Praktikumsbericht

*Betriebspraktikum bei der Firma Krohne Messtechnik GmbH*



KROHNE Messtechnik GmbH  
Ludwig-Krohne-Str. 5  
47058 Duisburg

Inhalt

[Erwartungen 3](#_Toc492232201)

[Vorstellung der Praktikumsstelle 3](#_Toc492232202)

[Tagesberichte 4](#_Toc492232203)

[Wochenbericht 1 7](#_Toc492232204)

[Wochenbericht 2 10](#_Toc492232205)

[Fazit 12](#_Toc492232206)

[Anhang 13](#_Toc492232207)

[Quelltext 13](#_Toc492232208)

[Oszilloskop-Bilder des PWM-Signals 23](#_Toc492232209)

[Sehr geringe Helligkeit 23](#_Toc492232210)

[Mittlere Helligkeit 24](#_Toc492232211)

[Hohe Helligkeit 24](#_Toc492232212)

[Fotos des Praktikumsprojekts 25](#_Toc492232213)

# Erwartungen

Von meinem Betriebspraktikum bei der Firma Krohne Messtechnik GmbH erwarte ich, dass es mich in meinem Plan Informatik zu studieren bestätigt. Ich hoffe, dass die Arbeit als solche so ist, wie ich es mir vorgestellt habe und mir auch Spaß macht. Außerdem erwarte ich, dass es interessant wird und dass ich viele neue Sachen bezogen auf den Bereich Informatik aber auch auf andere Bereiche lerne.

# Vorstellung der Praktikumsstelle

Ich mache mein Betriebspraktikum bei der Firma Krohne Messtechnik GmbH. Krohne stellt Messgeräte für die Prozessautomatisierung her und verkauft diese an Firmen aus verschiedenen Branchen, z.B. Wasser und Abwasser, Öl und Gas, Lebensmittel, Chemie, Pharmazie, Energieversorgung und Marinetechnik. Die Firma wurde 1921 in Duisburg von Ludwig Krohne gegründet. Damals entwickelte Krohne Messtechnik nur Schwebekörper-Durchflussmessgeräte. Nach der Zerstörung der Fabriken während des Zweiten Weltkrieg übernahm im Jahre 1949 Anna Krohne zusammen mit ihrem Enkel Kristian Rademacher-Dubbick die Firma. Unter deren Führung wurde Krohne Messtechnik GmbH zu einem der führenden Unternehmen für Durchflussmessgeräte. 1979 gab Kristian Rademacher-Dubbick seine Position als Geschäftsführer an den momentanen Geschäftsführer Michael Rademacher-Dubbick ab und starb im Jahr 2014. Heute ist die Firma Krohne Messtechnik GmbH einer der führenden Messgeräte-Hersteller und entwickelte Messgeräte für verschiedene Durchflussmessverfahren, andere physikalische Größen und die Analysetechnik*.* Mein Praktikumsplatz ist in der Abteilung für Kommunikationssoftware und dort in der Gruppe „Human Machine Interfaces (HMI)“.

# Tagesberichte

|  |  |
| --- | --- |
| 1. **Tag 26.06.2017** | * Als erstes wurde mir die Firma in vorgestellt, dazu sind wir in einen Meetingraum gegangen. * In dem Meetingraum wurde mir und der anderen Praktikantin das Projekt, das wir bearbeiten sollen, vorgestellt. * Daraufhin sind wir in das Büro meines Praktikumsbetreuers gegangen und haben uns dort mit der Arduino-Entwicklungsumgebung auseinandergesetzt. Mein Praktikumsbetreuer ist Entwicklungsingenieur in der Gruppe HMI. * Dann habe ich mit der Programmierung einer Einlese-Funktion durch die Konsole begonnen. * Ich habe gelernt, wie man mit der Arduino-Entwicklungsumgebung arbeitet. * Ich habe gelernt, wie man mit der Programmiersprache C++ arbeitet. * Ich habe gelernt, was ein Arduino ESP8266 ist. * Besonders gut hat mir gefallen, dass ich eine, mir bisher unbekannte Programmiersprache kennengelernt habe. |
| 1. **Tag 27.06.2017** | * Ich habe die Programmierung der Eingabe-Funktion durch die Konsole fortgeführt. * Dann habe ich nach der Mittagspause mit der anderen Praktikantin zusammen die Leiterplatte bestückt (Löten). * Dann haben wir einen Rundgang durch die Außenbereiche und einige Büros der Firma gemacht. * Ich habe gelernt, wie man lötet. * Ich habe gelernt, wie verschiedene Durchflussmessverfahren funktionieren. * Besonders gut hat mir gefallen, dass ich gelernt habe wie man lötet. |
| 1. **Tag 28.06.2017** | * Ich habe die Einlese-Funktion durch die Konsole fertig programmiert. * Dann habe ich mit der Funktion begonnen, welche eine Lampe auf einen gewünschten Helligkeitswert einstellt. Diese Funktion habe ich noch am selben Tag fertiggestellt (Dimm-Funktion). * Außerdem habe ich die Funktion, welche für die WLAN-Verbindung zuständig ist, erstellt. * Dann habe ich angefangen den Webserver auf den Arduino ESP8266 aufzusetzen. * Wir haben außerdem eine Rundführung durch die Leiterplattenfertigung gemacht. * Ich habe gelernt, wie man den Arduino ESP8266 mit dem WLAN verbindet. * Ich habe gelernt, wie man einen Webserver auf den ESP8266 laufen lässt. * Außerdem habe ich gelernt, wie Leiterplatten bestückt werden. * Besonders gut hat mir gefallen, zu sehen wie die Maschinen Leiterplatten bestücken. |
| 1. **Tag 29.06.2017** | * Ich habe den Webserver fertiggestellt. * Dann habe ich die Dimm-Funktion mit dem Webserver und dem WLAN verbunden. * Außerdem haben wir einen Rundgang durch das EMV-Prüflabor gemacht und dort gelernt was EMV ist. * Besonders interessant war zu lernen, was EMV ist. |
| 1. **Tag 30.06.2017** | * Ich habe die Website, über welche die Lampe gesteuert wird, erstellt. * Diese habe ich mithilfe von der http get Funktion mit dem Arduino Code verbunden. So ist es möglich die Lampe über die Website zu steuern. * Damit war das ursprüngliche Projekt abgeschlossen. * Dann habe ich mit der Präsentation begonnen, welche ich mit der anderen Praktikantin am Ende des Praktikums vor einigen Mitarbeitern halten sollte. * Ich habe gelernt, wie http Request funktioniert. |
| 1. **Tag 03.07.2017** | * Ich habe die Präsentation weiter vorbereitet. * Dann habe ich in meinem Programm unnötige globale Variablen durch lokale ersetzt. * Außerdem habe ich einen kleinen Fehler bei der Datenannahme über die Website behoben. |
| 1. **Tag 04.07.2017** | * Ich habe das Programm so umgeändert, dass die Website immer auf dem neusten Stand ist. * Dazu musste ich es möglich machen, dass der Slider immer auf der richtigen Stelle der Anzeige liegt. * Ich habe dazu gelernt, wie man mit der Funktion jQuery.getJSON() arbeitet. * Besonders gut hat mir gefallen, etwas neues über jQuery zu lernen. |
| 1. **Tag 05.07.2017** | * Ich habe eine Einschlaffunktion eingebaut, wodurch die Lampe langsam, nicht ruckartig, ausgeschaltet wird. * Ich habe einen Timer programmiert, mit welchem man einen Countdown für das Ausschalten der Lampe setzen konnte. * Wir haben einen Rundgang zur Projektmanagement-Abteilung der Firma gemacht. * Besonders gut hat mir gefallen auch etwas über Projektmanagement zu lernen. |
| 1. **Tag 06.08.2017** | * Ich habe den Programmcode verschönert (refaktoriert), also beispielsweise leere Zeilen gelöscht und kleinere Fehler beseitigt. |
| 1. **Tag 07.08.2017** | * Die andere Praktikantin und ich haben unser Projekt vor einigen Entwicklern und Führungskräften aus der Entwicklungsabteilung vorgestellt. |

