

Automatischer Luftfeuchtigkeitsregler für Terrarium

Planung und Bau

Milan Somora

České Budějovice, 2018
Zweite überarbeitete Ausgabe

Vorwort

Dieses Dokument ist eine maschinelle Übersetzung aus dem Tschechischen. Es wurde mithilfe eines Online-Übersetzungstools erstellt. Daher können einige Begriffe ungenau oder die Formatierung abweichend sein. Dennoch ist das Dokument in einer zufriedenstellenden und verwertbaren Form.

Das tschechische Originaldokument finden Sie auf GitHub unter folgendem Link:

<https://shorturl.at/hoewY>.

Inhalt

| | |
|---|-----------|
| 1. Einleitung | 2 |
| 2. Luftfeuchtigkeit im Terrarium | 3 |
| 2.1. Faktoren für die Aufrechterhaltung der Luftfeuchtigkeit..... | 3 |
| 2.2. Drei Zonen | 3 |
| 2.3. Automatische vs. manuelle Regulierung | 4 |
| 3. Arduino | 5 |
| 4. Beschreibung des Reglers | 7 |
| 4.1. Funktionen des Reglers..... | 7 |
| 4.2. Regelungsart | 9 |
| 5. Reglerentwurf | 12 |
| 5.1. Wasserpumpe..... | 12 |
| 5.2. Wasserspeicher..... | 13 |
| 5.3. Sensoren..... | 14 |
| 5.4. Düse..... | 17 |
| 5.5. Spannungsquelle | 17 |
| 5.6. Beschreibung der elektronischen Schaltung | 18 |
| 5.7. Quellcode für Arduino..... | 19 |
| 6. Anschluss und Einstellung des Reglers | 21 |
| 6.1. Anschluss..... | 21 |
| 6.2. Einstellung | 23 |
| 7. Montage des Reglers | 24 |
| 8. Fazit | 30 |
| Verwendete Literatur | 31 |
| Anhänge | 32 |
| A. Verzeichnis der verwendeten Bauteile | 33 |
| B. Elektrisches Schema des Reglers | 36 |
| C. Elektrisches Schema des Reglers mit PC-Netzteil | 38 |

D. Anschluss auf Steckfeld **40**

E. Quellcode für Arduino Nano **42**

1. Einführung

Ziel dieser Arbeit ist es, ein einfaches und kostengünstiges Gerät zu entwerfen und zu realisieren, das die Luftfeuchtigkeit im Terrarium automatisch reguliert. Diese Idee entstand aufgrund der Bedürfnisse von Terraristen, deren Rückmeldungen in verschiedenen Internetforen zu finden sind. Es muss nicht extra betont werden, dass die richtige Luftfeuchtigkeit für die meisten Reptilien notwendig ist und dass die manuelle Aufrechterhaltung dieses Niveaus in der Regel mehr Aufmerksamkeit erfordert. Die meisten Terraristen lösen dieses Problem mit einem handelsüblichen Blumensprüher und versuchen, regelmäßig zu gießen. Denjenigen, die mehr Zeit für andere Tätigkeiten benötigen, können automatische Feuchtigkeitsregler helfen, die jedoch eine Investition in Höhe von mehreren tausend Kronen bedeuten⁽¹⁾. Glücklicherweise gibt es eine Lösung, die mit ein wenig Fleiß für wenige hundert Kronen realisiert werden kann, nämlich mit Hilfe der Arduino-Plattform, einigen elektronischen Bauteilen und Teilen, die in der Garage oder auf dem Schrottplatz zu finden sind. Bei der Entwicklung des automatischen Reglers wurde auf einen niedrigen Anschaffungspreis und eine gute Verfügbarkeit der Komponenten geachtet. Die Einstellung und Messung erfolgte in einem Standardterrarium² mit einem Jemenchamäleon. Die Kapitel wurden mit Blick auf die praktische Anwendung und möglichst ohne unnötigen Ballast verfasst.

Der Autor würde sich freuen, wenn diese Arbeit nicht nur im Kreis der Terraristen Anwendung findet.

¹ Z. B. der Befeuchter Hagen Exo Terra Monsoon für 4.000 CZK (zum Zeitpunkt der Veröffentlichung dieser Arbeit) [4].

² Terrarium 30 x 40 x 80 cm mit 35-W-Glühlampe, UVB-Lampe und Belüftungsöffnungen.

2. Luftfeuchtigkeit im Terrarium

2.1. Faktoren für die Aufrechterhaltung der Luftfeuchtigkeit

„Die Menge an Wasserdampf ist zeitlich sehr variabel und unterscheidet sich auch von Ort zu Ort [5]. Mehrere Faktoren beeinflussen die Luftfeuchtigkeit im Terrarium. Vor allem die Anzahl und Größe der Belüftungsöffnungen spielen eine Rolle. Sind zu viele oder zu große Öffnungen vorhanden, entweicht der Wasserdampf schnell aus dem Terrarium und es muss häufiger bewässert werden.“

Ein weiterer Faktor ist die Anzahl und Leistung der Wärmequellen im Terrarium. Beispielsweise benötigt das Jemenchamäleon in unserem Terrarium mit den Maßen 30 x 40 x 80 cm eine Punktlampe mit einer Leistung von 35 W, um ein gewisses Maß an Wärmekomfort zu erreichen. Würden wir beispielsweise eine Halogenlampe mit Wolframfaden und einer Leistung von 100 W in unser Terrarium stellen, würde dies natürlich zu einer schnelleren Austrocknung der Luft führen.

Ein weiterer Faktor für die Aufrechterhaltung der Luftfeuchtigkeit ist ein angefeuchteter Substrat. Während unserer Messungen haben wir festgestellt, dass die Luftfeuchtigkeit bei angefeuchtem Substrat etwa doppelt so langsam sinkt (mehr dazu in 6.2).

Der letzte Faktor für die Aufrechterhaltung der Luftfeuchtigkeit ist die Bepflanzung des Terrariums mit Pflanzen oder Wurzeln. Der Autor ist der Meinung, dass bei einem fast leeren Terrarium die Aerosolwolke nur auf den Boden und die Wände des Terrariums sinkt, wo die Wassermoleküle aufgrund des Volumens des Terrariums ungleichmäßig verdunsten. Ist das Terrarium gleichmäßig bepflanzt, z. B. mit Laubpflanzen und Wurzeln, sollten die aus dem Aerosol abgeschiedenen Wassermoleküle gleichmäßig im gesamten Terrarium verdunsten.

2.2. Drei Zonen

Aus physikalischer Sicht könnte man ein Terrarium für Chamäleons in drei Zonen unterteilen – eine obere, eine mittlere und eine untere. Die höchste Luftfeuchtigkeit herrscht unten, da das versprühte Aerosol schwerer ist als Luft und daher nach unten sinkt. Von dort verdampfen die Wassermoleküle und steigen in Form von Wasserdampf nach oben. Mit steigender Temperatur in der zweiten und dritten Zone erwärmen sich die Dämpfe noch mehr, gehen allmählich in den gasförmigen Zustand über, wodurch die Luft trockener und leichter wird und daher leicht durch die obere Belüftungsöffnung aus dem Terrarium entweichen kann. Logischerweise kann man davon ausgehen, dass die niedrigste

Die Luftfeuchtigkeit ist oben in der Nähe von Wärmequellen und am oberen Lüftungsöffnungen am höchsten. Aus dem oben Gesagten folgt, dass der beste Ort für die Anbringung des zukünftigen Feuchtigkeitssensors die mittlere Zone des Terrariums ist.

In diesem Kapitel sollte aufgezeigt werden, dass nicht jedes Terrarium für eine automatische Luftfeuchtigkeitsregelung geeignet ist und dass bestimmte Bedingungen erfüllt sein müssen.

2.3. Automatische vs. manuelle Regulierung

Die automatische Regulierung hat im Terrarium physikalisch denselben Effekt wie die manuelle Regulierung. Der einzige Unterschied besteht darin, dass die automatische Regulierung präziser und bequemer ist (die Luftfeuchtigkeit muss nicht kontrolliert werden, es muss lediglich Wasser nachgefüllt werden).

3. Arduino

Der vorgeschlagene Regler nutzt die Entwicklungsplatine Arduino Nano (Abb. 3.1). Was ist Arduino? Arduino ist eine Entwicklungsplattform, die mehrere Arten von Platinen mit Mikroprozessoren¹ der Firma Atmel anbietet, die mit der Sprache Wiring² beliebig programmiert werden können. Jedes Board ist neben dem Mikroprozessor mit weiteren elektronischen Schaltungen für verschiedene Zwecke (HDMI, SATA, Ethernet, Audioanschlüsse usw.) ausgestattet und kann mit Zusatzmodulen (sogenannten Shields) verbunden werden, die seine Funktionalität noch erweitern (z. B. WLAN, Relais, eine Reihe von Sensoren usw.) [11].

Auf den meisten Platinen ist ein Wandler integriert, der es Arduino ermöglicht, mit anderen Geräten zu kommunizieren. Derzeit ist der Wandler für die Kommunikation über eine serielle USB-Schnittstelle am weitesten verbreitet. Platinen ohne Wandler wurden für möglichst kleine Abmessungen entwickelt und müssen mit einem externen Wandler (z. B. Arduino Mini) programmiert werden. Schaltpläne und Software für Arduino sind kostenlos online verfügbar.

Die Arduino-Plattform ist heute so weit verbreitet, dass Platinen und Zubehör in fast allen Elektronikfachgeschäften erhältlich sind. Weitere Platinen sind beispielsweise Arduino Uno, Mega 2560, Due und andere.

Arduino Nano

Dieser Platinen-Typ wurde aufgrund seiner Eigenschaften für den Regler ausgewählt: geringe Abmessungen, ausreichende Anzahl von Anschlüssen und USB-Konverter.

Die Platine hat Abmessungen von 18 x 45 mm. Sie enthält einen ATmega328-Prozessor mit einer Taktfrequenz von 16 MHz und einen Speicher von 32 kB. Was die Anschlüsse betrifft, so verfügt sie über 8 analoge, 14 digitale und davon 6 PWM-Anschlüsse. Jeder Anschluss kann als Ein- oder Ausgang konfiguriert werden. Jeder Port kann mit einem Strom von maximal 40 mA betrieben werden.

Die Logikspannung beträgt 5 V und die Versorgungsspannung für die Platine liegt im Bereich von 7 bis der 12 V. Der Stromverbrauch der Platine beträgt 19 mA. Detailliertere Parameter finden Sie unter

[10].

¹ Genauer gesagt Mikrocontroller. Ein Mikrocontroller (MCU) enthält neben einem Prozessor auch weitere Schaltkreise wie -Timer, Speicher, Ein-/Ausgangsschaltungen usw., sodass er eigenständig Funktionen ausführen kann [9].

² sehr ähnliche Syntax wie in der Sprache C/C++

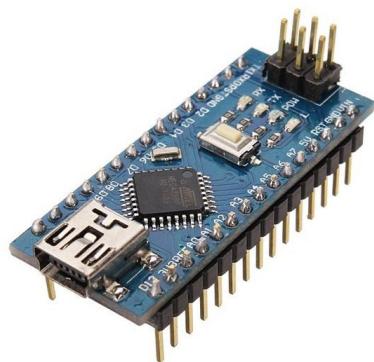


Abbildung 3.1.: Arduino Nano v3.0

Arduino Nano wird in drei Ausführungen verkauft – Original, Klon³ (Platine von einem anderen Hersteller) mit/ohne Stromversorgung. Das Original-Arduino wirkt sich natürlich auf den Preis aus. Für den Regler wurde eine Platine ohne Stromversorgung gekauft, da die Platine anders platziert werden musste.

³ Es ist nicht notwendig, ein Original-Arduino zu kaufen. Klone werden von den meisten Elektronikgeschäften in Tschechien verkauft, und da Arduino eine Open-Source-Plattform ist, handelt es sich um eine legale Vorgehensweise. Ein Beispiel ist der Klon Arduino Nano V3.0 ATmega328 16M 5V CH340G, der in den Geschäften von GM Electronic erhältlich ist.

