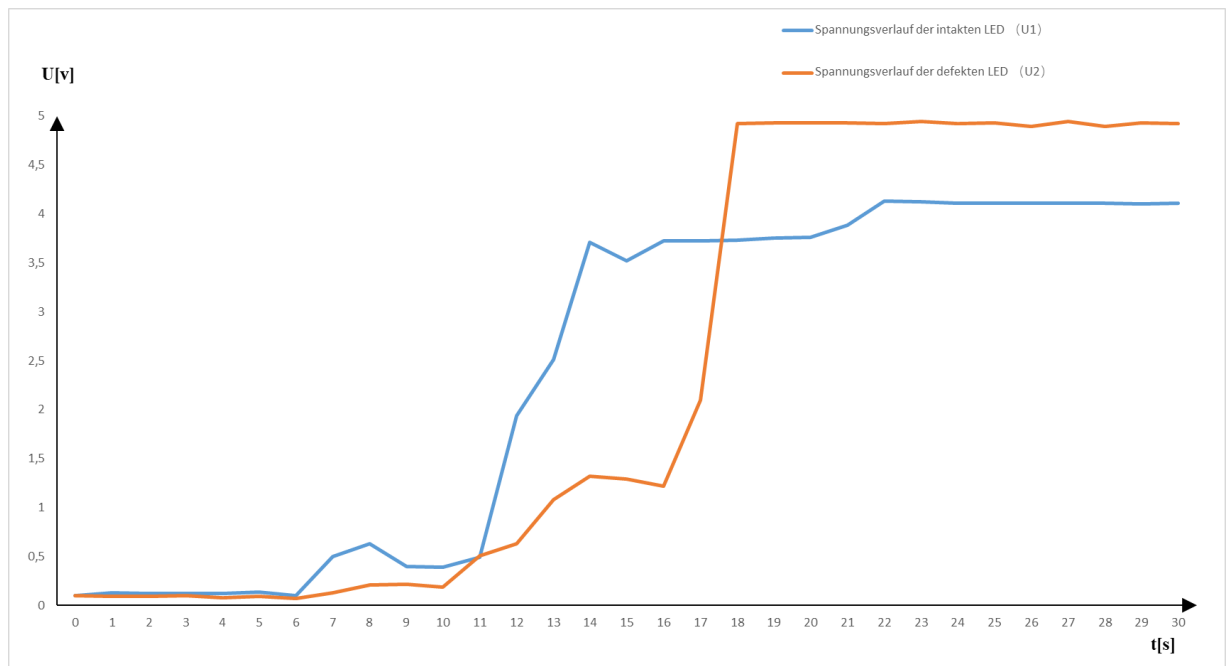


Abbildung 27: Spannungsverläufe der intakten LED (U1) und defekten LED (U2) beim Fall 3

Aus den Daten ist ersichtlich, dass die Helligkeit der bekannten Zündkerze nach längerem Gebrauch 5V nicht erreichen kann. Die unbekannte Zündkerze ist nach dem Start heller als die bekannte Zündkerze. Das bedeutet, wenn Sie das System ordnungsgemäß funktioniert möchten, brauchen vielleicht eine neue Zündkerze.

