



Warm und hell

Blumenheizungssteuerung selbst gebaut

Zieht man Pflanzen im späten Winter selbst aus Samen an, muss man schon etwas Obacht geben, damit der Erfolg auch garantiert ist. Oft scheitert die Anzucht aber an schwankenden bzw. ungenügenden Umgebungsbedingungen, wie etwa stark schwankenden Temperaturen bei der Anzucht auf dem Fensterbrett. Unser Leser Hans-Joachim Beck hat sich dieses Themas angenommen und auf Arduino-Basis eine Blumenheizung samt Beleuchtungssteuerung entwickelt, die den kleinen Pflanzen definierte Wachstumsbedingungen schafft.

Gut beleuchtet und mit warmen „Füßen“

Jedes Jahr ziehen wir in unserem Garten Peperoni, und da wir verschiedene und nicht im Gartenmarkt zu kaufende Varianten haben, müssen

wir diese selber aus dem Samen ziehen. Um einen vernünftigen Ertrag zu haben, muss man damit schon im Februar beginnen. Leider ist das aber die Jahreszeit, zu der häufig die Umweltbedingungen dafür nicht optimal sind, es ist einfach zu kalt und zu dunkel.

Immer wieder haben wir improvisiert und so ganz befriedigend war das Ergebnis nie, dem werden wohl die meisten zustimmen, die sich ebenfalls mit diesem Thema beschäftigen.

Aus diesen Gründen hatte ich mich entschieden, hier etwas selber zu bauen, das die beiden oben genannten Probleme beheben sollte. Es musste also eine Heizung her, die die Anzuchtschale auf die entsprechende Keimtemperatur bringen soll, eine Zeitschaltung für eine Beleuchtung sollte natürlich auch gleich integriert werden.

In **Bild 1** ist das Ergebnis gleich vorweg gezeigt – die Steuerung wurde zusammen mit einer Steckdose für die Heizung und zwei Steckdosen für die Beleuchtung auf einem Holzbrett montiert.

Die Komponenten

In **Bild 2** sind die Komponenten der Steuerung zusammengefasst, diesen widmen wir uns später noch detailliert:



Hans-Joachim Beck

hat für seinen Beitrag zum Leserwettbewerb
einen 200-Euro-Warengutschein erhalten!



Bild 1: Die komplette Steuerung, gemeinsam mit den zugehörigen Netzsteckdosen auf einem Brett montiert

Die Zentrale

Das ist der alle Prozesse steuernde Mikroprozessor, hier ein ATmega 328P in einem Arduino-Pro-Mini-System.

Die Stromversorgung

Diese stellt, gespeist von einem Steckernetzteil, die Betriebsspannung für alle Schaltungskomponenten zur Verfügung. Der Netzanschluss versorgt einmal über ein Halbleiterrelais die Kabelheizung und einmal über ein Relais die Beleuchtung mit 230 V.



Wichtiger Hinweis:

Vorsicht! Aufgrund der im Gerät frei geführten Netzspannung dürfen Aufbau und Inbetriebnahme ausschließlich von unterwiesenen Elektrofachkräften durchgeführt werden, die aufgrund ihrer Ausbildung dazu befugt sind. Die einschlägigen Sicherheits- und VDE-Bestimmungen sind unbedingt zu beachten.

Display und Eingabe

Dieser Funktionsblock dient der Eingabe von Vorgaben für die Soll-Temperatur und der Einschaltzeiten für die Beleuchtung und der Anzeige aller relevanten Parameter und Messwerte.

Temperaturregelung

Die Temperaturregelung besteht, neben der Softwaresteuerung, aus einer Regelung und dem Leistungsteil.

Lichtsteuerung

Die Lichtsteuerung wird aus dem Softwareteil und dem Leistungsteil gebildet, ein RTC-Baustein sorgt für exakte Zeitabläufe.

Temperaturmessung

Sie basiert auf einem in der Anzuchtschale platzierten Temperatursensor, der die Messdaten für die Temperaturregelung liefert.