# Wochenbericht 1

Am ersten Tag meines Betriebspraktikums sind wir in einen Meetingraum gegangen, dort hat unser Praktikumsbetreuer ein Projekt vorgestellt, welches wir während der Praktikumszeit bearbeiten sollen. Mit mir zusammen macht eine Schülerin eines Duisburger Gymnasiums ihr Praktikum. Dieses Projekt basiert darauf, dass wir mithilfe eines Arduino ESP8266 Controllers eine Lampe so programmieren, dass man sie über WLAN und eine Website steuern kann. Der Arduino ESP8266 ist ein kleiner Microcontroller, welcher über ein WLAN-Modul verfügt. Dadurch kann man diesen mit einem WLAN verbinden und darüber eine Website laufen lassen.

Daraufhin sind wir in das Büro unseres Praktikumsbetreuers gegangen, wo die andere Praktikantin und ich jeweils einen eigenen Schreibtisch mit Computer zugewiesen bekommen haben.

Als erstes mussten wir uns eigenständig mit der Entwicklungsumgebung des Arduino-Systems vertraut machen und anfangen eine Funktion zu programmieren, welche es möglich macht, die Lampe über die in das Arduino-Programm integrierte Konsole zu steuern. Ich habe dann damit begonnen eine Funktion zu schreiben, welche in die Konsole eingegebene Zahlen abspeichert. Das erste Problem bestand darin, dass die Zahlen nicht als richtige Zahlen, sondern als einfache Zeichen abgespeichert wurden. Darum konnte man mit diesen die Lampe, welche zur Steuerung richtige Zahlen benötigt, nicht einstellen. Um dieses Problem zu lösen habe ich die Zeichen mit der *atoi*-Funktion in Zahlen umgewandelt.

Am zweiten Tag sollten wir die Leiterplatten, auf welchen der ESP8266 befestigt ist, selber zusammenlöten. Dazu sind wir in die Ausbildungswerkstatt gegangen. Unter Anleitung durch den Ausbildungsleiter für Industrieelektroniker mussten wir als erstes eine kleine Lötübung machen. Diese bestand aus ein paar kleinen Bauteilen und einer Batterie. Wenn man die Leiterplatte mit der Batterie verbunden hat, hat eine kleine LED-Lampe angefangen zu blinken und wenn man dann ein kleines Potentiometer verstellt hat, haben sich die zeitlichen Abstände, in welchen die Lampe aus und an geschaltet wurde, entweder verkürzt oder verlängert. Nachdem wir diese Übung erfolgreich abgeschlossen hatten, haben wir mit der richtigen Leiterplatte begonnen. Diese bestand aus 14 Bauteilen, welche alle auf der Platine verlötet werden mussten. Diese Leiterplatte konnten wir dann am Ende des Praktikums auch mit nach Hause nehmen. Nachdem wir die Leiterlatte fertiggestellt hatten, sind wir wieder in unser Büro gegangen und haben weiter an der Programmierung gearbeitet.

Meiner Meinung nach ist das eine gute Einteilung der Tätigkeiten, da wir so nicht nur den Teil unseres Projektes bearbeiten mussten, bei dem wir am Computer arbeiten müssen, sondern auch den praktischen Teil übernommen haben.