4. Beschreibung des Reglers

Das Gerät zur Regulierung der Luftfeuchtigkeit im Terrarium besteht aus einem Regler mit LCD-Display, einem DHT11-Sensor, der Feuchtigkeit und Temperatur misst, einem Wasserstandssensor, einem Wasserbehälter, einer Pumpe und einem Netzteil.

Der Regler wurde so konzipiert, dass die Luftfeuchtigkeit im Terrarium den eingestellten Mindestwert nicht überschreitet. Der Mindestfeuchtigkeitswert wurde standardmäßig auf 60 % eingestellt. Der Terraristen kann diesen Feuchtigkeitsgrenzwert je nach den gehaltenen Tieren anpassen, indem er einen Wert im Quellcode für Arduino Nano ändert. Dieser Vorgang wird in [5.7](#) näher beschrieben.

Wenn die Luftfeuchtigkeit unter 60 % fällt, versprüht der Regler Wasser aus der Düse und die Luftfeuchtigkeit steigt schnell auf etwa 80 % an (dieser Intervall kann ebenfalls neu eingestellt werden). Danach dauert es einige Dutzend Minuten, bis die Luftfeuchtigkeit wieder unter 60 % fällt.

Es handelt sich hierbei nur um eine Zweipunktregelung mit Hysteresis [\[6\]](#), die manchen vielleicht unzureichend erscheint, aber eine präzise Regelung auf einen konstanten Wert ist nicht immer wünschenswert, wie weiter unten erläutert wird. In den folgenden beiden Unterkapiteln wird der Regler näher beschrieben.

4.1. Funktion des Reglers

Nach dem Einschalten des Reglers erscheint die Startmeldung „Feuchtigkeitsregler“. Nach etwa zwei Sekunden überprüft der Regler die Funktion des DHT11-Sensors, den Wasserstand im Behälter sowie die Luftfeuchtigkeit und -temperatur im Terrarium. Wenn alles in Ordnung ist, leuchtet die grüne LED auf und auf dem LCD-Display werden folgende Statusinformationen angezeigt:



Wenn die Luftfeuchtigkeit unter 60 % fällt, leuchtet die rote LED auf, der Regler beginnt 6 Sekunden lang Wasser zu versprühen¹ und überprüft anschließend 3 Minuten lang die Luftfeuchtigkeit im Terrarium. Auf dem Display wird

¹ können zusätzlich eingestellt werden

werden folgende Informationen angezeigt:



Dieser Schritt war notwendig, da es eine Weile dauert, bis sich das versprühte Aerosol im gesamten Terrarium verteilt hat. Würde die Luftfeuchtigkeit beispielsweise alle 30 Sekunden überprüft, würde der Regler zu häufig befeuchten, da sich ein Anstieg der Luftfeuchtigkeit nicht sofort bemerkbar macht. Nach einem solchen Eingriff könnte die Luftfeuchtigkeit auf bis zu 100 % ansteigen.

Kehren wir nun zu den ursprünglich eingestellten drei Minuten zurück. Wenn die Luftfeuchtigkeit im Terrarium auch nach drei Minuten die eingestellten 60 % nicht überschreitet, versprüht der Regler eine weitere Dosis. In der Praxis hat sich gezeigt, dass der Regler in der Regel 1-2 Dosen versprüht und die Luftfeuchtigkeit dann auf etwa 80 % ansteigt. Danach dauert es einige Dutzend Minuten, bis die Luftfeuchtigkeit wieder unter 60 % sinkt (siehe Grafik in Abb. 4.2). Diese Schritte werden wiederholt, bis der Wasserstand im Vorratsbehälter unter den Mindeststand sinkt.

Wenn der Wasserstand unter den Mindeststand fällt, piept der Regler dreimal, leuchtet die gelbe LED auf, stoppt die Zerstäubung und wartet 5 Minuten lang auf die Nachfüllung von Wasser. Auf dem Display wird folgende Information angezeigt:



Diese Funktion des Reglers ist sehr wichtig, da sie die Pumpe schützt, die sonst bei Leerlauf zerstört würde. Wenn innerhalb von fünf Minuten kein Wasser nachgefüllt wird, schaltet der Regler wieder ein und auf dem Display erscheint die gleiche Warnung.

Wenn die Kommunikation zwischen dem DHT11-Sensor und dem Arduino aus irgendeinem Grund unterbrochen wird (schlechter Kontakt, Verschleiß usw.), unterbricht der Regler seinen Betrieb und auf dem Display wird folgende Meldung angezeigt:





Diese Funktion ist ebenfalls wichtig, da der Regler ohne den Sensor DHT11 nicht funktionieren kann. In einem solchen Fall muss der Sensor überprüft oder gegebenenfalls durch einen neuen ersetzt werden. Anschließend muss der Regler aus- und wieder eingeschaltet werden.

4.2. Regelungsart

Wie bereits angedeutet, handelt es sich hierbei lediglich um eine Zweipunktregelung mit Hysteresis [6], die bewusst gewählt wurde. Bei Verwendung einer fortschrittlicheren Regelung, z. B. einer Proportionalregelung, die proportional auf die Regelabweichung⁽²⁾ reagiert wäre es möglich, die Feuchtigkeit auf einem konstanten Wert zu halten, jedoch in unserem Fall auf Kosten eines hohen Wasserverbrauchs und einer deutlich höheren Aktivität des Reglers. Die Ursache dafür wären die Lüftungsöffnungen im Terrarium, durch die nach jeder Bewässerung ein großer Teil des Aerosols aus dem Terrarium entweichen würde und der Regler aufgrund dieser Störgröße ineffizient und unwirtschaftlich arbeiten würde – siehe die deutlichen Feuchtigkeitsabfälle im Diagramm in Abb. 4.1 nach jeder Bewässerung, die sich aus unseren Messungen ergeben haben. Eine proportionale Regelung wäre wahrscheinlich sinnvoller, wenn ein Terrarium ohne Belüftungsöffnungen verwendet würde. Dies würde jedoch den Bedingungen widersprechen, die der Terrarienhalter für das Tier gewährleisten muss.

Die einfache Zweipunktregelung in unserem Regler sorgt dafür, dass die Luftfeuchtigkeit in einem bestimmten Bereich bleibt, z. B. zwischen 60 und 80 %, und nie unter 60 % fällt, was für Terraristen super wichtig ist. Auf der Grundlage der Informationen aus [5] kann ferner festgestellt werden, dass die Zweistuifenregelung eine natürlichere Art der Feuchtigkeitsregulierung gewährleistet, da auch in der Natur nicht 24 Stunden am Tag an einem Ort genau 70 % Feuchtigkeit herrschen. Der Grund dafür ist, dass „die Feuchtigkeit zeitlich sehr variabel ist und auch von Ort zu Ort variiert“ [5].

² Der Regler würde die Sprühdauer proportional zur aktuellen Feuchtigkeit im Terrarium verkürzen und versuchen, die Feuchtigkeit im Terrarium genau bei 80 % (zum Beispiel) zu halten.

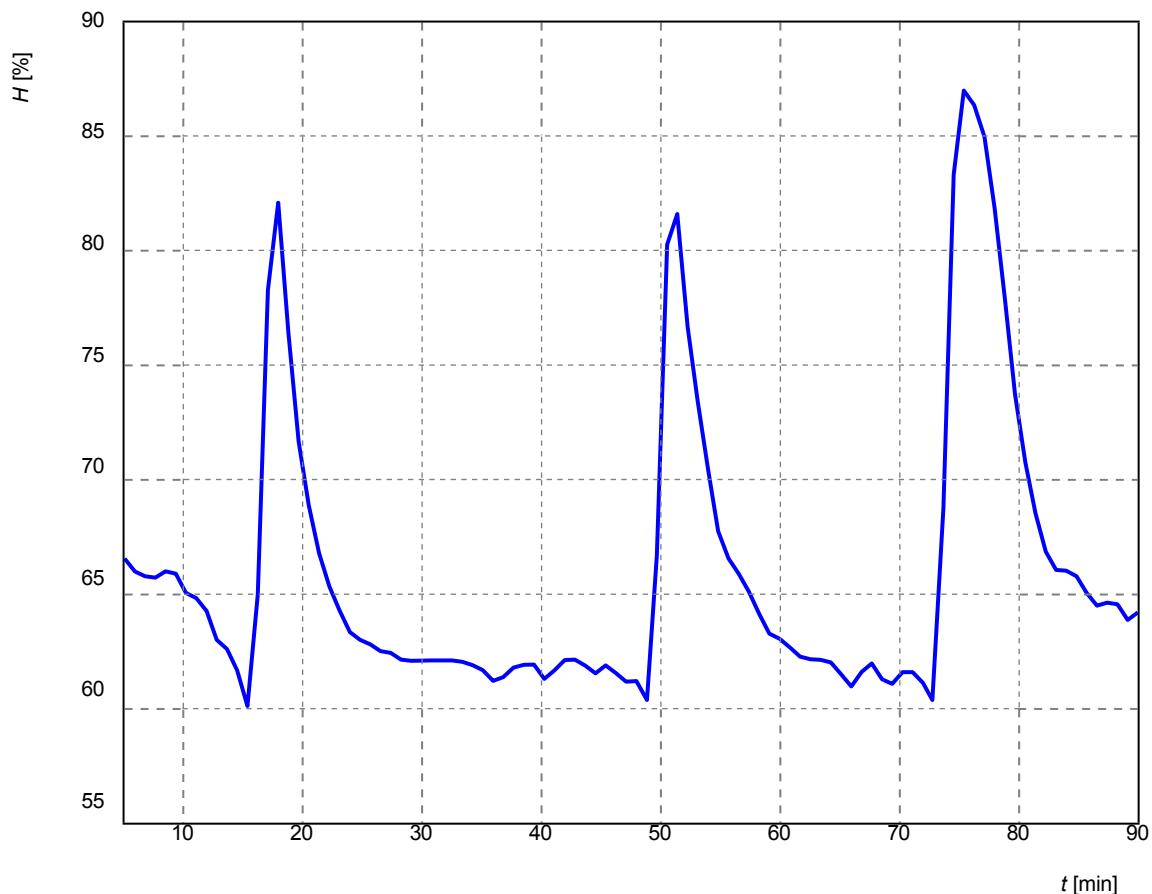


Abbildung 4.1: Messung der Luftfeuchtigkeit im Terrarium mit zweistufiger Regelung – Änderung der Luftfeuchtigkeit über die Zeit