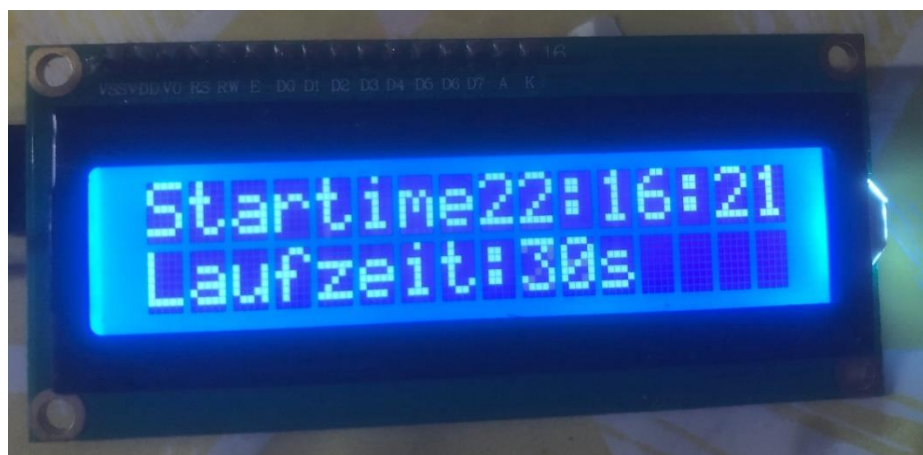


Abbildung 28: Anfangszeit und Laufzeit beim Fall 3

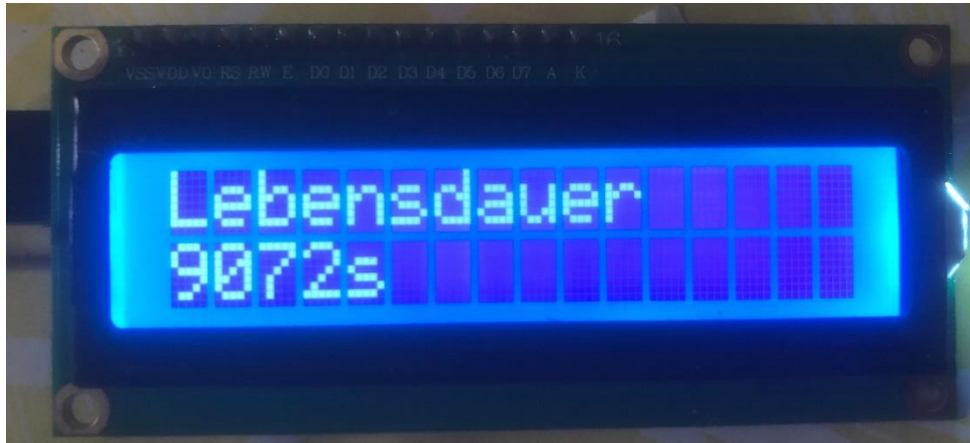


Abbildung 29:Abbildung 22: Lebensdauer der bekannten Zündkerze beim Fall 3

### **Auswertung:**

Das System zur Messung der Zündkerzenhelligkeit war erfolgreich. Die Zeit der drei Testdatensätze beträgt 30s, jedoch kann es aufgrund der Testumgebung zu einem Einfluss von externem Licht kommen, das heißt die Daten können schwanken, bevor sich die Helligkeit stabilisiert.

## 8 Zusammenfassung

Dieser Bachelorarbeit kann hauptsächlich in vier Prozesse unterteilt werden, nämlich Gesamtkonzept, Erstellung des Aufbaus, Programmierung und Test. Die ersten drei Schritte sind relativ einfach, aber es gibt ein Problem mit elektromagnetischer Verträglichkeit im Test, zu diesem Zeitpunkt wurde noch die echte Zündkerze verwendet. Alle Komponenten wurden zum Testen auf dem Steckbrett aufgebaut, aber aufgrund der Störungen durch mangelnde elektromagnetische Verträglichkeit war das endgültige System nicht stabil und nach einer Weile hörte die serielle Kommunikation auf. Die möglichen Lösungen sind zu hoch im Zeit- und Kostenaufwand, daher wird die Zündkerze durch Led ersetzt, dann wird der ganze Prozess einfach.

Im ursprünglichen Design wurde auch ein Komparator LM393 verwendet, um die Frequenz des Zündkerzenflimmerns zu testen, aber nach dem Ersatz des LEDs macht die Einstellung des Komparators nicht viel Sinn, deshalb wurde es in der endgültigen Entscheidung entfernt. Insgesamt ist es im Test nicht zufriedenstellend, die mögliche Lösung wurde aufgrund der hohen Kosten und Schließung des Labors nicht fortgefahren, aber die mögliche Lösung und Testdetails findet sich im nächsten Kapitel „Ausblick“.