Zwischendurch haben wir fast jeden Tag einen kleinen oder auch größeren Rundgang durch verschiedene Abteilungen der Firma gemacht. Am zweiten Tag sind wir durch die Außenbereiche des Firmengeländes gegangen. Dort haben wir zum Beispiel einen Hof gesehen auf welchem fertige Produkte in Lieferwagen verfrachtet wurden und dann an den Kunden geliefert wurden. Als wir dann nach drinnen gegangen sind, sind wir durch die Abteilung gegangen wo die Messgeräte hergestellt wurden. Dort wurden zum Beispiel, im Falle eine Durchfluss Messgeräts, das Rohr und die inneren Bestandteile angefertigt. Als wir wieder zurück gekommen sind haben wir weiter an dem Programm geschrieben.

Am dritten Tag habe ich dann schließlich die Konsolen-Steuerung abgeschlossen und habe sie so erweitert, dass man die Lampe nicht nur ein- und ausschalten konnte, sondern auch die Helligkeit bestimmen kann. Dies hat so funktioniert, dass man verschiedene Zahlen zwischen 0 und 1023 in die Konsole eingeben konnte. So ist bei der Zahl 0 ist die Lampe ausgeschaltet worden, während sie bei der Zahl 1023 auf die volle Helligkeit eingestellt wurde. Mit der Zahl 500 konnte man sie ungefähr auf halbe Helligkeit stellen. Die erforderliche Technik wird Pulsweitenmodulation genannt und ist in den Microcontroller integriert.

Anschließend habe ich das Programm um eine Funktion erweitert, welche eine Verbindung zwischen dem Arduino ESP8266 und einem von der IT-Abteilung eigens für das Praktikum eingerichtetes WLAN herstellt. Dazu habe ich zwei Konstanten angelegt, welche den WLAN-Namen (SSID) und das WLAN-Passwort beinhalten. So kann man relativ einfach das WLAN, welches benutzt wird, ändern. Wenn man also das Programm zuhause nutzen will, könnte man einfach bei den Konstanten den eigenen WLAN-Namen und das eigene WLAN-Passwort eintragen.

An diesem Tag haben wir eine Führung durch die Leiterplattenfertigung gemacht. In dieser Abteilung gibt es Maschinen, die große und komplizierte Leiterplatten, für die ein Mensch Stunden bräuchte, innerhalb von wenigen Minuten bestücken. Außerdem wurde uns dort auch die Abteilung gezeigt, welche die fertigen Leiterplatten testet. Dazu gibt es eine weitere Maschine, die jedes Bauteil einer Leiterplatte für sich ansteuert und so prüft ob es eventuell defekt ist (In-Circuit-Tester). Außerdem werden die Platten unter unterschiedlichen klimatischen Bedingungen getestet, beispielsweise unter Hitze, Kälte oder Feuchtigkeit.

Ich finde es gut, dass wir immer programmiert haben und dann aber auch ab und zu Rundgänge durch die Firma gemacht haben, wobei uns immer die entsprechenden Sachen von einem zuständigen Mitarbeiter erklärt wurden.

Am vierten Praktikumstag habe ich eine Funktion erstellt, welche einen Webserver auf dem Arduino Controller laufen lässt. Anschließend haben wir einen Rundgang durch das EMV-Prüflabor gemacht. Dort werden fertige Produkte getestet, bevor sie auf den Markt gebracht werden können. Hierbei geht es nicht, wie bei den Tests in der Leiterplattenfertigung darum, ob die Geräte funktionieren, sondern um die Sicherheit des Umfelds in dem die Geräte eingesetzt werden. Wenn ein Gerät den Test besteht, bekommt es das CE-Zeichen und kann auf den Markt gebracht werden. Es darf dann in ganzen europäischen Union verkauft werden. Wenn es nicht besteht, muss es zurück in die Entwicklung und die Fehler müssen behoben werden.

Am fünften Tag habe ich die Website, über welche die Lampe gesteuert wird, erstellt. Mithilfe von http Requests habe ich eine Verbindung zwischen dem Code der Website und dem des Arduino Controllers erstellt. Wenn also die Website Daten empfängt, werden diese an den Arduino Controller weitergeleitet, wodurch dieser mit den Daten reagieren kann. Wenn man z.B., mithilfe eines Sliders (jQuery-Control-Element auf der Website), eine 500 eingibt, schickt die Website den Wert an den Arduino (ESP8266) weiter und dort wird dann mithilfe einer Funktion die Helligkeit der Lampe auf den Wert 500 eingestellt.

Damit habe ich das eigentliche Projekt beendet.

Ich finde das dieses Projekt sehr sinnvoll war, da ich so die Programmiersprache C++ neu kennengelernt habe und gelernt habe wie das Prinzip von Arduino funktioniert. Da ich später beruflich etwas in dieser Richtung machen möchte, denke ich, dass dieses Projekt sehr sinnvoll gewesen ist. Außerdem habe ich so neue Sachen gelernt wie zum Beispiel, was EMV ist oder welche Messgeräte-Typen es gibt. Diese Woche war auch sehr interessant, da ich nicht nur programmiert habe, sondern auch die Firma kennengelernt und andere Abteilungen besucht habe.

# Wochenbericht 2

Zu Beginn der zweiten Woche habe ich den Auftrag bekommen, mit der anderen Praktikantin zusammen, eine Präsentation zu erstellen. In der Präsentation sollte das Projekt vorgestellt werden. Es sollte kurz erklärt werden, was der Arduino ESP8266 eigentlich ist und was die besonderen Funktionen sind. Außerdem sollten wir kurz auf den Begriff IoT eingehen.