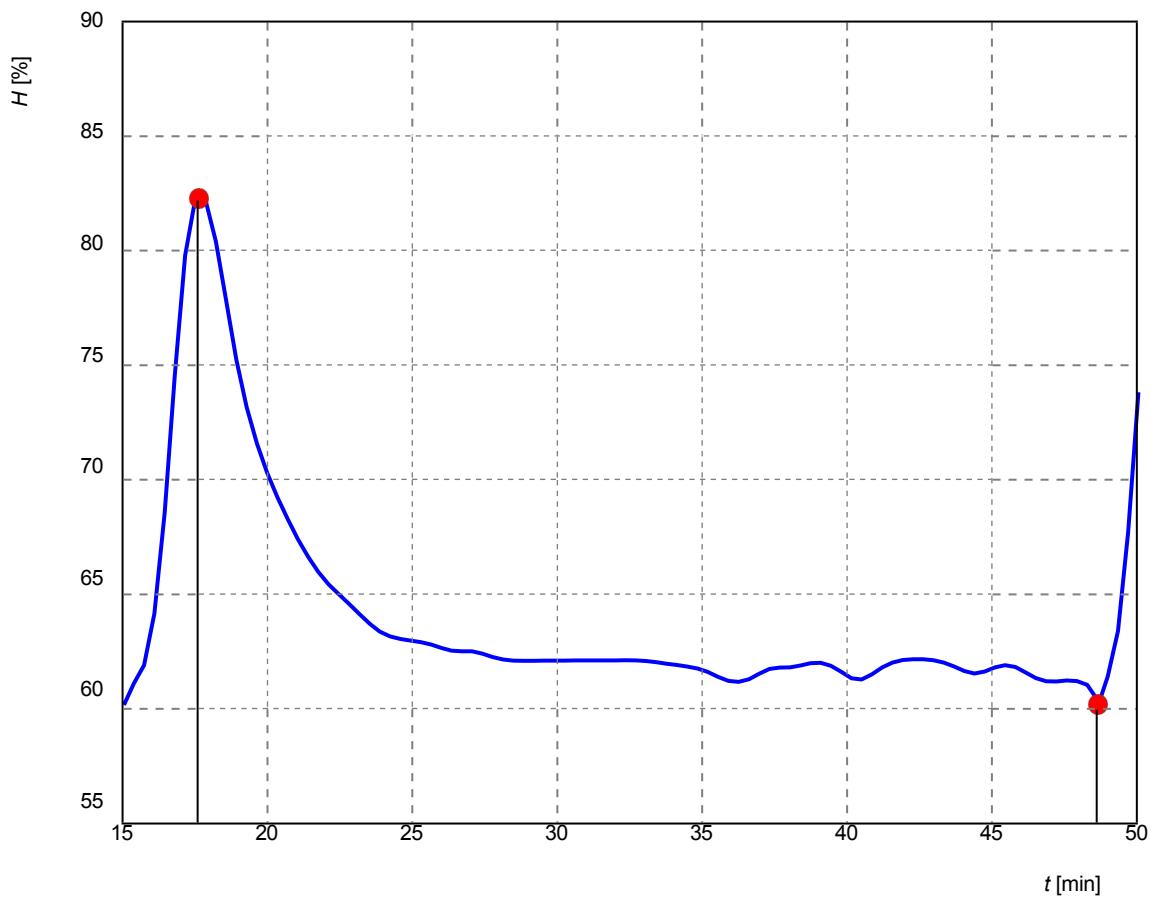


Abbildung 4.2: Messung der Luftfeuchtigkeit im Terrarium mit zweistufiger Regelung – Zeit bis zum Abfall der Luftfeuchtigkeit nach der Befeuchtung des Terrariums

5. Entwurf des Reglers

Im vorigen Kapitel wurde beschrieben, dass die Vorrichtung zur Regelung der Luftfeuchtigkeit aus einem Regler mit LCD-Display, zwei Sensoren, einem Wassertank, einer Pumpe und einem Netzteil besteht. Die folgenden Kapitel befassen sich näher mit diesen und weiteren Teilen und beschreiben einzeln die Anforderungen, die erfüllt sein müssen, um die Funktionsfähigkeit des gesamten Regelkreises zu gewährleisten.

Der elektronische Schaltplan des Reglers ist in Anhang **B**, die Liste der verwendeten Bauteile in Anhang **A** aufgeführt. Da die Funktion der Schaltung vor der Endmontage standardmäßig auf einer Lötfreifeldplatine getestet wird, ist in Anhang **D** noch die Schaltung auf einer Lötfreifeldplatine dargestellt, aus der auch die Schaltung des gesamten Regelkreises ersichtlich ist.

5.1. Wasserpumpe

Die Pumpe ist einer der wichtigsten Bestandteile des Reglers. Für eine korrekte Wasserverteilung im Terrarium ist eine Pumpe mit ausreichendem Druck und ausreichender Förderleistung erforderlich. Außerdem muss sie klein sein und mit einer niedrigen Gleichspannung betrieben werden (wegen Störungen und aus Sicherheitsgründen⁽¹⁾). Eine solche Pumpe ist leider nicht im Handel erhältlich und wäre zu teuer. In diesem Fall ist es einfacher, einen Autoverwerter aufzusuchen und nach einer Scheibenwaschpumpe mit Autostecker zu suchen.

Autopumpen haben zwar einen höheren Stromverbrauch (in Ampere) und können nicht selbst ansaugen, aber sie sind leistungsstark, langlebig, können mit Gleichspannung von 12 V betrieben werden und sind in Autoverwertungshöfen für wenige Euro erhältlich. Wir haben eine Pumpe der Marke Peugeot mit einem Stromverbrauch von 2,3 A für 200 CZK gefunden (Abb. 5.1).

Es ist wichtig zu beachten, dass der Schaltkreis unseres Reglers für eine Pumpe ausgelegt ist, die einen maximalen Strom von ² 3 A aufnimmt. Je geringer der Strom, desto weniger belastet die Pumpe natürlich den Schaltkreis des Reglers. Aus diesem Grund wird empfohlen, die Pumpe vorab zu messen und bei höherem Stromverbrauch durch eine andere zu ersetzen.

¹ Beispielsweise könnte das Schalten einer Pumpe mit einer Wechselspannung von 230 V im Arduino-Schaltkreis aufgrund elektromagnetischer Störungen Probleme verursachen, zumal eine Spannung von 230 V bereits als gefährlich gilt [8].

² Der Grund dafür ist der integrierte Spannungsstabilisator LM350T, der mit maximal 3 A arbeiten kann [3].

Abschließend ist darauf hinzuweisen, dass die Pumpe nicht über einen längeren Zeitraum leer (ohne Wasser) laufen darf, da sie sonst zerstört wird. Der Regler ist auf diese Situation vorbereitet und schaltet die Pumpe bei Unterschreiten des Mindestwasserstands automatisch ab und wartet auf den Bediener, bis Wasser in den Vorratsbehälter nachgefüllt wird (siehe weiter unten).



Abbildung 5.1: Autopumpe der Marke Peugeot mit angeschlossenem Autokonnektor, Aquarienschlauch, Düse, Gartenschlauch und Schneckenschnalle

5.2. Wassertank

Als Wasservorratsbehälter wurde ein 5-Liter-Kanister mit einer ausgeschnittenen Öffnung zum Nachfüllen von Wasser verwendet (Abb. 5.2). Da die Pumpe kein Wasser ansaugen kann, muss sie an der Stelle angeschlossen werden, an der die Flüssigkeit austritt. In unserem Fall war es am besten, ein Loch in den Boden des Kanisters zu bohren, aus dem das Wasser direkt in die Pumpe fließen konnte (Abb. 5.6).

Die Auslassöffnung musste mechanisch mit einer Gummidichtung abgedichtet werden (Silikon und andere Klebstoffe lösten sich immer ab³), und die Pumpe musste mit Verbindungsstücken (Hydrauliknippel, Stück Gartenschlauch, Schneckenbeschrauben usw. – siehe Abb. 5.3) an den Kanister angeschlossen werden. Dieser Teil muss ausgeklugelt sein, damit kein Wasser austreten kann. Daher ist es besser, ein Fachgeschäft aufzusuchen und passende Verbindungsstücke zu besorgen.

³ Der Grund dafür war das nicht haftende und weiche Material des Kanisters, die Schwerkraft der befestigten Pumpe, die auf die Verbindung wirkte, usw.



Abbildung 5.2: Wassertank, Pumpe, Düse und Leitungen für den Wasserstandssensor



Abbildung 5.3.: Verbindungselemente zum Anschluss der Pumpe an den Wassertank

5.3. Sensoren

Der Regelkreis enthält zwei Sensoren. Der erste Sensor vom Typ DHT11 misst die Luftfeuchtigkeit mit einer Genauigkeit von $\pm 5\%$ ⁴ und die Temperatur⁵ mit einer Genauigkeit von $\pm 2^\circ\text{C}$ [1]. Der zweite Sensor besteht aus zwei Leitern und dient zur Erkennung eines Absinkens des Wasserstands im Behälter unter den Mindeststand.

⁴ Die Genauigkeit von $\pm 5\%$ mag manchen gering erscheinen, aber das Ziel ist nicht, die Luftfeuchtigkeit mit höchster Genauigkeit zu erreichen, sondern einen Bereich zu finden, der für die Tiere im Terrarium geeignet ist. Ein solcher Bereich kann beispielsweise zwischen 60 und 80 % liegen.

⁵Die Temperaturmessung wurde verwendet, da der Sensor dies ermöglichte. Obwohl es sich um einen praktischen Wert handelt, wird die Temperatur nur in der Umgebung gemessen, in der der Sensor platziert ist (voraussichtlich in der mittleren Zone des Terrariums).

Feuchtigkeits- und Temperatursensor

Der Sensor DHT11 sollte zwischen den Anschlüssen 1– nd 2 einen sogenannten Pull-up-Widerstand mit einem Wert von $5\text{ k}\Omega$ haben, um mit Arduino kompatibel zu sein ([2], Abb. 5.4, Anhang B). Am besten kaufen Sie den Sensor DHT11 in Form eines sogenannten Shields (mehr dazu in 3), der bereits einen Widerstand enthält und auch mit einem Anschlussstecker ausgestattet ist. Der dritte Pin wird bei diesem Sensor nicht angeschlossen.

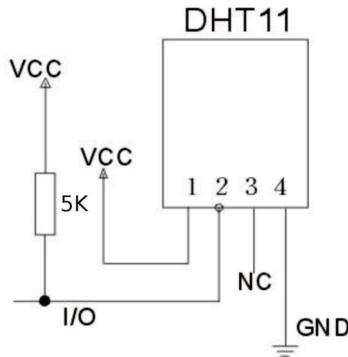


Abbildung 5.4: Sensor DHT11 mit $5\text{-k}\Omega$ -Widerstand [1]

Der Sensor muss im Terrarium geeignet platziert werden, idealerweise in der mittleren Zone des Terrariums. Er sollte nicht in der Nähe einer Wärmequelle (Heizlampe) platziert werden, wo die Luftfeuchtigkeit am niedrigsten ist, aber auch nicht am Boden des Terrariums, wo die Luftfeuchtigkeit am höchsten ist (mehr dazu in 2.2).

Wasserstandssensor

Der Wasserstandssensor informiert den Bediener, wenn kein Wasser mehr vorhanden ist, und schützt gleichzeitig die Pumpe vor Beschädigungen (wie in 5.1 beschrieben). Er besteht aus zwei Leitern, deren Stromkreis durch Wasser verbunden ist. Der erste Leiter liefert eine Gleichspannung von 5 V, der zweite Leiter misst diese Spannung.

Beide Leiter sollten abgeschirmt sein. Dies dient dazu, dass keine Spannung in den nicht unter Spannung stehenden Leiter induziert wird. Andernfalls würde der Sensor auch bei Wassermangel einen ausreichenden Wasserstand melden, was zur Zerstörung der Pumpe führen würde.

Abgeschirmte Leiter sind im Schaltplan mit einer gestrichelten Linie gekennzeichnet (Anhang B). Die Abschirmung beider Leiter muss auf Nullpotenzial GNDD geerdet sein, damit ein eventuell auftretendes Nicht-Null-Potential auf Null abfällt. Das zweite Ende der Abschirmung darf nicht in Wasser eintauchen, da es sonst zu einem Kurzschluss in diesem Stromkreis kommt – es reicht aus, ein Stück Kabel zu isolieren und die Abschirmung der Leiter über dem Vorratsbehälter mit Zinn zu verbinden (Abb. 5.6). Sollte die Wasserstandserkennung dennoch fehlerhaft sein, wird

Die abgeschirmten Leiter müssen über die gesamte Länge des Kabels voneinander getrennt sein, damit sie sich nicht gegenseitig magnetisch beeinflussen können.

Im Behälter müssen beide Leiter so befestigt werden, dass sie nicht „schwimmen“. Bewährt hat sich die Verwendung von zwei festen Kupferdrähten, die mit einem Kabelbinder befestigt werden. Diese Drähte dienen als Elektroden, und die Kabel werden mit Zinn an sie angelötet (Abb. 5.5 und 5.6).