## 9 Ausblick

Dieses Projekt realisierte die Helligkeitsmessung der simulierten Zündkerze durch den Mikrocontroller, aber leider ist es noch keine echte Zündkerze. Durch die Wechselspannung der Starterzündkerze hochfrequente Wechsellmagnetfeld wurden sehr große Einflüsse auf den Low-Power-Mikrocontroller erzeugt. Das Oszilloskop zeigt, dass die Erregerspannung in der Leitung etwa 14 V ist. Die maximale Eingangsspannung des Arduino-Boards beträgt jedoch 5 V und die serielle Kommunikation konnte nicht stabil funktionieren.

Im Labor kann zum Anschluss des Sensors Koaxial Kabel mit der Fähigkeit zur Abschirmung magnetischer Felder verwendet werden, dann verbindet den Kondensator parallel mit Photowiderstand, um zu verhindern, dass die plötzliche Spannungsänderung zu groß wird, aber nach dem Test beträgt die maximale Laufzeit etwa 10 Sekunden und dann wird die serielle Kommunikation beendet, das ganze System ist sehr instabil.

Es gibt eine andere Möglichkeit, einen elektromagnetisch abschirmenden Raum mit Metall oder elektromagnetischem Abschirmmaterial zu bauen, wie ein faradayscher Käfig und die Zündkerze darin einzusetzen.

Das bedeutet, dass bei dieser Methode wenig Platz für die Sensormessung gelassen werden könnte. Theoretisch ist diese Methode machbar: ein elektromagnetischer Abschirmraum könnten um die beiden Zündkerzen gebaut werden, dann können genaueren lichtempfindlichen Sensor ausgewählt und die Eingangsspannung zum Mikrocontroller stabilisiert werden. Dies sollte das Problem der Elektromagnetischen Verträglichkeit lösen.

# Literatur

- [1] Formelsammlung\_Elektronik\_Analoge-Schaltungstechnik , Prof. Dr.-Ing. Michael Kuhl  
[Index of /intranet/Lehre/INW/Kuhl/Elektronik - Analoge Schaltungstechnik EA-MG-LT-ME-IN/04 E-Learning-Content/01 e-Vorlesung \(hs-mittweida.de\)](#)
  
- [2] „PhotoelektrischerEffekt“ [Online]  
[http://biancahoegel.de/quanten/photoelekt\\_effekt.html](#)
  
- [3] „GL 5506 datasheet“ [Online]  
[Datasheet GL55 - Vorschau und Download \(rlocman.de\)](#)
  
- [4] „Arduino Mega 2560“, 2020. [Online]  
[http://eprints.polsri.ac.id/4598/8/File%20VIII%20%28Lampiran%29.pdf](#)
  
- [5] „Datasheet I2C 1602 Serial LCD Module“, 2020. [Online]:  
[https://opencircuit.shop/resources/file/da88acc1702a90667728fcf4ac9c75c455475706466/I2C-LCD-interface.pdf](#)
  
- [6] „ Datasheet DS 1302 Modul “[Online]  
[https://html.alldatasheet.com/html-pdf/58478/DALLAS/DS1302/180/1/DS1302.html](#)
  
- [7] „PLX-DAQ“[Online]  
[https://www.parallax.com/package/plx-daq/](#)