Anforderungen und Vorüberlegungen

Die Blumenheizung sollte in der Lage sein, die Temperatur in einer Anzuchtschale konstant zu halten und auch eine zusätzliche Beleuchtung zu steuern. Um beides möglichst variabel halten zu können, müssen die entsprechenden Parameter natürlich eingegeben werden können, also mussten ein Display und eine Eingabemöglichkeit – hier habe ich mich für einen Encoder entschieden – integriert werden.

Die beiden Leistungsteile Heizung und Licht sollten unterschiedlich ausgeführt werden (die Heizung bildet eine rein ohmsche Last, bei der Beleuchtung können die Lasten hingegen ganz unterschiedlich sein – von der einfachen Glühlampe über Leuchtstofflampen bis hin zur LED-Beleuchtung). Bei der Temperatur muss eine Regelung möglich sein, das Licht hingegen muss nur ein- und ausgeschaltet werden.

Die Komponenten im Detail

In Bild 3 ist die Gesamtschaltung der Blumenheizung zu sehen, deren Funktionsteile wir nun genauer betrachten wollen.

Die Zentrale

Klar war, das Ganze muss über einen Mikroprozessor gesteuert werden, hier habe ich zu einem Arduino Pro Mini gegriffen. Dieser vereinbart in meinen Augen die Einfachheit der Programmierung und Handhabung des Arduino mit der einfachen Integrierbarkeit in eine eigene Baugruppe.

Da ja das Licht zeitgesteuert ein- und ausgeschaltet werden soll, ist auch ein Uhrenbaustein, der DS1307, integriert worden. Dieser wird durch einen Supercap im Falle eines Spannungsausfalls mit Strom versorgt, somit muss man in diesem Falle nicht nach Spannungswiederkehr die Zeit neu einstellen.

Die Stromversorgung

Diese wird, wie bereits beschrieben, durch ein Steckernetzteil realisiert, das 8 V Gleichspannung liefert. Ein 5-V-Linearregler (IC2) erzeugt daraus die benötigten 5 V für alle Schaltungskomponenten.

Display und Eingabe

Hier habe ich ein zweizeiliges LC-Display eingesetzt. Um die Anzahl der benötigten Mikroprozessor-Ports möglichst gering zu halten, habe ich mich für einen Anschluss über den I²C-Bus entschieden, über diesen wird ja auch schon die Echtzeituhr angesteuert. Dazu ist zwischen Mikroprozessor und dem HD44780-Display ein I²C-Adapter geschaltet.

In anderen selbst gebauten Geräten hatte ich schon öfter zu einem Encoder als Bedienelement gegriffen, so auch hier. Ein Encoder ist in meinen Augen das optimale Eingabegerät, um über eine entsprechende Menüsteuerung diverse Parameter eingeben zu können.

Die Temperaturregelung

Die Temperaturregelung erfolgt über einen PID-Regler, der Bestandteil des Arduino-Programms [1] ist. Beim Leistungsteil denkt man dann natürlich gleich an eine Phasenanschnittsteuerung oder Ähnliches. Hier habe ich mich aber gegen solch eine Lösung entschieden, denn eine Temperaturregelung so einer Blumenheizung ist ein sehr träges Element, hier muss ich nicht so schnell reagieren. Zudem fand ich das hier als zu viel Aufwand für diesen Anwendungszweck und habe folgende Lösung realisiert.