Abbildung 5.5: Befestigung des Füllstandssensors mit zwei festen Kupferdrähten

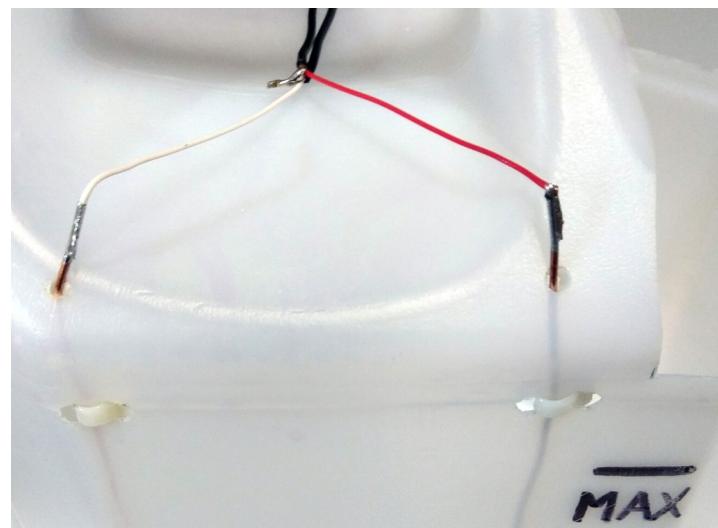


Abbildung 5.6: Anschluss des abgeschirmten Kabels an den Füllstandsensor. Die Abschirmung ist mit Zinn verbunden.

5.4. Düse

Auch wenn es nicht so aussieht, kann die Düse zum Versprühen von Wasser das teuerste Teil sein. Du kannst sie z. B. in Terraristikgeschäften kaufen, aber sie kostet normalerweise mehr als 200 CZK. Es empfiehlt sich, einen handelsüblichen Handsprüher für Blumen zu kaufen und die Düse daraus zu entfernen (Abb. 5.2). Der Sprüher wird dadurch natürlich unbrauchbar, aber er kostet nur etwa 50 CZK. Am besten eignen sich Sprüher, bei denen sich der Aerosolnebel aus der Düse regulieren lässt⁽⁶⁾.

Die Düse mit der Pumpe wird mit einem handelsüblichen Aquarienschlauch mit einem Durchmesser von etwa 6 mm verbunden. Je kürzer, desto besser, da eine längere Leitung den Druck am Auslass der Düse verringert.

Die Düse sollte am besten im oberen Bereich des Terrariums angebracht werden, um eine Wolke nach unten zu erzeugen. Eine Wolke nach oben zu erzeugen, ist nicht empfehlenswert, da ein Teil davon durch die obere Belüftungsöffnung aus dem Terrarium entweichen würde, was wiederum zu einer höheren Aktivität des Reglers führen würde. Es ist auch nicht sinnvoll, in Richtung einer heißen Glühbirne zu sprühen.

Die Düse wurde mit einer Kunststoffhalterung im Terrarium befestigt, die mit transparentem Isolierband am Glas befestigt wurde.

5.5. Stromquelle

Der Regelkreis muss von einer Stromquelle gespeist werden, die eine Gleichstrom-Leerlaufspannung im Bereich von 15 bis 30 V⁷ hat und mindestens 4 A liefern kann. Diese Parameter müssen eingehalten werden. Außerdem muss die Stromquelle, ähnlich wie bei der Pumpe, nicht zu groß sein und über eine gute Isolierung verfügen, die vor gefährlichen Berührungen und Spritzwasser schützt.

Der Bau einer solchen Stromquelle ist in der Regel aufwendig und erfordert fundierte Elektronikkenntnisse.

Am einfachsten ist es, eine Netzstromquelle aus einem ausrangierten Notebook zu beschaffen. Solche Stromquellen sind gleichgerichtet, gut gefiltert, ihre Ausgangsspannung liegt zwischen 18 und 25 V, sie sind klein und sicher. Interessanterweise können einige trotz ihrer geringen Abmessungen 5 A und mehr liefern. Ein weiterer Vorteil ist die Ausführung mit einem Netzkabel zum Anschluss an eine Steckdose.

Wer kein solches Netzteil kaufen möchte und den integrierten Schaltkreis LM350T⁸ vermeiden möchte, kann ein Computernetzteil mit einer Leistung von 250 W (oder mehr) verwenden. Dank dieser Stromversorgung muss keine Pumpe mit geringerem Stromverbrauch gesucht werden, da Computer-Netzteile einen Strom von 10 A und mehr bewältigen können. Ein weiterer Vorteil ist ihre gleichgerichtete Ausgangsspannung im Leerlauf, die genau 12 V beträgt.

⁶ Das kann ein Problem sein, da heutige Zerstäuber eine statische Düse haben und einige keine feine Aerosolwolke erzeugen. Es ist daher eine sorgfältige Auswahl erforderlich.

⁷ Je näher die Spannung an 15 V liegt, desto besser.

⁸ Obwohl es sich um einen hervorragenden Spannungsstabilisator handelt, begrenzt er durch seine Eigenschaften den Einsatz der Pumpe auf maximal 3 A und erwärmt sich bei höherer Belastung (daher muss er über einen Kühler zur Ableitung der Verlustwärme verfügen).

Nachteile des PC-Netzteils sind jedoch seine größeren Abmessungen, der laute Lüfter, die offene Bauweise und die Tatsache, dass es in der Regel mit einem externen Schalter ausgestattet werden muss, um separat eingeschaltet werden zu können. Ein Schaltplan für den Regler unter Verwendung dieses Netzteils ist in Anhang C enthalten. Dieser Schaltung wurde jedoch keine größere Aufmerksamkeit gewidmet.

5.6. Beschreibung der elektronischen Schaltung

Die folgende Beschreibung basiert auf dem Schaltplan in Anhang B.

Die Netzversorgung des Reglers wird über den Schalter S1 geschaltet. Außerdem enthält die Schaltung zwei Spannungsstabilisatoren – den Stabilisator L7808, der 8 V für das Arduino Nano liefert, und den Stabilisator LM350T, der 12 V für die Pumpe liefert. Die übrigen Teile des Schaltkreises werden vom Arduino auf der Grundlage des in Anhang E beschriebenen Algorithmus gesteuert, d. h. das Relais, das die Pumpe schaltet, die Piezo-Sirene und die LEDs.

Der Nullpotenzial für die von Arduino gesteuerten Schaltkreise⁹ wurde von den anderen Schaltkreisen getrennt, um mögliche Störungen zu vermeiden (d. h. Arduino hat GNDD, alles andere GND).

Beide Leitungen für den Wasserstandssensor müssen aufgrund von Störungen abgeschirmt und geerdet sein. Im Gerätegehäuse sollte darauf geachtet werden, dass ihre Anschlüsse so weit wie möglich voneinander entfernt sind und nicht zusammen mit unter Spannung stehenden Leitungen verlegt werden (siehe weiter unten).

Der Stabilisator LM350T muss mit einem Kühlkörper ausgestattet sein, falls er sich übermäßig erwärmt (siehe Teileliste in Anhang A). Je größer, desto besser. In den Instrumentenkasten über dem Kühlkörper müssen mehrere Löcher gebohrt werden, um die Abwärme abzuleiten.

Die Einstellung der Spannung von 12 V für die Pumpe mit dem Trimmer R2 sollte am besten erst nach dem Einbau aller Bauteile, bei abgeklemmter Pumpe und mit einem digitalen Voltmeter vorgenommen werden.

Während der Wert des Trimmers R2 $5\text{ k}\Omega$ beträgt, wird sein tatsächlich eingestellter Wert um $2,1\text{ k}\Omega$ schwanken. Eine ähnliche Situation gilt auch für den Trimmer R8, der besser mehrfach gedreht werden sollte, damit der Kontrast des Displays genau eingestellt werden kann. Die Pins DB0 - DB3 werden nicht an das LCD-Display angeschlossen. Die anderen Pins werden wie in der Abbildung in

Anhang B angeschlossen. Der Schalter S2 dient zum Ausschalten des LCD-Displays, dessen Hintergrundbeleuchtung manche Personen im Dunkeln stören könnte. Durch das Ausschalten wird gleichzeitig der Stromverbrauch des Reglers um ca. 2 W reduziert. Bei ausgeschaltetem LCD Das Display lässt sich mithilfe von drei LEDs (weitere Details siehe unten) leicht ablesen.

Die Piezosirene erzeugt für ihre Größe einen relativ lauten Ton. Wenn Ihnen die Lautstärke zu hoch ist, empfehlen wir, die Sirene mit Isolierband zu überkleben. Wenn Ihnen die Lautstärke zu leise ist, empfehlen wir, den Vorwiderstand R6 auf $50\text{ }\Omega$ zu reduzieren.

⁹ Schaltungen, die mit einer Spannung von 5 V gesteuert werden

Der Quellcode für Arduino sollte erst am Ende hochgeladen werden.

5.7. Quellcode für Arduino

Der Quellcode, der für die Steuerung des geregelten Systems¹⁰ geschrieben wurde, ist in Anhang E enthalten und wurde für Arduino Nano v3.0 optimiert. Er kann über ein USB-Kabel¹¹ auf das Arduino geladen werden. Dazu ist jedoch die Anwendung Arduino IDE erforderlich, die kostenlos auf der offiziellen Website von Arduino heruntergeladen werden kann: <https://www.arduino.cc/>.

Damit alles reibungslos funktioniert, müssen alle Bibliotheken, die in den Zeilen 2 bis 5 definiert sind, in die Arduino IDE importiert werden. Alle können auf Github heruntergeladen werden (einschließlich und der Quell -Code): <https://github.com/milansomora/regulator-vlhkosti>.

In den Zeilen 38 bis 41 befinden sich globale Variablen, deren Werte je nach gewünschtem Verhalten der Luftfeuchtigkeitsregelung im Terrarium (abhängig vom Terrariumtyp) beliebig geändert werden können. Es handelt sich um die folgenden vier Werte:

- **Einstellung der minimalen Luftfeuchtigkeit** – die Standardeinstellung ist 60 %. Wenn die Luftfeuchtigkeit unter diesen Wert fällt, beginnt der Regler mit der Bewässerung.
- **Startzeit der Pumpe** – die Standardeinstellung ist 6 000 ms, d. h. 6 s¹². Dies ist die Zeit, die die Sprühdauer bei der Bewässerung des Terrariums bestimmt. Es werden mindestens 4 s empfohlen, da sonst die Regelung möglicherweise nicht optimal ist.
- **Feuchtigkeitskontrollzeit** – der Ausgangswert beträgt 180 000 ms, d. h. 3 min. Dies ist die Zeit, die der Regler wartet, bis der Feuchtigkeitsgrad nach dem Sprühen auf einen bestimmten Wert angestiegen ist. Es werden mindestens 2 min empfohlen, da der Regler sonst ineffizient und unwirtschaftlich arbeitet.
- **Zeit für die Überprüfung der Wasserauffüllung** – der Ausgangswert beträgt 300 000 ms, d. h. 5 min. Wenn das Wasser im Vorratsbehälter aufgebraucht ist, wartet der Regler 5 min und überprüft dann, ob Wasser vom Bediener nachgefüllt wurde. Aus der Praxis werden 5–10 min empfohlen.

In den Zeilen 178 und 210 kann die Zeichenfolge (Text) weiter angepasst werden, die zwischen Anführungszeichen „...“ geschlossen sein muss. Weitere im Code enthaltene Zeichenfolgen, die nicht auf dem LCD-Display angezeigt werden, dienen der Serviceanalyse und können nur über die serielle Schnittstelle gelesen werden.

¹⁰ Das geregelte System ist in unserem Fall das Terrarium

¹¹ genauer gesagt in den MCU ATMega328P, den Arduino Nano enthält

¹² Arduino arbeitet mit einer Einheit von Millisekunden

Der Code kann natürlich auch für andere Platinen angepasst werden, z. B. für Arduino Uno oder Arduino Mega. In der Regel reicht es aus, die Pin-Nummern in den Zeilen 18 bis 31 zu ändern. Die übrigen Teile des Codes sollten nur von erfahrenen Programmierern geändert werden.