# Anlagen

## 1.Schalplan und PCB Layout

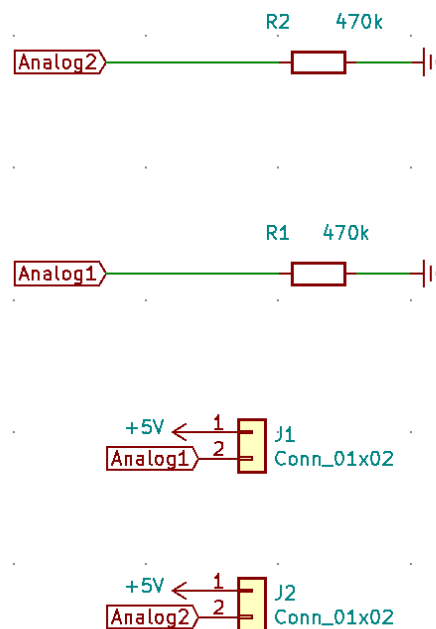


Abbildung 30: Schaltungsplan1

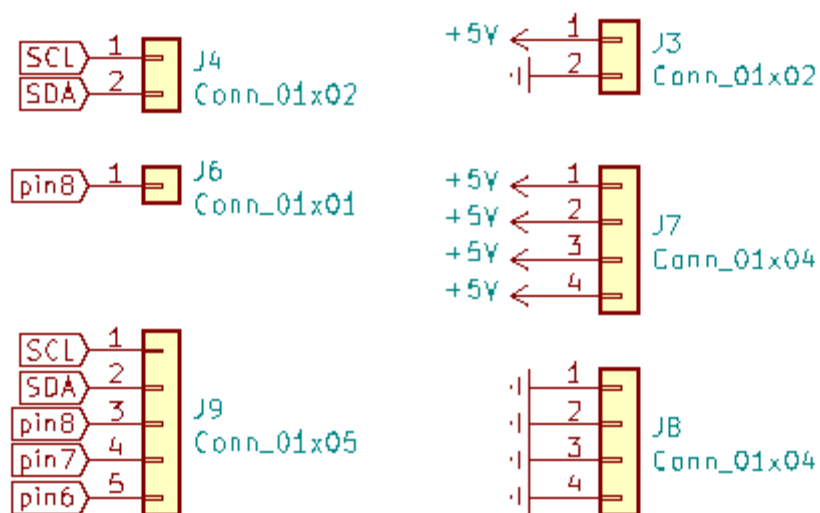


Abbildung 31 : Schaltungsplan 2

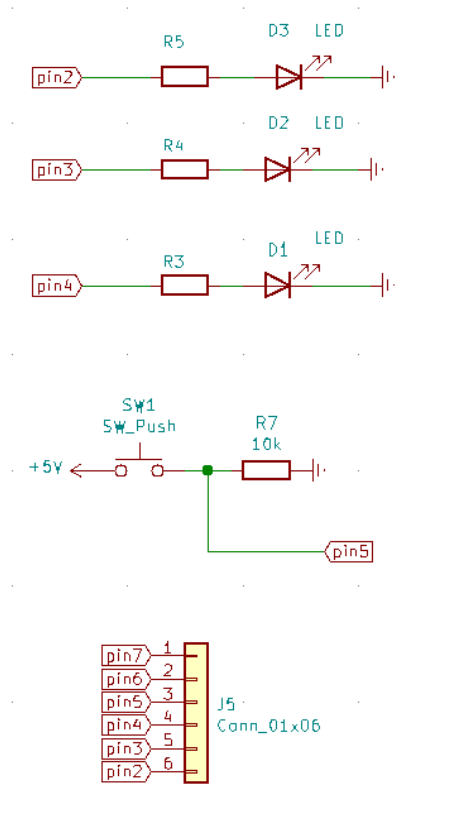