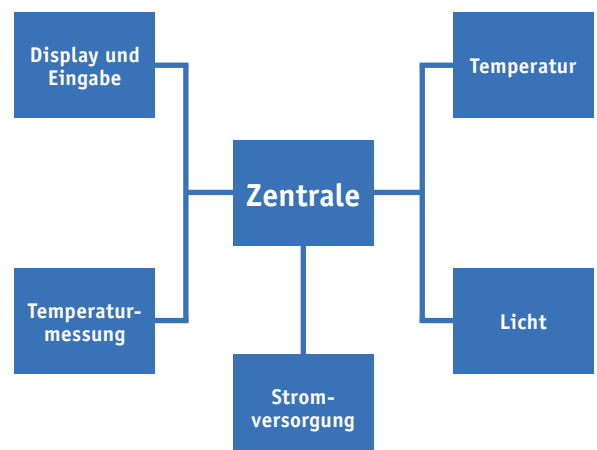


Bild 2: Überblick über die Komponenten der Steuerung

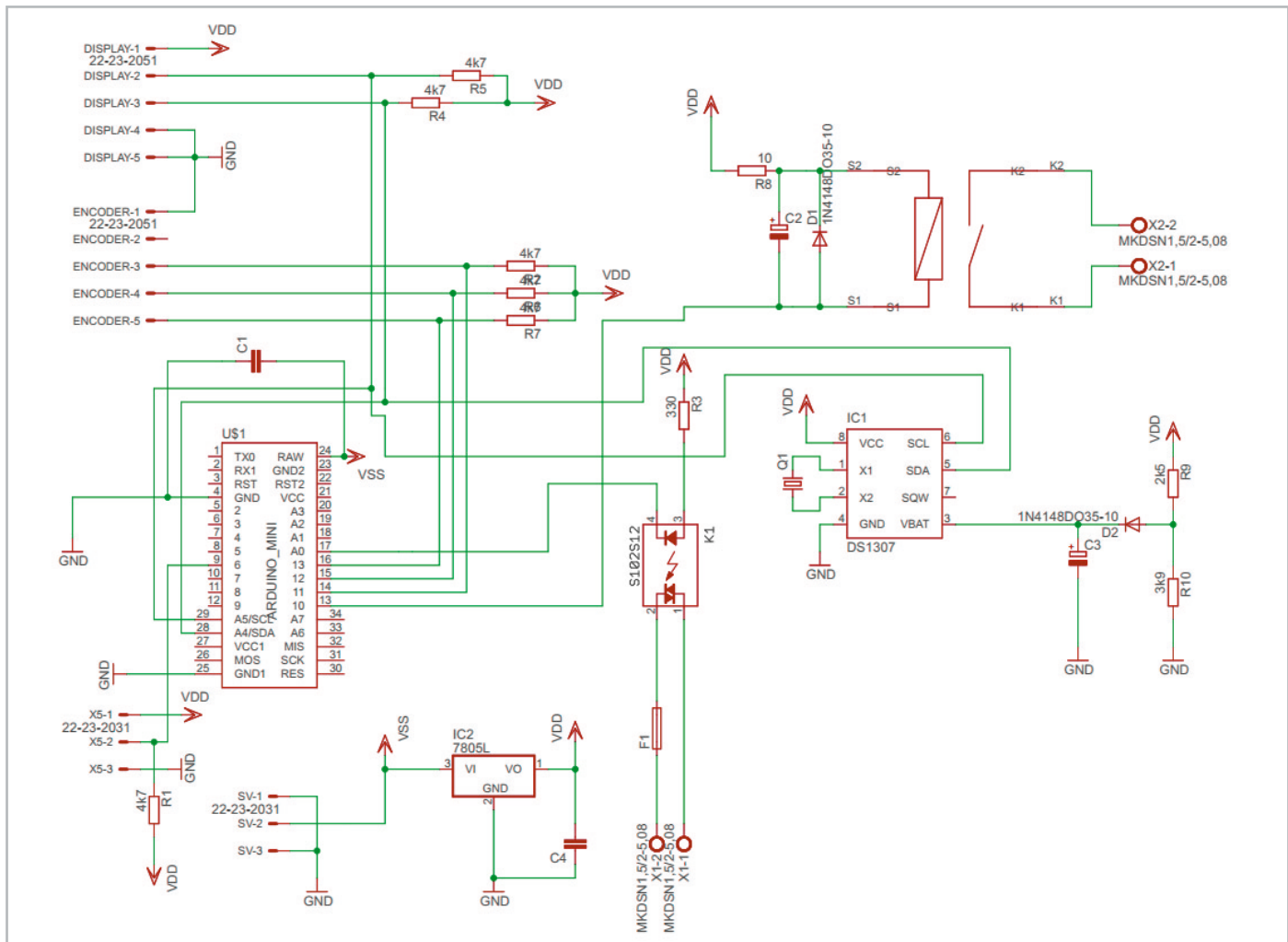


Bild 3: Das Schaltbild der Elektronik-Komponenten der Blumenheizung

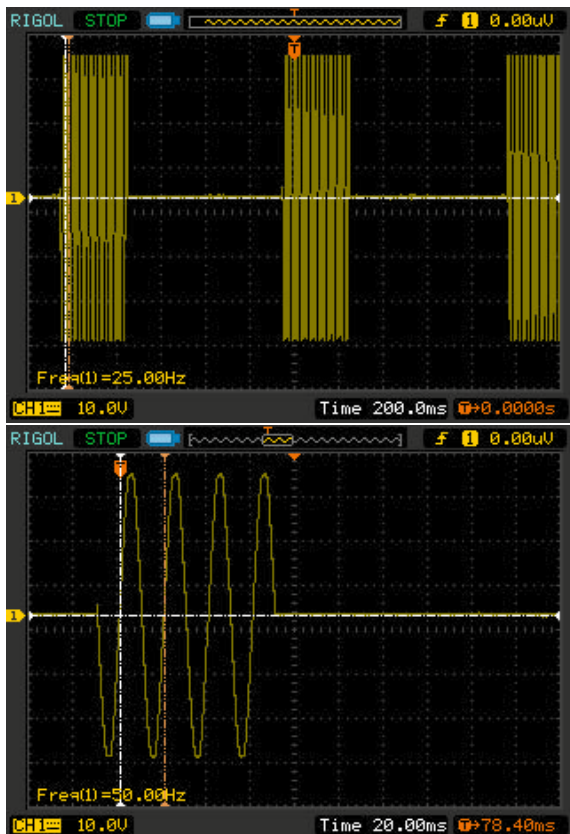


Bild 4: Die Auswirkung der Regelung ist hier gut zu sehen, einmal mit 30 % Heizleistung, einmal mit 8 %

Unsere 50-Hz-Netzspannung erzeugt in 1 Sekunde 100 Halbwellen. Bei der Zahl 100 kommt man natürlich auch sofort auf 100 %, und damit war dann meine Entscheidung klar – ich schalte pro Sekunde nur die entsprechende Anzahl von Halbwellen ein und habe damit eine sehr feine Regelungsmöglichkeit.

Modere Halbleiterrelais beinhalten bereits eine Nulldurchgangserkennung und damit werden hier auch alle Störungen eliminiert, die bei einer Phasenanschnittsteuerung entstehen können. Zusätzlich macht die komplette galvanische Entkopplung der Netzspannung zu meiner restlichen Schaltung solch ein Halbleiterrelais zu dem idealen Bauelement in meiner Temperaturregelung.

Der PID-Regelalgorithmus erfasst die Ist-Temperatur und errechnet mit den anderen Regelparametern einen prozentualen Steuerwert. In meinem Mikroprozessor läuft ein zyklischer Interrupt alle 10 ms, und wenn der PID-Regler z. B. 30 % ermittelt hat, wird das Halbleiterrelais 30 Interruptzyklen lang eingeschaltet, also 30 Halbwellen, gleich 300 ms.

In Bild 4 ist die Auswirkung der Regelung gut zu sehen, einmal mit 30 % Heizleistung, einmal mit 8 %. Im Bild mit 8 % Heizleistung ist gut zu sehen, dass die Heizung immer im Nulldurchgang ein- und ausschaltet.

Die Lichtsteuerung

Im Gegensatz zur Temperatursteuerung wird hier einfach ein Schaltrelais eingesetzt, das unkompliziert verschiedene Lastarten schalten kann.

Die Temperatursteuerung

Da die Temperatur direkt in der Anzuchterde gemessen werden sollte, habe ich hier einen DS18B20-Tempersensor eingesetzt, der bereits in



einem gekapselten Edelstahlgehäuse untergebracht ist und so direkt in die Erde gesteckt werden kann.

Parametereingabe per Encoder

Um alle Parameter der Blumenheizung eingeben zu können, habe ich in die Arduino-Software eine entsprechende Menüsteuerung implementiert. Bild 5 zeigt den Aufbau des Menüs.

Mit dem Taster des Encoders kann man vom Normalbetrieb in das Set-up wechseln. Mit der Drehung des Encoders wird zwischen den einzelnen Parametern gewechselt. Durch Tastendruck erfolgt die Anwahl der nächsten Ebene oder auch die Eingabe der einzelnen Werte, wenn man sich auf der untersten Ebene befindet.

Ein Tastendruck beim Menüpunkt Regelparameter wechselt auf die nächste Ebene und man kann die Soll-Temperatur oder die Regelparameter anwählen.

Ein Tastendruck im Menüpunkt Soll-Temperatur wechselt in die Eingabe der Soll-Temperatur. Jetzt kann man die Soll-Temperatur durch das Drehen des Encoders ändern. Mit dem nächsten Tastendruck wird wieder eine Ebene höher angewählt.

Wie man erkennen kann, sind drei Ein- und Ausschalzeiten des Lichts einstellbar. Beim PID-Regler können neben der Soll-Temperatur (in 1-Grad-Schritten) auch die drei PID-Regelparameter eingegeben werden.

Aufbau, Betrieb und Anwendungshinweise

Die gesamte Steuerung wurde in ein Kunststoff-Modulgehäuse eingebaut, in dessen Deckel lediglich Ausschnitte für das Display und die Achse des Encoders eingebracht wurden.

Die Schaltung ist auf einer zweilagigen Leiterplatte aufgebaut, die auch den Arduino Pro Mini aufnimmt.

Bild 6 zeigt das Gerät in Aktion. Im Display erkennt man links oben die aktuelle Zeit, rechts die eingestellte Soll-Temperatur von hier 27 °C, links unten die aktuelle Ist-Temperatur (26,4 °C), und rechts unten wird die aktuell vom Regler eingestellte Heizleistung von hier 7 % angezeigt.

Als Heizung habe ich ein Heizkabel aus dem Terrarienbereich benutzt. Dieses lässt sich leicht verlegen und ich muss mir keine Gedanken wegen der Isolation machen. Bei uns reicht die eingesetzte 15-W-Heizung vollkommen aus.

Angefangen haben wir mit einer Anzuchtschale, in die ich das Heizkabel eingelegt hatte. Mittlerweile ist diese Anzuchtschale einem alten, kleinen Aquarium gewichen. Das Heizkabel habe ich mit Silikon auf einem Brett mäanderförmig aufgeklebt und darauf steht nun das Aquarium (ca. 50 cm breit). Der Vorteil ist nun, dass ich die Luftfeuchtigkeit für die Keimlinge besser regulieren kann und das Heizkabel liegt trocken.

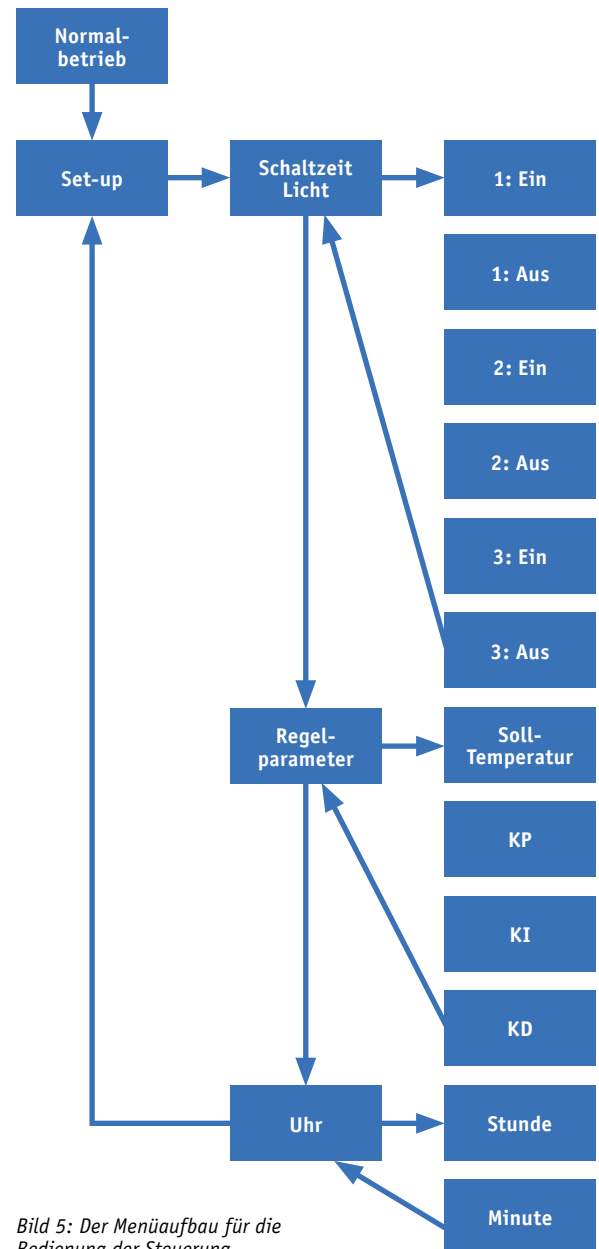
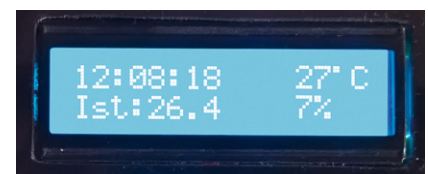


Bild 5: Der Menüaufbau für die Bedienung der Steuerung

Bild 6: Die Blumenheizung im Betrieb, das Display stellt alle relevanten Parameter übersichtlich dar



[1] Arduino-Programm der Blumenheizung:
www.elv.de: Webcode #10182

Machen Sie mit!

Jede veröffentlichte Anwendung wird mit einem Warengutschein in Höhe von 200 Euro belohnt.



Wir wollen es wissen – Ihre Anwendungen und Applikationen!

Welche eigenen kreativen Anwendungen und Applikationen haben Sie mit den ELV Haustechniksystemen, aber auch mit anderen Produkten und Bausätzen realisiert – mit Standardbausteinen oder eingebunden in eigene Applikationen? Alles, was nicht gegen Gesetze oder z. B. VDE-Vorschriften verstößt, ist interessant. Denn viele Applikationen verhelfen sicher anderen zum Aha-Erlebnis und zur eigenen Lösung. Schreiben Sie uns, fotografieren Sie Ihre Applikation, berichten Sie von Ihren Erfahrungen und Lösungen. Die interessantesten Anwendungen werden redaktionell bearbeitet und im ELV Journal mit Nennung des Namens vorgestellt.



Per E-Mail
leserwettbewerb@elv.de



Per Post
ELV Elektronik AG, Leserwettbewerb, 26787 Leer

Die Auswahl der Veröffentlichungen wird allein durch die ELV Redaktion ausschließlich nach Originalität, praktischem Nutzen und realisierter bzw. dokumentierter Ausführung vorgenommen, es besteht kein Anspruch auf Veröffentlichung, auch bei themengleichen Lösungen. **Der Rechtsweg ist ausgeschlossen.** Für Ansprüche Dritter, Beschädigung und Verlust der Einsendungen wird keine Haftung übernommen. Alle Rechte an Fotos, Unterlagen usw. müssen beim Einsender liegen. Die eingesandten Unterlagen und Aufnahmen verbleiben bei der ELV Elektronik AG und können von dieser für Veröffentlichungen und zu Werbezwecken genutzt werden.