6. Anschluss und Einstellung des Reglers

Die Netzquelle wird mit einer Wechselspannung von 230 V versorgt, die als gefährlich gilt [8], daher ist besondere Vorsicht geboten!

6.1. Anschluss

Der Regler sollte vorab auf einer Lötfreiplatine angeschlossen werden, damit die Parameter seines Schaltkreises eingestellt und die Gesamtfunktion vor der endgültigen Montage überprüft werden können (Abb. 6.1).

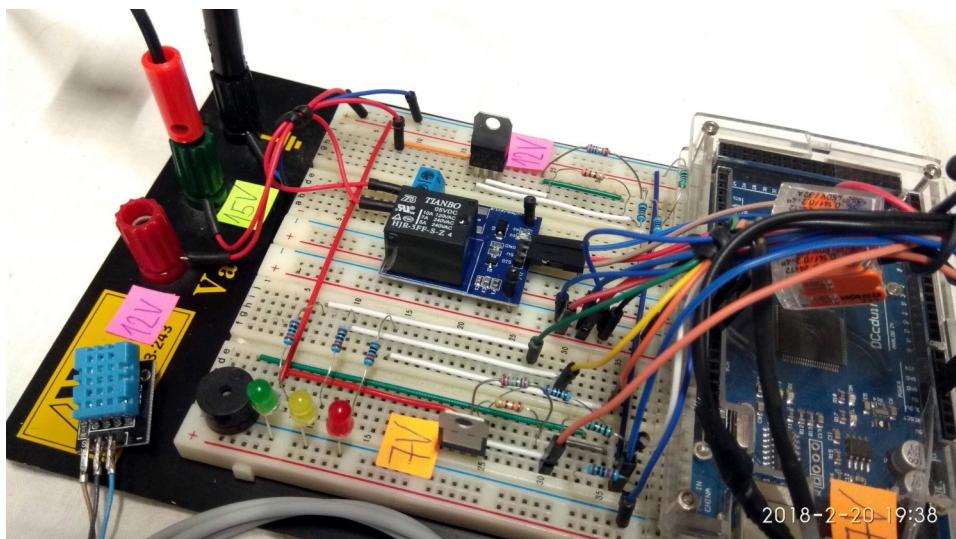


Abbildung 6.1: Beispiel für den Anschluss an einer lötfreien Platine mit Arduino Mega
(Anfangsphase der Entwicklung)

Der Reglerkreis wird gemäß dem Schaltplan in Anhang B oder D angeschlossen und gemäß den Anweisungen in Unterkapitel 5.6 auf Seite 18 vorgegangen.

Achten Sie auf die Nummerierung der Anschlüsse der integrierten Schaltungen. Beim Schaltkreis LM350T ist beispielsweise der mittlere Anschluss der Ausgang, unabhängig davon, ob er die Nummer 2 oder 3 hat (die Hersteller verwenden sowohl die Nummerierung 1-2-3 als auch 1-3-2). Die richtige Kennzeichnung der Anschlüsse für den Schaltkreis LM350T ist in Abb. 6.3 und für den Schaltkreis L7808 in Abb. 6.2 dargestellt.

Besondere Aufmerksamkeit muss auch der Einstellung der Spannung von 12 V für die Pumpe mit dem Trimmer R2 und der Einstellung des Kontrasts des LCD-Displays mit dem Trimmer R8 gewidmet werden. Nach dem ersten Anschließen des Displays ist in der Regel nichts zu sehen.

Außerdem darf der Kühlkörper nicht vergessen werden, der mit Wärmeleitpaste und einer Schraube mit Mutter am integrierten Schaltkreis LM350T befestigt wird. Der Wassertank, die Pumpe, die Düse und die Sensoren werden ebenfalls gemäß den Anweisungen in den vorherigen Unterkapiteln vorbereitet und angeschlossen. Zuletzt wird der Quellcode auf das Arduino geladen, der Tank mit Wasser gefüllt, die Düse (vorerst) in den Tank getaucht, damit kein Wasser herausspritzt, und der Regler durch Anschließen des Netzteils an die Steckdose aktiviert.

Auf dem LCD-Display sollte die Startmeldung „Automatische Feuchtigkeitsregulierung“ erscheinen und nach etwa zwei Sekunden die aktuellen Werte im Terrarium und der Status des Reglers. Wenn die Luftfeuchtigkeit unter 60 % liegt, beginnt der Regler mit der Bewässerung. Ist dies nicht der Fall, blasen Sie mit einem Föhn auf den Sensor DHT11. Umgekehrt erhöhen Sie die Luftfeuchtigkeit, indem Sie mehrmals tief auf den Sensor atmen. Wenn alles wie erwartet funktioniert, können Sie mit der Einstellung des Reglers für das Terrarium fortfahren.

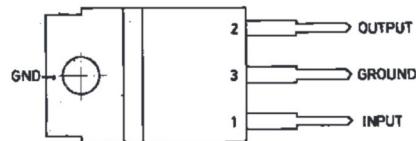


Abbildung 6.2: Bezeichnung der Anschlüsse für den integrierten Schaltkreis L7808 [7]

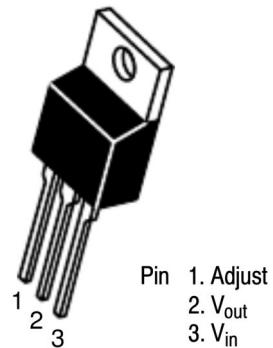


Abbildung 6.3.: Bezeichnung der Anschlüsse für den integrierten Schaltkreis LM350T [3]

6.2. Einstellung

Vor der Einstellung des Reglers müssen die Ausgangsbedingungen im Terrarium sichergestellt werden. Zunächst muss der Substratboden im Terrarium ausreichend befeuchtet werden, die Beleuchtung eingeschaltet, die Tür geschlossen und etwa eine halbe Stunde gewartet werden, damit sich das Klima im Terrarium ausgleichen kann. Schließlich wird der Regler eingeschaltet und der Bewässerungsvorgang in Abhängigkeit von der Zeit beobachtet. Wenn der Regler zu häufig bewässert, sollten zunächst die Düse und der DHT11-Sensor an eine andere Stelle versetzt werden, um bessere Bedingungen für die Bewässerung und Messung zu gewährleisten, wie in 5 beschrieben. Wenn der Regler weiterhin schwankt, versuchen Sie, die Sprühzeit zu verlängern, aber stellen Sie nicht mehr als 10 s ein, um eine starke Erwärmung des LM350T-Schaltkreises zu vermeiden.

Eine weitere Ursache kann das große Volumen des Terrariums sein, wodurch es länger dauert, bis sich die Feuchtigkeitsänderung bemerkbar macht. In diesem Fall empfiehlt es sich, die Feuchtigkeitskontrolle beispielsweise auf 5 Minuten zu verlängern. Wenn auch diese Maßnahmen nicht helfen, empfiehlt es sich, die Eignung des Terrariums für die automatische Regulierung zu überprüfen, wie in 2. Wenn hingegen die Luftfeuchtigkeit nach dem Besprühen zu hoch ist, kann die Sprühzeit um z. B. 2 Sekunden verkürzt werden.

Die praktische Erfahrung hat gezeigt, dass ein richtig eingestellter Regler etwa alle 30 Minuten eine Dosis sprüht (siehe Grafik in Abb. 4.2). Dies ist ein Richtwert, der den gemessenen Werten in unserem Terrarium mit den Abmessungen 30 x 40 x 80 cm entspricht. In einem größeren Terrarium wird der Zeitraum wahrscheinlich länger sein¹.

Aus dem Diagramm in Abb. 4.1 ist ersichtlich, wie der Regler funktioniert: Wenn die Luftfeuchtigkeit unter 60 % fällt, bewässert der Regler einmal, die Luftfeuchtigkeit steigt innerhalb von 3 Minuten auf etwa 80 % und sinkt dann innerhalb von etwa 30 Minuten auf natürliche Weise wieder unter 60 %. Dieser Zyklus wiederholt sich, bis das Wasser im Vorratsbehälter aufgebraucht ist.

¹ Denn in einem größeren Terrarium ist der Substrat auf eine größere Fläche verteilt und die Wärmequellen trocknen die Luft langsamer als in einem kleinen Terrarium.

7. Montage des Reglers

In dieser Phase wird davon ausgegangen, dass der Regler betriebsbereit und somit für den endgültigen Einbau in den Instrumentenkasten bereit ist.

Dieses Kapitel enthält nur kurze Informationen, da eine ausführliche Darstellung den Rahmen dieser Arbeit sprengen würde. Der Autor ist davon überzeugt, dass jeder diesen Teil nach seinen eigenen Vorstellungen umsetzen wird (anderer Typ des Instrumentenkastens, andere Anordnung der Bauteile usw.), daher enthält dieses Kapitel nur unsere Empfehlungen und Fotos des Reglers in unserer Ausführung als Anregung.

1. Es wird empfohlen, ein Instrumentengehäuse mit Schutzart IP54 zu verwenden. Der Grund dafür ist nicht, dass diese Schutzart den Inhalt des Gehäuses vor Spritzwasser schützt¹ ([8]), sondern dass ein solches Gehäuse stabiler und gut zu bearbeiten ist.
2. Einfacher ist es, eine vorgebohrte Universal-Leiterplatte zu verwenden, die nicht geätzt werden muss und nur wenige Euro kostet (Abbildungen 7.1, 4.1, 4.2).
3. Damit die Platine und das Arduino Nano nicht in der Box „herumfliegen“, können sie an einigen Stellen mit Sanitärsilikon festgeklebt werden. Silikon ist hitzebeständig und im Falle einer Wartung lassen sich die Platine und das Arduino leicht wieder ablösen. Heutzutage ist es kein Problem mehr, eine kleine Tube mit etwa 50 ml Inhalt zu kaufen. Siehe auch das aufgeklebte LCD-Display in Abb. 7.6.
4. In den Kasten müssen über dem Kühler mehrere Löcher gebohrt werden, um die Abwärme abzuleiten. Die Position der Löcher für die LEDs, Schalter, Stecker und das LCD-Display kann jeder nach eigenem Ermessen wählen.
5. Der Gerätebehälter sollte doppelt so groß sein wie in Abb. 7.2 und 7.3, damit einige Teile des Reglers nicht zu dicht beieinander liegen. Der fertige Regler in Abb. 7.7 funktioniert zwar und sieht aufgrund seiner Abmessungen ansprechend aus, jedoch traten während der Montage Probleme mit elektromagnetischen Störungen auf, die nicht einfach zu beheben waren. Ein größerer Kasten sollte auch eine bessere Verteilung der Verlustwärmegewährleisten. Daher wird Folgendes empfohlen:

¹ In den Kasten werden mehrere Löcher gebohrt, die Anschlussklemmen haben ebenfalls keine höhere Schutzart und das LCD-Display wird ebenfalls nicht abgedeckt. Daher ist eine Schutzart IP54 in diesem Fall sinnlos.

- a) Die Klemmleiste für den „InT-Füllstandssensor“ sollte separat in der Box so weit wie möglich von der bestückten Leiterplatte entfernt (z. B. auf einer kleinen speziellen Universalverbindung) angebracht werden. Das abgeschirmte Kabel, das von dieser Klemmleiste zum Wassertank führt, sollte so verlegt werden, dass es nicht unter Spannung steht.
- b) Das Arduino Nano sollte ebenfalls so weit wie möglich von der bestückten Leiterplatte entfernt platziert werden.
- c) Die drei Kabel, die vom Arduino zu den drei LEDs führen, sollten ebenfalls separat verlegt werden.

Die Maßnahmen mögen kompliziert erscheinen, aber es geht nur um die richtige Anordnung der Komponenten im Inneren des Gehäuses.