Abbildung 32: Schaltungsplan 3

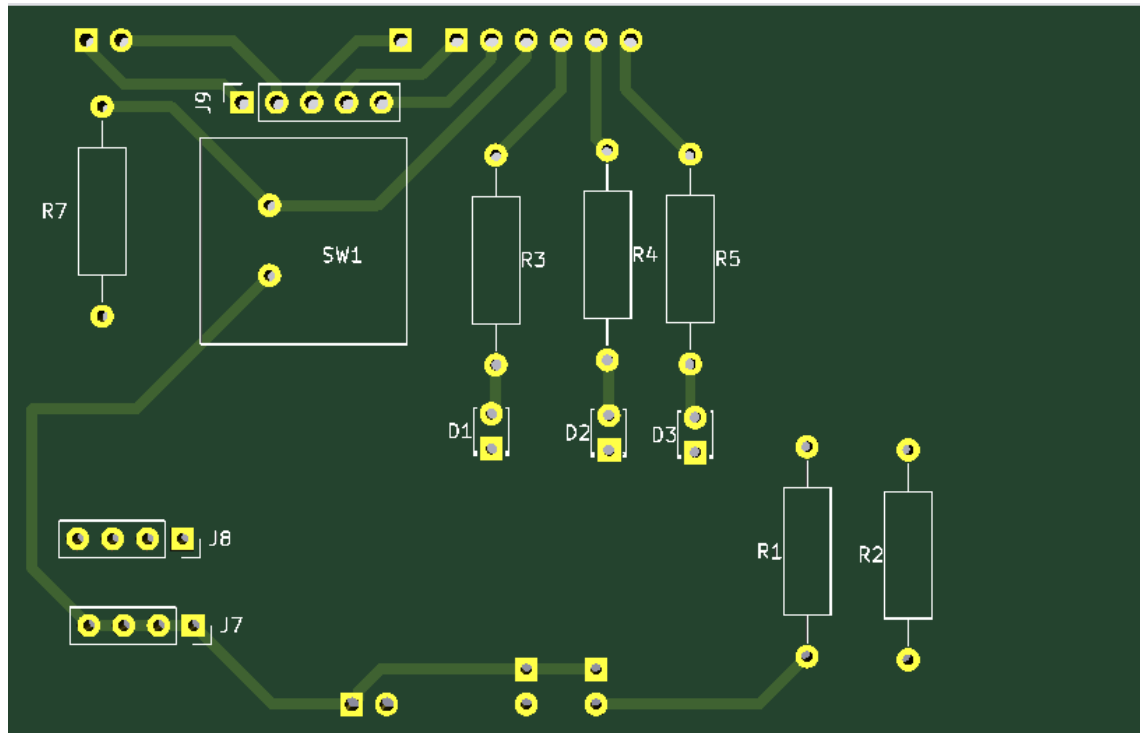


Abbildung 33: Abbildung 30: PCB(Vorderseite)

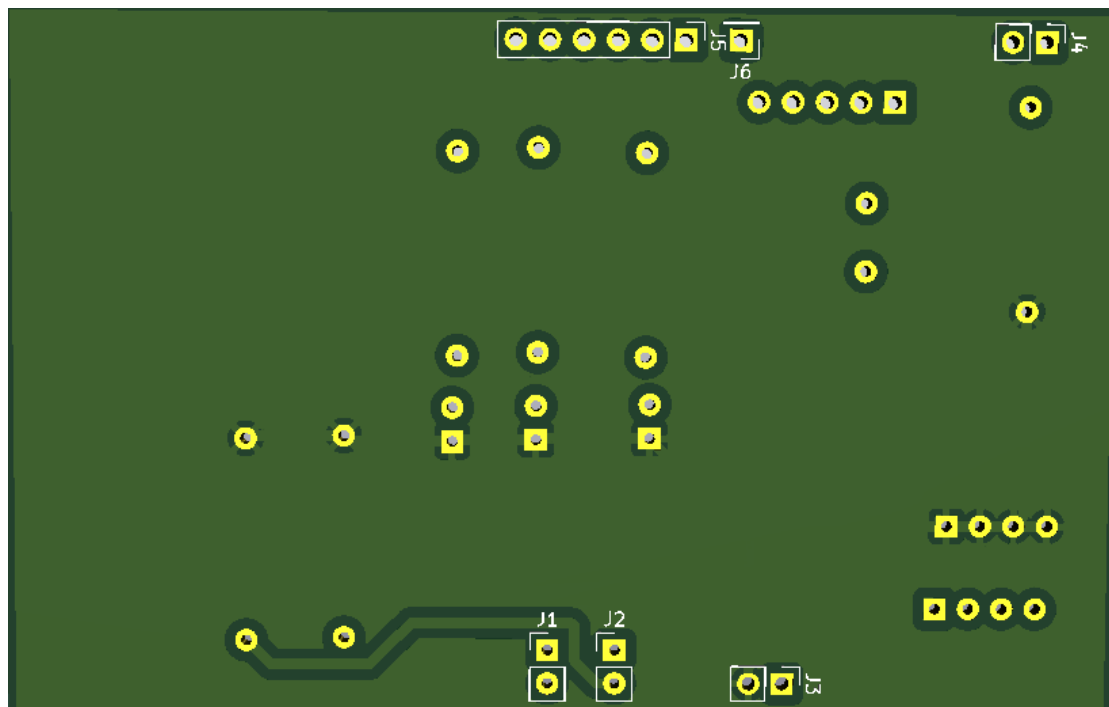


Abbildung 34: Abbildung 31: PCB(Rückseite)