IoT ist die Abkürzung von Internet of Things, auf Deutsch Internet der Dinge.

Es beschreibt, dass ein elektronischer Gegenstand über das Internet beziehungsweise über ein WLAN steuerbar ist. In diesem Fall ist es eine Lampe, welche man über das WLAN und eine integrierte Website steuern kann. Außerdem sollten wir auf besondere Stellen des Projekts eingehen, wie zum Beispiel welche Probleme es während der Programmierung gab. Dann sollten wir noch kurz erklären was ein http Request ist und wozu er benutzt wird. Wir haben zusätzlich dazu noch den Quelltext interessanter Funktionen als Screenshot eingefügt und ebenfalls als Struktogramm, also einen kurzen Plan wie diese Funktion aussieht und funktioniert.

Diese Präsentation mussten wir nämlich am Ende des Betriebspraktikums vor einigen Entwicklern und Führungskräften halten. Dazu gehörten auch Personen, die nichts mit der Entwicklung von Software zu tun haben. Konkret haben neben unserem Praktikumsbetreuer teilgenommen: der Ausbildungsleiter, eine Patent-Ingenieurin, die Gruppenleiterin für Analyse-Sensoren und der Gruppenleiter für Kommunikationsarchitektur.

Am sechsten Tag den Programmcode verschönert (refaktoriert), also beispielsweise Variablen, welche global, also für alle Funktionen verfügbar, definiert wurden, allerdings nur lokal, also nur in einer Funktion, genutzt werden, so umgeändert, dass sie direkt in dieser einen Funktion lokal definiert wurden.

Am siebten Tag habe ich das Programm so erweitert, dass, wenn sich die Helligkeitswerte ändern, sofort auch der Slider sich auf diesen Wert anpasst. Dazu musste ich eine Funktion einbringen welche die Website durchgehend aktualisiert ohne sie wirklich neu aufzurufen.

Am nächsten Tag habe ich noch eine Funktion hinzugefügt, welche bewirkt, dass die Lampe nicht von einem auf den anderen Moment ausgeschaltet wird, sondern innerhalb einer bestimmten Zeitspanne ausgeht (Fader). Außerdem habe ich einen Timer eingebaut, welcher die Lampe nach einer bestimmten Zeit automatisch ausschaltet. Diese Zeit kann man auf der Website einstellen. Im Programmcode kann die Zeiteinheit mittels einer Konstanten zwischen Minuten und Sekunden umgestellt werden.

An diesem Tag sind wir in die Projektmanagement-Abteilung gegangen und haben uns vom Abteilungsleiter erklären lassen, wofür genau seine Abteilung zuständig ist. Jede Person die dort arbeitet (Projektmanager) muss nämlich ein oder mehrere Projekte betreuen. Er muss also von Anfang bis Ende alles planen und dafür sorgen das die anderen Abteilungen ihre Arbeit machen.

Am neunten Tag habe ich eigentlich nur weiter den Programmcode refaktoriert.

Am letzten Tag meines Betriebspraktikums haben die andere Praktikantin und ich die Präsentation gehalten. Dazu sind wir wieder in den Meetingraum gegangen, wo dann auch weitere fünf Mitarbeiterinnen und Mitarbeiter gesessen haben. Wir haben am Ende der Präsentation außerdem noch eine Vorführung gemacht, in welcher wir die Funktionen des Projekts gezeigt und erklärt haben.

Meiner Meinung nach war die erste Woche des Praktikums produktiver, da wir aber schon früher als geplant das Projekt abgeschlossen hatten, musste der Praktikumsbetreuer sich neue Aufgaben ausdenken. Ich finde es allerdings sehr sinnvoll, dass wir am Ende noch eine Präsentation halten mussten, da dies, später, wenn man in dem Bereich arbeitet, auch immer gemacht wird. Deshalb ist dies im allgemeinem auch eine interessante Woche gewesen.

# Fazit

Ich finde das der Beruf des Software-Entwicklers / Informatikers interessant ist und, dass ich mir gut vorstellen kann, später auch einmal diesen ausführen zu wollen. Da ich mich generell für Computer und das Programmieren interessiere, war es auch abzusehen, dass sich mein Berufswunsch bestätigt. Demnach haben sich meine Erwartungen an das Betriebspraktikum erfüllt.

# Anhang

## Quelltext

#include <ESP8266WebServer.h>

#include <ESP8266WiFi.h>

ESP8266WebServer server(80); //Webserver wird festgelegt

//Änderbare Variablen

const int LIGHT\_PIN = 2; //Nummer des Pins

const int MAX\_WLAN\_CON\_TRYS = 3; //Maximale WLAN Verbindungsversuch Anzahl

const char SSID[] = "WLan-KI-Pro"; //WLAN Adresse

const char PASS[] = "sVAPHCmo"; //WLAN Passwort

const String TIME\_UNIT = "Minuten"; //Timer: Sekunden / Minuten

const unsigned long DIMMER\_INTERVAL\_MS = 0.1; //Dim Geschwindigkeit

const unsigned long HEART\_BEAT\_MS = 60000;

//Nicht änderbare Variablen

boolean consoleStatus = false;

int intervalTimeUnit = 0; //Timer Zeit

int timerTimeTimeUnit = 0; //Zeigt die übrige Timerzeit an

int dimValue = 0; //PWM Wert

int lampValue = 0;

int timeFactor = 0;

unsigned long startTimeMs = 0; //Zeit nachdem der Timer läuft

const int INPUT\_STRING\_LENGTH = 5; //Anzahl der möglichen Zeichen +1, die in die Konsole eingegeben werden können

const String HTML = "<!DOCTYPE html>"

"<html>"

"<head>"

"<title> Lampen Dimmer </title>"

"<meta charset = \"UTF-8\">"

"</head>"

"<script src = \"https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js\"></script>"

"<meta name = \"viewport\" content = \"width=device-width, initial-scale=1\">"

"<link rel = \"stylesheet\" href = \"https://code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css\">"

"<script src = \"https://code.jquery.com/jquery-1.12.4.js\"></script>"

"<script src = \"https://code.jquery.com/ui/1.12.1/jquery-ui.js\"></script>"

"<script>"

"$(document).ready(function(){" //Setzt den Slider auf den "dimValue" Wert, welcher auf dem Webserver geschpeichert ist

"$.getJSON(\"/getInfo\", function(result){"

"$.each(result, function(i, field){"

"$( \"#slider\" ).slider( \"option\", \"value\", field );"

"});"

"});"

"});"

"$(function() {"

"$(\"#slider\").slider({min: 0, max: 1023});"

"$(\"#slider\").on(\"slide\","

"function(event, ui) {"

"$.get(\"/dv?dv=\" + ui.value);"

"});"

"$(\"#on\").button().click(function() {$( \"#slider\" ).slider( \"option\", \"value\", 1023 ); $.get(\"/dv?dv=1023\"); });"

"$(\"#off\").button().click(function() {$( \"#slider\" ).slider( \"option\", \"value\", 0 ); $.get(\"/dv?dv=0\"); });"

"$(\"#send\").button().click(function(){$.get(\"/timer?timer=\" + $(\"#txt\").val());});"

"window.setInterval(function() {"

"$.getJSON(\"/getTimer\", function(result){"

"$.each(result, function(i, timer){"

"$(\"#time\").text(\"\"+timer);"

"});"

"});"

"}, 100);"

"timer.play();"

"});"

"</script>"

"<style>"

"div.header {"

"font-family: Arial;"

"font-size: 300%;"

"text-align: center;"

"background-color: #404040;"

"color: white;"

"padding: 3%;"

"border-radius: 3px"

"}"

"div.dimDiv {"

"border: 2px solid #404040; "

"padding: 3%;"

"margin-top: 2%;"

"border-radius: 3px"

"}"

"div.discription {"

"text-align: center;"

"border: 1px solid #404040;"

"border-radius: 3px"

"margin-top: 10%;"

"padding: 5%;"

"font-size: 100%;"

"font-family: Arial;"

"}"

"</style>"

"<body>"

"<div class = \"header\">"

"Lamp Control"

"</div>"

"<div class = \"discription\">"

"Mithilfe dieser Werkzeuge kann die Lampe an und aus geschaltet, sowie ihre Helligkeit und einen Auschalttimer eingestellt werden."

"</div>"

"<div class = \"dimDiv\" >"

"<div id = \"on\" style = \"margin-left: 30%;\">" //An Knopf

"on"

"</div> "

"<div id=\"off\">" //Aus Knopf

"off"

"</div>"

"</div>"

"<div class = \"dimDiv\">"

"<div id = \"slider\" style=\"width: 50%; margin: auto; margin-top: -1%; \">" //Dimmer Slider

"</div>"

"</div>"

"<div class =\"dimDiv\">"

"<input type=\"text\" id=\"txt\">"

"</input>"

"<div id=\"send\" style=\"margin-left: 2%;\">"

"Start Timer"

"</div>"

"<div id=\"time\" style=\"font-family: Arial; color: #626262; background-color: #f6f6f6; border-radius: 4px; margin-top: 1.5%; border: 1px solid #acacac; text-align: center;\">"

"</div>"

"</div>"

"<div id=\"div\">"

"</div>"

"</body>"

"</html>"; //HTML Code wird eingebunden

void setup() { //Wird zu Beginn einmal aufgerufen

Serial.begin(74880);

pinMode(LIGHT\_PIN, OUTPUT); //Setzt den Pin Modus auf Output

checkTimeFactor();

wlanConfig();

if(WiFi.status() != WL\_CONNECTED) {

} else {

webserverInit();

instructions();

}

}

void loop() { //Dauerschleife

if(consoleStatus == true) { //Konsoleneingabe wird bei fehlender WLAN Verbindung aktiviert

checkInput();

}

server.handleClient();

execTimer(); //Timer wird ausgeführt

execLampValue();

checkWlan();

heartBeat();

}

void checkTimeFactor() {

if(TIME\_UNIT == "Sekunden") {

timeFactor = 1000;

}

if(TIME\_UNIT == "Minuten") {

timeFactor = 60000;

} else {

Serial.print("Der Zeitfaktor ");

Serial.print(TIME\_UNIT);

Serial.println(" ist falsch. Du kannst nur zwischen Sekunden und Minuten wählen.");

while(timeFactor != 1000 && timeFactor != 60000) {

}

}

}

void checkWlan() {

if(WiFi.status() != WL\_CONNECTED) {

Serial.println("");

Serial.println("< Die WLAN-Verbindung wurde unterbrochen >");

}

}

void instructions() { //Gibt die WLAN Adresse und das Passwort an und sagt was man tun muss

IPAddress ip = WiFi.localIP();

Serial.println("");

Serial.println("");

Serial.print("Du kannst dich jetzt mit dem WLAN verbinden und auf die Lamp Control Website zugreifen");

//Serial.println("Oder Gebe in der Konsole einen Wert für die Lampe ein.");

Serial.println("");

Serial.println("");

Serial.println("");

Serial.println("|-----------INFOS-----------|");

Serial.println("");

Serial.print(" WLAN: ");

Serial.println(SSID);

Serial.println("");

Serial.print(" PASSWORT: ");

Serial.println(PASS);

Serial.println("");

Serial.print(" WEBSITE: ");

Serial.println(ip);

Serial.println("");

Serial.println("|-----------INFOS-----------|");

Serial.println("");

Serial.println("");

Serial.println("|------------LOG------------|");

Serial.println("");

}

void checkInput() {

//Prüft ob die, in die Konsole eingegebenen Werte, mit den Richtlinien übereinstimmen

if(Serial.available() < INPUT\_STRING\_LENGTH) { //Guckt ob es nicht zu viele Zeichen sind

if(Serial.available() > 0) { //Guckt ob es genug Zeichen sind

delay(10);

char inputString[INPUT\_STRING\_LENGTH];

input(inputString);

int value = atoi(inputString);

if(value >= 0 && value <= 1023) { //Guckt ob die Zahl in dem möglichem Bereich liegt

setLampValue(value);

}

else {

Serial.println("Die Zahl muss zwischen 0 und 1023 liegen!"); //Gibt Fehlermeldung aus

}

}

}

else {

Serial.println("Die Zahl muss zwischen 0 und 1023 liegen!"); //Gibt Fehlermeldung aus

}

while(Serial.read() >= 0); //Löscht den Inhalt des Eingabefeldes

}

void input(char \* inputString) { //Liest den Inhalt des Eingabefeldes aus

int inputCounter = 0;

for(int resetCounter = 0; resetCounter <= INPUT\_STRING\_LENGTH-1; resetCounter++) {

//Setzt die Counter zurück

inputString[resetCounter] = '\0';

}

while(Serial.available() > 0, Serial.available() < INPUT\_STRING\_LENGTH) {

//Wird ausgeführt solange etwas im Eingabefeld steht

char inputChar = Serial.read();

inputString[inputCounter] = inputChar;

inputCounter++;

if(inputCounter >= INPUT\_STRING\_LENGTH-1) {

//Beendet die Funktion, wenn die Maximale Anzahl an Zeichen erfasst wurde

inputString[INPUT\_STRING\_LENGTH-1] = '\0';

break;

}

}

if(Serial.available() < INPUT\_STRING\_LENGTH) {

inputString[inputCounter] = '\0'; //Beendet die Funktion

}

}

void wlanConfig() { //Die Verbindung mit dem WLAN wird hergestellt

int wlanConTrys = 0;

while(WiFi.status() != WL\_CONNECTED && wlanConTrys <= MAX\_WLAN\_CON\_TRYS - 1) {

delay(1000);

wlanConTrys++;

Serial.println("");

Serial.print(wlanConTrys);

Serial.print(". ");

Serial.println("Verbindungsversuch");

WiFi.begin(SSID, PASS); //WLAN Verbindung wird mit der Adresse und dem Passwort aufgebaut

int connectionCounter = 0;

while (WiFi.status() != WL\_CONNECTED && connectionCounter <= 20) {

//Wartet 10 Sekunden bis die WLAN Verbindung hergestellt worden ist

delay(500);

Serial.print(".");

connectionCounter++;

}

Serial.println("");

if (WiFi.status() != WL\_CONNECTED) {

//Bricht den Verbindungsversuch ab wenn die WLAN Verbindung nicht besteht

Serial.print("< Verbindung zum WLAN konnte nicht hergestellt werden");

if(wlanConTrys <= MAX\_WLAN\_CON\_TRYS - 1){

Serial.print(". Noch ");

Serial.print(MAX\_WLAN\_CON\_TRYS - wlanConTrys);

//Hat eine Bestimmte Anzahl von Verbindungsversuchen

Serial.print(" Verbindungsversuche");

}

Serial.println(" >");

}

else { //Teilt, bei WLAN Verbindung, die IP-Adresse mit

Serial.println("< Verbindung zum WLAN wurde hergestellt >");

}

}

if(WiFi.status() != WL\_CONNECTED) {

consoleStatus = true;

Serial.println("");

Serial.print("Die Verbindung zum WLAN, konnte, nach ");

Serial.print(wlanConTrys);

Serial.println(" Verbindungsversuchen, nicht hergestellt werden.");

Serial.println("Die Konsole kann jetzt zum Eingeben der Werte benutzt werden.");

Serial.println("");

Serial.println("");

Serial.println("|------------LOG------------|");

Serial.println("");

}

}

void webserverInit() { //Webserver wird Initialisiert

server.on("/", []() { //Zeigt die Website an

server.send(200, "text/html", HTML);

});

server.onNotFound([]() { //Zeigt, beim Aufruf einer nicht verhandenen Seite, eine Fehlermeldung an

server.send(404, "text/html", "404/Error");

});

server.on("/dv", []() { //Reagiert auf HTTP Get für die Lampe

String dvString = server.arg("dv");

dimValue = atoi(dvString.c\_str());

setLampValue(dimValue);

server.send(200, "text/html", ""); //Sendet zurück, dass er die Nachricht bekommen hat

});

server.on("/getInfo", []() { //Übergibt dem HTML Code die Variable "dimValue"

String valueJson = "{\"value\":";

valueJson += dimValue;

valueJson += "}";

server.send(200, "text/json", valueJson);

});

server.on("/getTimer", []() { //Übergibt dem HTML Code die verbliebende Timer Zeit

String timeJson = "{\"time\":";

timeJson += timerTimeTimeUnit;

timeJson += "}";

server.send(200, "text/json", timeJson);

});

server.on("/timer", []() { //Reagiert auf HTTP Get für den Timer

String timerString = server.arg("timer");

int timerInt = atoi(timerString.c\_str());

setTimer(timerInt);

server.send(200, "text/html", "");

});

server.begin(); //Webserver wird gestartet

Serial.println("< Webserver ist online >");

}

void setLampValue(int newValue) { //Setzt den Wert der Lampe

lampValue = newValue;

Serial.println("");

Serial.print("Die Value wurde auf ");

Serial.print(lampValue);

Serial.print(" gesetzt.");

if(timerTimeTimeUnit > 0) {

Serial.print(" Noch ");

Serial.print(timerTimeTimeUnit);

Serial.print(" ");

Serial.print(TIME\_UNIT);

Serial.print(" bis die Lampe ausgeschaltet wird.");

}

Serial.println("");

}

void execLampValue() { //Schaltet die Lampe mit "DIMMER\_INTERVAK\_MS" Geschwindigkeit aus

static int previousValue = 0;

unsigned long currentTimeMs = millis();

static unsigned long previousTimeMs = 0;

if(lampValue != previousValue) {

if(previousTimeMs + DIMMER\_INTERVAL\_MS < currentTimeMs) {

previousTimeMs = currentTimeMs;

int directionFactor = (previousValue > lampValue) ? 1 : -1;

previousValue = previousValue - 1 \* directionFactor;

analogWrite(LIGHT\_PIN, previousValue);

}

}

}

void heartBeat() { //Schaltet die Lampe mit "DIMMER\_INTERVAK\_MS" Geschwindigkeit aus

static int previousValue = 0;

unsigned long currentTimeMs = millis();

static unsigned long previousTimeMs = 0;

if(previousTimeMs + HEART\_BEAT\_MS < currentTimeMs) {

previousTimeMs = currentTimeMs;

Serial.println("< System ist online >");

}

}

void setTimer(int offTimerTimeUnit) { //Die Timer Zeit sowie die Start Zeit wird gesetzt

intervalTimeUnit = offTimerTimeUnit;

startTimeMs = millis();

Serial.println("");

Serial.print("Der Timer wurde auf ");

Serial.print(offTimerTimeUnit);

Serial.print(" ");

Serial.print(TIME\_UNIT);

Serial.println(" gesetzt.");

}

void execTimer() { //Der Timer wird ausgeführt

if(intervalTimeUnit != 0) {

unsigned long intervalMs = intervalTimeUnit \* timeFactor;

unsigned long currentMs = millis();

unsigned long timeCounterUntilMs = 0;

unsigned long timeCounterMs = currentMs - startTimeMs;

timeCounterUntilMs = intervalMs - timeCounterMs;

timerTimeTimeUnit = timeCounterUntilMs / timeFactor; //Restliche Zeit wird ausgerechnet

timerTimeTimeUnit = timerTimeTimeUnit + 1;

if(timeCounterMs >= intervalMs) { //Guckt ob die Zeit abgelaufen ist

Serial.println("");

Serial.println("Der Timer ist abgelaufen.");

intervalTimeUnit = 0;

timerTimeTimeUnit = 0;

setLampValue(0); //Lampe wird ausgeschaltet

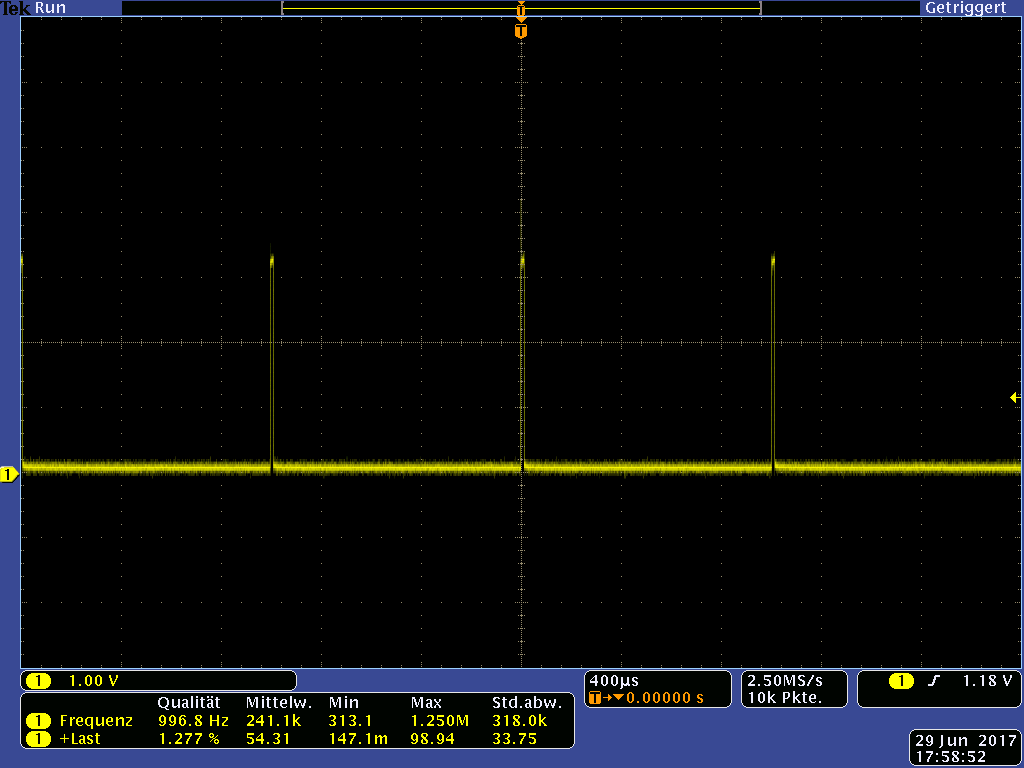
}

}

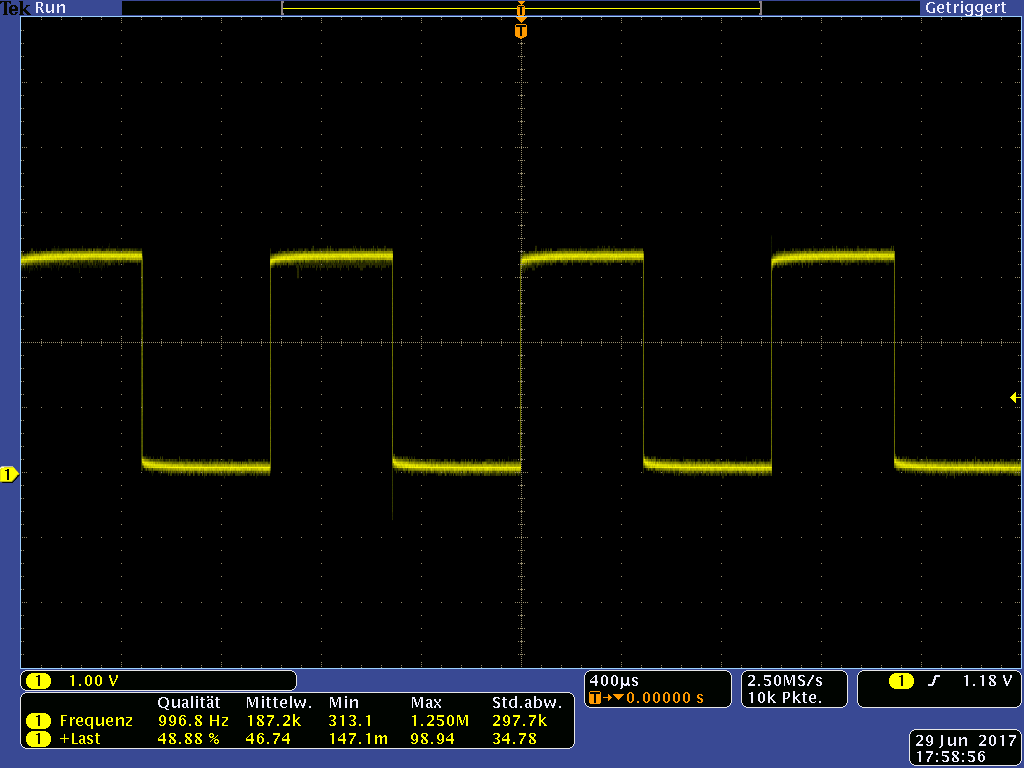
}

## Oszilloskop-Bilder des PWM-Signals

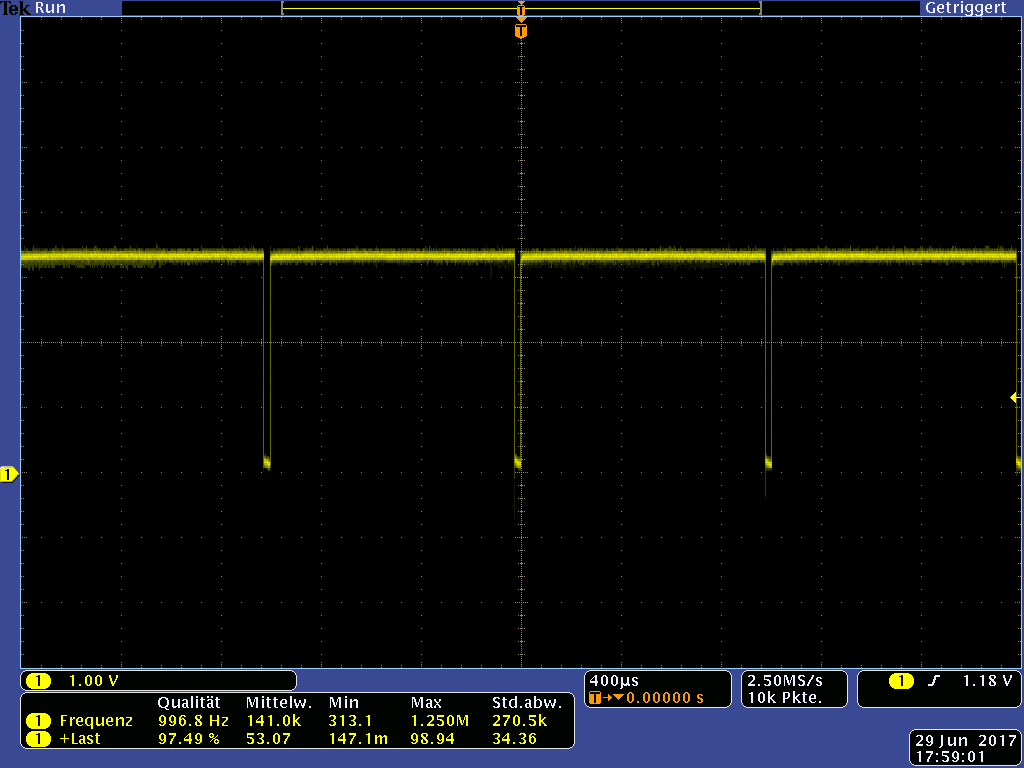
### Sehr geringe Helligkeit



### Mittlere Helligkeit



### Hohe Helligkeit



## 

## Fotos des Praktikumsprojekts

Ein Bild, das Tisch, drinnen, Boden enthält.

Mit sehr hoher Zuverlässigkeit generierte Beschreibung

Ein Bild, das Elektronik, Schaltkreis enthält.

Mit sehr hoher Zuverlässigkeit generierte Beschreibung