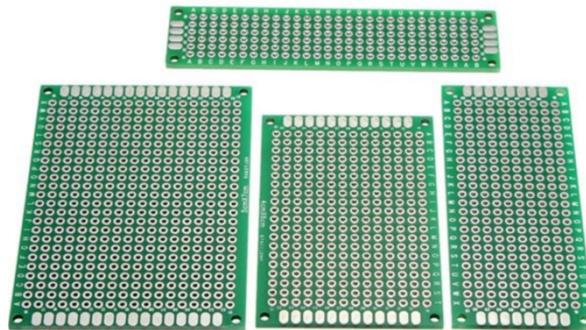


Abbildung 7.1: Vorgebohrte Universalplatten für die Herstellung von Leiterplatten Abb. 7.8 zeigt ein Foto der fertigen Baugruppe zur Regulierung der Luftfeuchtigkeit im Terrarium.



Abbildung 7.2: Gerätebox – Ansicht 1

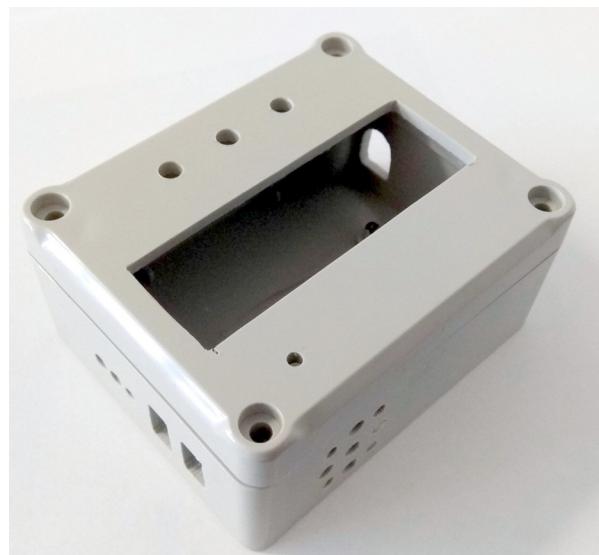


Abbildung 7.3: Instrumentenkasten – Ansicht 2

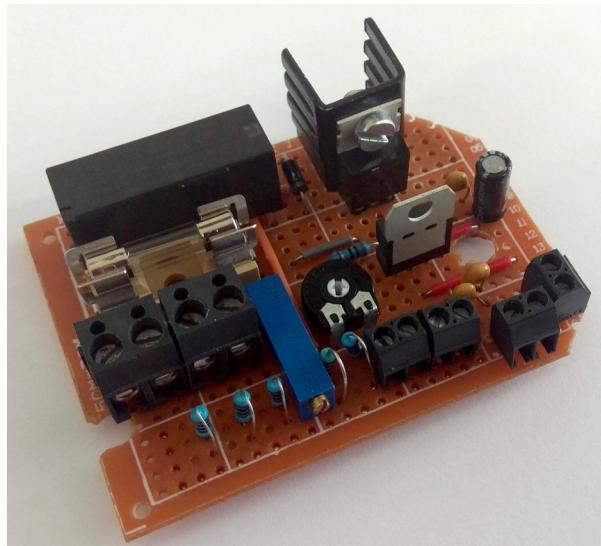


Abbildung 7.4: Bestückte Universalplatine

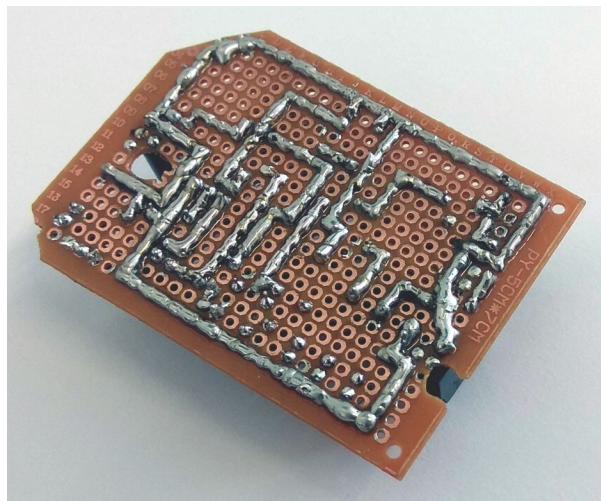


Abbildung 7.5: Erstellte Leiterplatten

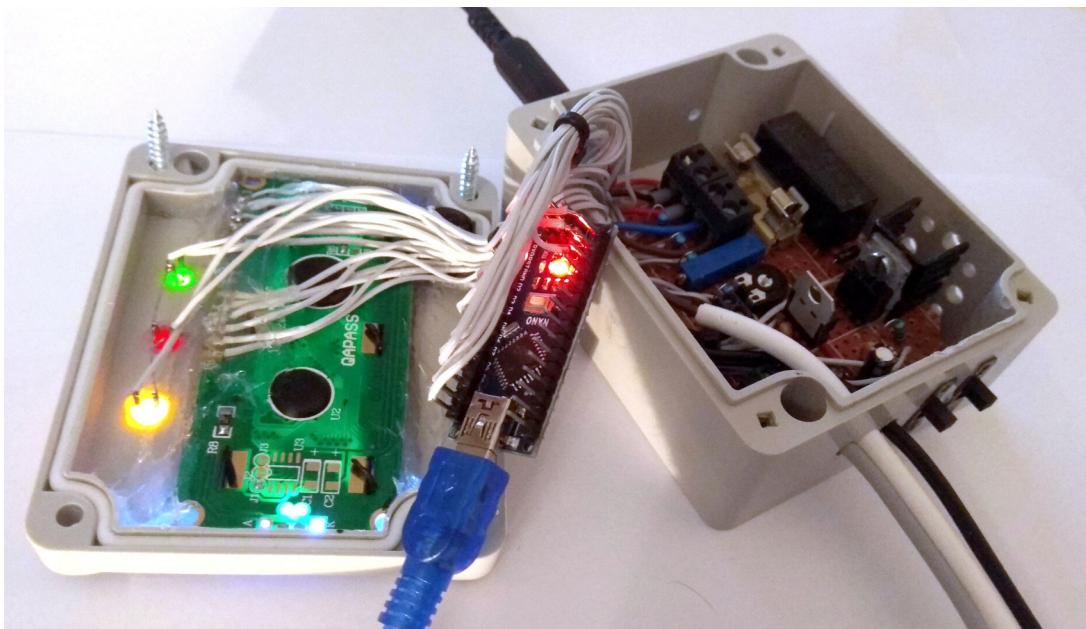


Abbildung 7.6: Laden des Quellcodes auf den Arduino Nano



Abbildung 7.7: Fertiger Regler



Abbildung 7.8: Fertige Baugruppe zur Luftfeuchtigkeitsregelung

8. Fazit

Das Ziel dieser Arbeit war es, ein einfaches und kostengünstiges Gerät zu entwerfen und zu realisieren, das die Luftfeuchtigkeit im Terrarium im gewünschten Bereich hält.

Das Gerät besteht aus einem Regler mit LCD-Display, zwei Sensoren, einem Wassertank, einer Pumpe und einem Netzteil.

Während der Testphase traten einige Probleme auf, die zu einer Verbesserung der Wasserversorgung und -verteilung, einer Optimierung der elektronischen Schaltung, einer Verbesserung der Betriebssicherheit und insbesondere zu einer Optimierung des Quellcodes für Arduino Nano führten.

Während der Tests wurde auch die zeitabhängige Änderung der Luftfeuchtigkeit im Terrarium gemessen. Die Messungen bestätigten die Funktionsfähigkeit des Regelkreises und stellten dessen Verhalten grafisch dar.

Obwohl es sich nicht um eine fortschrittliche Form der Regelung handelt, wie in [6.2](#) erläutert, erfüllt das Gerät seinen vorgesehenen Zweck und wird für den Betrieb empfohlen.

Literatur

- [1] *Temperatur- und Feuchtigkeitsmodul: DHT11 Produktmanual*. b.m.: Aosong (Guan-gzhou) Electronics Co., 2017.
- [2] *DHT11 Digitales Thermometer und Hygrometer: Datenblatt*. Havlíčkův Brod: ECLIPSERA s.r.o, 2016.
- [3] *LM350: Datenblatt*. 3. Auflage. b.m.: Semiconductor Components Industrie, 2004.
- [4] *Hagen Exo Terra Monsoon Multi Luftbefeuchter* [online]. Prag: Heureka Shopping s.r.o., 2018 [Stand: 17.03.2018]. Verfügbar unter: <https://ostatni-doplneky-teraria.heureka.cz/hagen-exo-terra-monsoon-multi-zvlhcovac/>
- [5] *Luftfeuchtigkeit*. Wikipedia: die freie Enzyklopädie [online]. San Francisco (CA): Wikimedia Foundation, 2018. Verfügbar auch unter: https://cs.wikipedia.org/wiki/Vlhkost_vzduchu
- [6] *Automatisierung und Automatisierungstechnik*. Prag: Computer Press, 2000. ISBN 80-722-6247-5
- [7] *L7800 SERIES: POSITIVE VOLTAGE REGULATORS*. b.m.: STMicroelectronics, 2004.
- [8] BROŽ, Karel. *Schutz vor gefährlicher Berührungsspannung*. Prag: SNTL, 1976.
- [9] *Mikroprozessor*. Wikipedia: die freie Enzyklopädie [online]. San Francisco (CA): Wi-kimedia Foundation, 2001-, 23. 3. 2018 [Zugriff am 26.03.2018]. Verfügbar unter: <https://cs.wikipedia.org/wiki/Mikroprozessor>
- [10] *Arduino Nano (V3.0): Benutzerhandbuch*. 3. Auflage. Kalifornien: Creative Commons Attribution, 2010. Auch verfügbar unter: www.arduino.cc
- [11] VODA, Zbyšek und das Team von HW Kitchen. *Průvodce světem Arduina* (Leitfaden durch die Welt von Arduino). 2. Auflage. Bučo-vice: Verlag Martin Stříž, 2017. ISBN 978-80-87106-93-8. Auch verfügbar unter: <https://www.hwkitchen.cz>