## 2.Programm für die Vorrichtung

```
1. #include <Wire.h>
2. #include <LiquidCrystal_I2C.h>
3. #include <DS1302.h>
4. #include <EEPROM.h>
5.
6. int Flag=0;
7. int X = 0;
8. int Y = 0;
9. int count = 0;
10. int lastbottonstate = 0;
11. int buttonstate =0 ;
12. unsigned long starttime;
13. unsigned long timeoff;
14. unsigned long timenow;
15. boolean started = false;
16. int addr=5;
17. String starttime;
18. unsigned long timeTotal=0;
19. unsigned long total=0;
20. float volt1;
21. float volt2;
22.
23. LiquidCrystal_I2C lcd(0x3F,16,2);
24. DS1302 rtc(8,7,6);
25.
26. void setup() {
27.   Serial.begin(9600);
28.   Serial.println("CLEARDATA");
29.   Serial.println("LABEL,Time,U1,U2");
30.   lcd.init();
31.   lcd.backlight();
32.   pinMode(A0,INPUT);
33.   pinMode(A2,INPUT);
34.   pinMode(2,OUTPUT);
35.   pinMode(3,OUTPUT);
36.   pinMode(4,OUTPUT);
37.   pinMode(5,INPUT);
38.
39.   EEPROM.get(addr,total);
40.   rtc.halt(false);
41.   rtc.writeProtect(false);
42.   //rtc.setDOW(THURSDAY);
```

```

43. // rtc.setTime(12,25,12);
44. //rtc.setDate(16,12,2021);
45. }
46.
47. void loop(){
48. //Serial.print("Pin 5 stasue");
49. //Serial.println(digitalRead(5));
50. //Serial.print("count statues");
51. //Serial.println(count);
52. //Serial.print(rtc.getTimeStr());
53. Serial.print("LDR1 Reading");
54. Serial.println(analogRead(A0));
55. Serial.print("LDR2 Reading");
56. Serial.println(analogRead(A1));
57.
58. X = analogRead(A0); //Daten von Sensor 1 = X
59. volt1= X * (5.0 / 1024.0); //Die Spannung des gemessenen Widerstands 1
60. Y = analogRead(A2); //Daten von Sensor 2 = Y
61. volt2= Y * (5.0 / 1024.0); //Die Spannung des gemessenen Widerstands 2
62. buttonstate=digitalRead(5); //buttonstate mit digitalpin 5 zugeordnet
63. if (buttonstate == 1) {
64.     delay(200); Flag++; //Die vier Zustände der Umschaltung,
        Jedes Status-LCD zeigt unterschiedliche Inhalte an
65.     if (Flag>=4) {
66.         Flag = 1; //Ein Zyklus: Die Taste wird dreimal gedrückt
67.     }
68. }
69.
70.
71. if (Flag==0)
72. {
73.     digitalWrite(4,LOW);
74.     digitalWrite(2,LOW);
75.     digitalWrite(3,LOW);
76.
77.     lcd.setCursor(0,0);
78.
79.     lcd.print(rtc.getDOWStr());
80.     lcd.print(" ");
81.     lcd.print(rtc.getDateStr());
82.     lcd.setCursor(0,1);
83.     lcd.print(" Willkommen ");
84. }