Anhänge

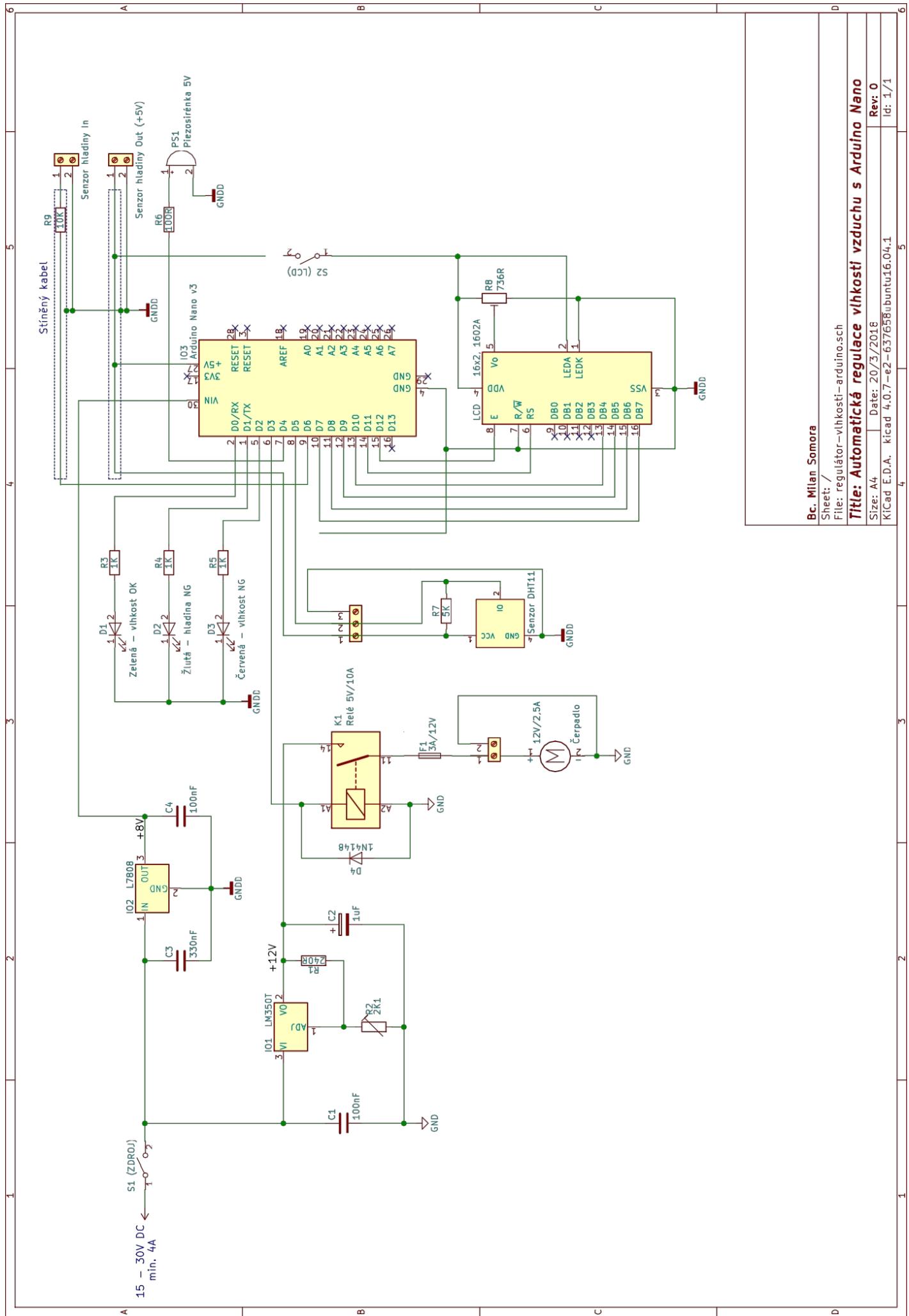
A. Liste der verwendeten Bauteile

| Bezeichnung | Bezeichnung | Wert/Beschreibung | Anzahl |
|--|-------------|-------------------------------------|---------|
| Spannungsquelle | - | 15 VDC/min. 3 A | 1 |
| Spannungsstabilisator LM350T | IO1 | 1,2..33 V/3 A, TO220 | 1 |
| Spannungsstabilisator L7808 | IO2 | 8 V/0,5 A, TO220 | 1 |
| Arduino Nano v3 | IO3 | - | 1 |
| Keramikkondensator | C1 | 100 nF/50 V | 1 |
| Elektrolytkondensator | C2 | 1 µF/50 V | 1 |
| Keramikkondensator | C3 | 330 nF/50 V | 1 |
| Keramikkondensator | C4 | 100 nF/50 V | 1 |
| Metallisierter Widerstand | R1 | 240 R/0,6 W | 1 |
| Kohle-Trimmopotentiometer, liegend | R2 | 5K/0,15 W | 1 |
| Metallisierter Widerstand | R3,R4,R5 | 1K/0,6 W | 3 |
| Metallisierter Widerstand | R6 | 300R/0,6 W | 1 |
| Metallisierter Widerstand | R7 | 5K/0,6 W | 1 Stück |
| Mehrfachdreh-Trimmopotentiometer, stehend | R8 | 2K | 11 |
| Metallisierter Widerstand | R9 | 10K | 1 |
| LED-Diode | D1 | 5 mm, grün | 1 |
| LED-Diode | D2 | 5 mm, gelb | 1 |
| LED-Diode | D3 | 5 mm, rot | 1 |
| Gleichrichterdiode | D4 | 1N4148 | 1 |
| Feuchtigkeits- und Temperatursensor | DHT11 | - | 1 |
| Elektromagnetisches Relais mit Gleichstromspule | K1 | 5 V/10 A | 1 |
| Rohrsicherung | F1 | 5x20 mm, 3 A/12-24 V | 1 |
| Piezosirene | Sirene | 5 V | 1 |
| Schalter für Spannungsquelle (oder Kippschalter) | S1, S2 | Schiebe, Hebel oder Wippe | 2 Stück |
| Pumpe | Pumpe | Spritzwasserpumpe für Auto für 12 V | 1 Stück |

| Bezeichnung | Bezeichnung | Wert/Beschreibung | Menge |
|---------------------------|--------------------|---|--------------|
| Aquarienschlauch | - | d= 6 mm | 1 m |
| Düse | - | Wasserzerstäubung | 1 |
| Kunststoffkanister 5 l | - | - | 1 |
| LCD-Display | LCD | 16X2, Controller HD44780 | 1 |
| Abgeschirmter Kabel | - | 2x0, 15 mm ² | 0,5 m |
| Dreileiterkabel für DHT11 | - | 3x0, 15 mm ² | 1 m |
| Zweipolare Klemmenleiste | Tiefensensor | - | 2 |
| | Tiefsensor Out | - | |
| | - | für Pumpe und Quelle | 2 |
| Kühler für LM350T | - | Aluminium, für Gehäuse TO220, 13 x 19 x 13 mm | 1 Stück |
| Kunststoffbox | - | Schutzart IP54, Abmessungen 100 x 80 x 40 mm (BxTxH) | 1 |
| Universal-Leiterplatte | - | 70 x 50 mm | 1 |
| Sicherungshalter | - | Sicherungshalter für Leiterplatte für Sicherung 5x20 mm | 1 |
| Koaxialer Stromanschluss | - | Stecker, d=5,5 mm/2,5 mm | 1 |
| Koaxialer Stromanschluss | - | Weiblich, Gabel d=5,5 mm/2,5 mm | 1 |
| Ferritmagnete, rund | - | d=20 mm | 4 |
| Gartenschläuche | - | 3/4" | 10 cm |
| Schlauchanschluss | - | Für 3/4"-Schlauch mit 1/2"-Gewinde Gewinde und Dichtung | 1 Stück |
| Reduzierstück 1/2" | - | Reduzierstück mit zwei Außengewinden 1/2" | 1 Stück |
| Dichtung für 1/2"-Gewinde | - | - | 2 |
| Mutter für 1/2"-Gewinde | - | - | 1 |
| Schneckenklemme | - | für 3/4"-Schlauch | 2 |
| Halterung für Düse | - | - | 1 |
| Wärmeleitpaste | | für den Kühler des LM350 | 1 |

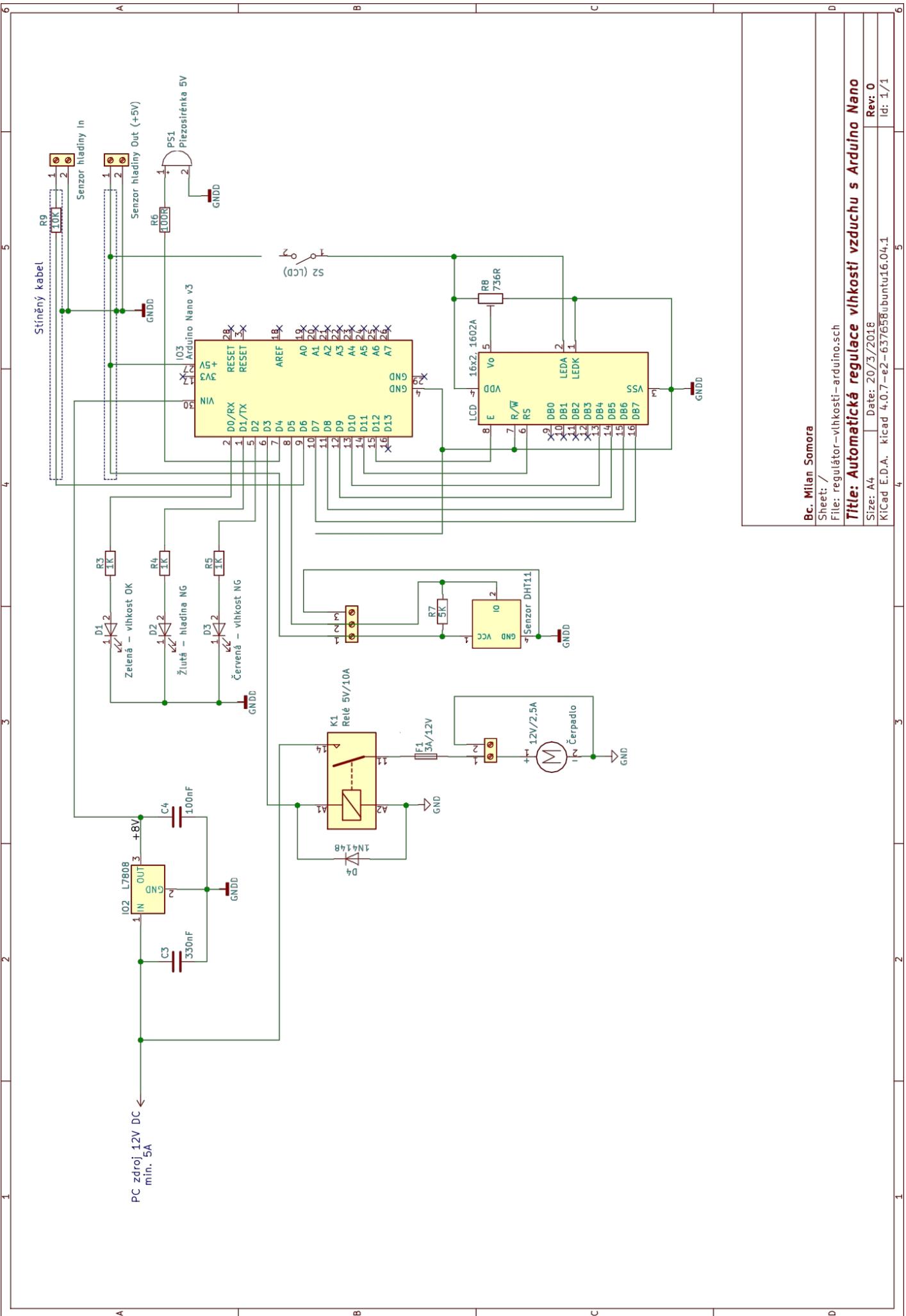
B. Elektrischer Schaltplan des Reglers

Die folgende Seite enthält den elektronischen Schaltplan des Reglers für eine Quelle mit einer Leerlaufspannung von 15 - 30 V.



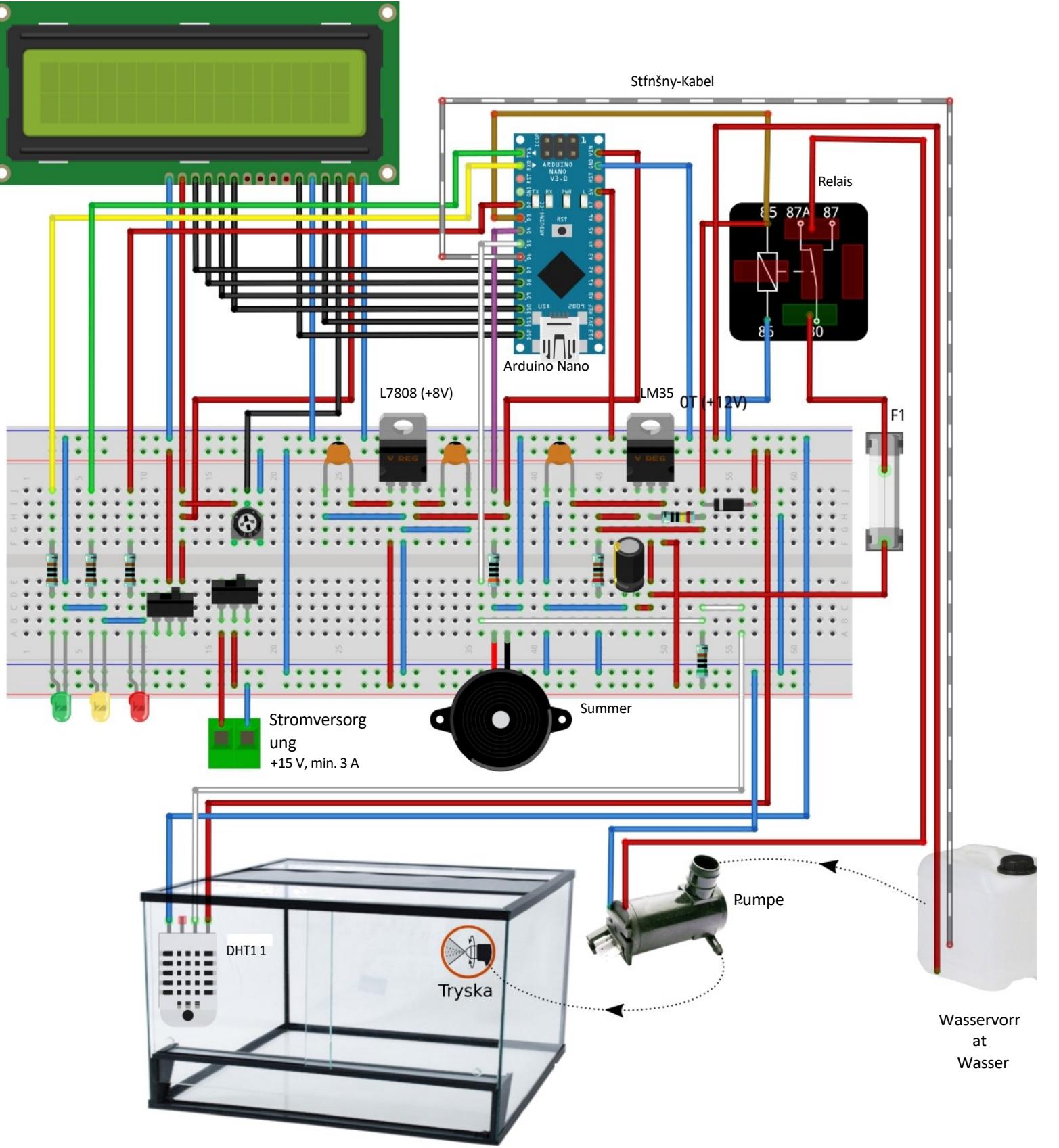
C. Elektroschaltplan mit PC-Netzteil

Die folgende Seite enthält den elektronischen Schaltplan des Reglers für die Verwendung mit einer PC-Stromversorgung.



D. Anschluss an einer lötfreien Platine

Die folgende Seite enthält ein Schaltbild mit einer lötfreien Steckbrücke.



E. Quellcode für Arduino Nano

Nachfolgend finden Sie einen Auszug aus dem Quellcode, der auf das Arduino Nano geladen werden kann. Die Werte in den Zeilen 38–41, 178 und 210 können je nach Verhalten des geregelten Systems beliebig geändert werden. Die angegebenen Bibliotheken in Zeilen 2 - 5 kann heruntergeladen auf Github herunter (einschließlich dieses Quellcodes):

<https://github.com/milansomora/regulator-vlhkosti>.

```
1 // Bibliotheken
2 #include "Adafruit_Sensor.h"
3 #include "Arduino.h"
4 #include "DHT.h"
5 #include "LiquidCrystalFast.h"
6
7 typedef unsigned short USHORT;
8 typedef unsigned long int ULONG;
9
10 // Funktionsprototypen
11 void sensorwertAusgeben();
12 void wasserstandPruefen(bool &niveau);
13 void feuchtigkeitPruefen(USHORT &vlh);
14 void pumpeEinschalten(bool &niveau);
15 void piepen();
16
17 // Definition digitaler Eingänge
18 const USHORT grueneLedPin = 0;
19 const USHORT gelbeLedPin = 1;
20 const USHORT roteLedPin = 2;
21 const USHORT pumpePin = 3;
22 const USHORT summerPin = 4;
23 const USHORT feuchtigkeitsSensorPin = 5;
24 const USHORT wasserstandsSensorPin = 6;
25
26 // Definition der Pins für das LCD-Display
27 const USHORT rs_pin = 11;
28 const USHORT e_pin = 12;
```

```

29 const USHORT d4_pin = 10;
30 const USHORT d5_pin = 9;
31 const USHORT d6_pin = 8;
32 const USHORT d7_pin = 7;
33
34 // Globale Variablen
35 USHORT temperatur, feuchtigkeit;
36 bool Wasserstand;
37
38 // Werte dieser globalen Variablen können beliebig geändert
39 // werden:
40 const USHORT minFeuchtigkeitsniveau = 60; // [%]
41 const USHORT pumpenlaufzeit = 6000; // 6s
42 const ULONG feuchtigkeitsPruefungszeit = 180000; // 3 min
43 const ULONG wasserauffuellPruefungszeit = 300000; // 5 min
44
45 // Einstellung des DHT-Sensortyps
46 #define sensortyp DHT11
47
48 // Deklaration des Feuchtigkeitssensors
49 DHT sensor(feuchtigkeitsSensorPin, sensortyp);
50
51 // Deklaration des LCD-Displays
52 LiquidCrystalFast LCD(rs_pin, e_pin, d4_pin, d5_pin, d6_pin,
53 d7_pin);
54
55 void setup() {
56     // Initialisierung der Eingänge
57     pinMode(pumpePin, OUTPUT);
58     pinMode(summerPin, OUTPUT);
59     pinMode(wasserstandsSensorPin, INPUT);
60
61     // Kommunikation über die serielle Schnittstelle mit 9600
62     // Baud
63     Serial.begin(9600);
64
65     // Einschalten der Kommunikation mit dem Feuchtigkeitssensor
66     sensor.begin();
67
68     // Einschalten des LCD-Displays

```

```

67     LCD.begin(16, 2);
68     LCD.setCursor(1, 0);
69     LCD.print("Aut. Regelung");
70     LCD.setCursor(4, 1);
71     LCD.print("der Feuchtigkeit");
72     delay(1000);
73 }
74
75 void loop() {
76     // Einlesen der Werte vom Feuchtigkeitssensor
77     temperatur = sensor.readTemperature();
78     feuchtigkeit = sensor.readHumidity();
79
80     // Prüfung des DHT-Sensors (Kommunikation mit Arduino)
81     if (feuchtigkeit == 0 || temperatur == 0) {
82         // Ausgabe auf das LCD
83         LCD.begin(16, 2);
84         LCD.setCursor(0, 0);
85         LCD.print("DHT Sensorfehler");
86         LCD.setCursor(0, 1);
87         LCD.print("!");
88         delay(2000);
89         LCD.begin(16, 2);
90         LCD.setCursor(0, 0);
91         LCD.print("Prüfen Sie");
92         LCD.setCursor(0, 1);
93         LCD.print("den Sensor.");
94
95         // Ausgabe auf die Konsole
96         Serial.println("Fehler beim Lesen vom DHT-Sensor!");
97         delay(2000);
98     }
99     else {
100         sensorwertAusgeben();
101         Wasserstand = digitalRead(wasserstandsSensorPin);
102
103         // Überprüfung des Wasserstands im Tank
104         wasserstandPruefen(Wasserstand);
105
106         // Prüfung und Regelung der Luftfeuchtigkeit im Terrarium
107         feuchtigkeitPruefen(feuchtigkeit);

```

```

108
109     if (feuchtigkeit >= minFeuchtigkeitsniveau && Wasserstand
110         == true) {
111         // Ausgabe auf das LCD
112         LCD.setCursor(0, 1);
113         LCD.print("                ");
114         LCD.setCursor(0, 1);
115         LCD.print("Alles ist OK!");
116
117         // Ausgabe na die Konsole
118         Serial.println("Alles ist OK!");
119     }
120
121     // Für bessere Lesbarkeit der Konsolenausgabe
122     Serial.println(" ");
123     delay(2000);
124 }
125
126 void sensorwertAusgeben() {
127     // Ausgabe auf das LCD
128     LCD.begin(16, 2);
129     LCD.setCursor(0, 0);
130     LCD.print("F: ");
131
132     LCD.setCursor(3, 0);
133     LCD.print(feuchtigkeit);
134
135     LCD.setCursor(5, 0);
136     LCD.print("%, ");
137
138     LCD.setCursor(8, 0);
139     LCD.print("T: ");
140     LCD.setCursor(11, 0);
141     LCD.print(temperatur);
142     LCD.setCursor(13, 0);
143     LCD.print((char)223);
144     LCD.setCursor(14, 0);
145     LCD.print("C");
146
147     // Ausgabe auf die Konsole

```

```

148     Serial.print("Feuchtigkeit: ");
149     Serial.print(feuchtigkeit);
150     Serial.print(" %, ");
151     Serial.print("Temperatur: ");
152     Serial.print(temperatur);
153     Serial.println(" C");
154 }
155 void wasserstandPruefen(bool &niveau) {
156     if (niveau == true) {
157         digitalWrite(gelbeLedPin, LOW);
158     }
159     else {
160         digitalWrite(gelbeLedPin, HIGH);
161
162         // Bis der Benutzer Wasser nachfüllt:
163         while (!niveau) {
164             piepen(); // 3x Piepsen aus dem Piezo-Summer
165
166             // Ausgabe auf das LCD
167             LCD.begin(16, 2);
168             LCD.setCursor(0, 0);
169             LCD.print("Kein Wasser");
170             LCD.setCursor(0, 1);
171             LCD.print("Warte 5 min...");
```

172

```

173             // Ausgabe na die Konsole
174             Serial.println("- Warte auf Wasserauffüllung (5 min)
175                         ...");
176             delay(wasserauffuellPruefungszeit);
177
178             // Wasserstandsprüfung zum Beenden der while-Schleife
179             Wasserstand = digitalRead(wasserstandsSensorPin);
180             if (Wasserstand) {
181                 LCD.begin(16, 2);
182                 break;
183             }
184         }
185     }
186
187 }
```

```

188 void pumpeEinschalten(bool &niveau) {
189     if (niveau == true) {
190         // Ausgabe auf das LCD
191         LCD.setCursor(0, 1);
192         LCD.print("                ");
193         LCD.setCursor(0, 1);
194         LCD.print("Bewässere...");
```

195

```

196         // Einschalten der Pumpe
197         digitalWrite(pumpePin, HIGH); // ein
198         delay(pumpenlaufzeit);
199         digitalWrite(pumpePin, LOW); // aus
```

200

```

201         // Ausgabe auf das LCD
202         LCD.begin(16, 2);
203         LCD.setCursor(0, 0);
204         LCD.print("Prüfe");
205         LCD.setCursor(0, 1);
206         LCD.print("Feuchtigkeit (3 min)");
```

207

```

208         // Ausgabe na die Konsole
209         Serial.println("+ Prüfe Feuchtigkeit...");
```

210 delay(feuchtigkeitsPruefungszeit);

211 }

212 else {

213 digitalWrite(pumpePin, LOW);

214 }

215 }

216

```

217 void feuchtigkeitPruefen(USHORT &vlh) {
218     if (vlh > minFeuchtigkeitsniveau) {
219         // Feuchtigkeit ist OK
220         digitalWrite(grueneLedPin, HIGH);
221         digitalWrite(roteLedPin, LOW);
222     }
223     else {
224         // Feuchtigkeit ist niedrig
225         digitalWrite(grueneLedPin, LOW);
226         digitalWrite(roteLedPin, HIGH);
227         pumpeEinschalten(Wasserstand);
228     }
```

```
229 }
230
231
232 void piepen() {
233     for (USHORT i = 0; i < 3; i++) {
234         digitalWrite(summerPin, HIGH);
235         delay(450);
236         digitalWrite(summerPin, LOW);
237         delay(200);
238     }
239 }
```