```

```

85.
86.
87.     if(Flag==1){                                     //Umstand 1
88.
89.         if (analogRead(A0)>300 && started == false) //Wenn die Zündkerze eingesc
            haltet ist und boolsche Umstand false ist
90.     {
91.         starttime=rtc.getTimeStr();                   //Echtzeit durch RTC aufzeic
            hnen
92.         starttime = millis();                         //Gesamtlaufzeit des Mikroco
            ntrollers
93.         started = true;                               //boolsche Umstand true
94.     }
95.         if(started=true&&analogRead(A0)>300)         //Wenn die Zündkerze eingesc
            haltet ist und boolsche Umstand true ist
96.     {
97.         timenow=(millis()-starttime)/1000;           //timenow:
            Laufzeit der Zündkerze bei diesem Test
98.         timeTotal=timenow+total;                     //timeTotal:
            Die Gesamtlaufzeit der bekannten Zündkerze, jeder Test
99.         EEPROM.put(addr,timeTotal);                 //timeTotal wird ins ROM ges
            peichert
100.         lcd.setCursor(0,0);                         //LCD zeigt Laufzeit
            und Startzeitpunkt
101.         lcd.print("Startime");
102.         lcd.print(starttime);
103.         lcd.setCursor(0,1);
104.         lcd.print("Laufzeit:");
105.         lcd.print(timenow);
106.         lcd.print("s      ");
107.     }
108.
109.         if (analogRead(A0) < 300){
110.             timeoff=millis()-starttime;
111.             digitalWrite(4,LOW);
112.             digitalWrite(2,LOW);
113.             digitalWrite(3,LOW);
114.             lcd.setCursor(0,0);
115.             lcd.print("Bitte das Auto   ");
116.             lcd.setCursor(0,1);
117.             lcd.print("starten      ");
118.
119.             if(started==true){
120.                 EEPROM.get(addr,total);

```

```

121.         started = false;
122.     }
123. }
124. }
125.
126.     if(Flag==2){
127.
128.         if(X-
            50 <= Y &&Y <= X+50) {           //Wenn der Messwert von Helligkeit 1 und die Da
            ten von Helligkeit 2 ähnlich sind
129.
130.             lcd.setCursor(0,0);           //LCD zeigt an, dass die unbeka
            nnte Zündkerze funktioniert
131.             lcd.print("Zündkerze   fun");
132.             lcd.setCursor(0,1);
133.             lcd.print("ktioniert H1=H2   ");
134.             digitalWrite(3,HIGH);        //Pin3 wird auf HighPegel gesetzt und di
            e grüne LED leuchtet
135.             digitalWrite(2,LOW);
136.             digitalWrite(4,LOW);
137.         }
138.
139.         if(Y < X-50) {
140.
141.             lcd.setCursor(0,0);
142.             lcd.print("Zündkerze       ");
143.             lcd.setCursor(0,1);
144.             lcd.print("kaputt   H1>H2       ");
145.             digitalWrite(4,HIGH);
146.             digitalWrite(2,LOW);
147.             digitalWrite(3,LOW);
148.         }
149.
150.         if(Y > X+50) {
151.
152.             lcd.setCursor(0,0);
153.             lcd.print("Zündkerze       ");
154.             lcd.setCursor(0,1);
155.             lcd.print("zu heller   H1<H2       ");
156.             digitalWrite(2,HIGH);
157.             digitalWrite(3,LOW);
158.             digitalWrite(4,LOW);
159.         }
160.

```

```

161.         Serial.print("DATA, ");
162.         Serial.print(rtc.getTimeStr());
163.         Serial.print(",");
164.         Serial.print(volt1);
165.         Serial.print(",");
166.         Serial.println(volt2);
167.     }
168.
169.     if(Flag==3){
170.
171.         if (analogRead(A0)>300 && started == false)
172.         {
173.             starttime = millis();
174.             started = true;
175.         }
176.
177.         if(started==true&&analogRead(A0)>300)
178.         {
179.             timenow=(millis()-starttime)/1000;
180.             timeTotal=timeNow+total;
181.             EEPROM.put(addr,timeTotal);
182.             Serial.println(timeTotal);
183.             lcd.setCursor(0,0);
184.             lcd.print("Lebensdauer      ");
185.             lcd.setCursor(0,1);
186.
187.             lcd.print(timeTotal);
188.             lcd.print("s          ");
189.
190.         }
191.
192.         if (analogRead(A1) < 300){
193.             timeoff=millis()-starttime;
194.             lcd.setCursor(0,0);
195.             lcd.print("Bitte das Auto      ");
196.             lcd.setCursor(0,1);
197.             lcd.print("starten          ");
198.
199.             if(started==true){
200.                 EEPROM.get(addr,total);
201.                 started = false;
202.             }
203.         }
204.     }

```



205. }

# Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Alle sinngemäß und wörtlich übernommenen Textstellen aus fremden Quellen wurden kenntlich gemacht.

Mittweida, 20.01.2022

Unterschrift: