

Horticulture Box: LED-Licht für Pflanzen

Mit „Horticulture-LEDs“ von Würth Elektronik

Von Luc Lemmens und Mathias Claußen (Elektor-Labor) mit Beiträgen von Würth Elektronik

Schon in der zweiten Hälfte des 19. Jahrhunderts erforschte und nutzte man künstliches elektrisches Licht zur Stimulation des Pflanzenwachstums – parallel zur generellen Entwicklung elektrischer Beleuchtung. Heute

werden zunehmend auch LEDs zur Pflanzenzucht eingesetzt. Dieses

Projekt präsentiert eine „pflanzenfreundliche“ LED-Beleuchtung, die kaum Wünsche offen lässt.



Die Theorie zu LEDs für Pflanzenleuchten ist reichlich komplex und geht weit über den Rahmen dieses Artikels hinaus. Interessierte finden aber nützliche Informationen zu diesen Hintergrundthemen in den Application Notes AN0002, AN0003, AN0004 auf der Webseite von Würth Elektronik [1]. Hier reicht es, auf die wichtigsten Vorteile von LEDs bei der Aufzucht von Pflanzen einzugehen. Im Wesentlichen geht es nicht nur um die Lichtausbeute, sondern auch das Spektrum des Lichts, das für Wachstum und Entwicklung von Pflanzen relevant ist. Nachfolgend einige typische spektrale Bänder und ihre bekannten Wirkungen:

- Rotes Licht (630...660 nm) ist der Hauptfaktor für die Photosynthese und wichtig für das Wachstum von Stämmen. Dieser Spektralbereich kann auch die Blüte, die Ruhezeit und die Samenkeimung regulieren.
- Blaues Licht (400...520 nm) ist ebenfalls wichtig für die Photosynthese, muss aber sorgfältig kontrolliert und mit anderen Frequenzen vermischt werden, da eine Überbelichtung mit blauem Licht das Wachstum hemmen kann. Dieser Spektralbereich ist auch an der Regulierung der Chlorophyllkonzentration, am Wachstum seitlicher Knospen und der Blattdicke beteiligt.
- Nahes IR-Licht (720...740 nm) beeinflusst die Keimung und kann die Blütezeit der Pflanzen verkürzen, aber auch den Stamm durch eine „Schattenvermeidungsreaktion“ verlängern.

- Grünes Licht (500-600 nm) wurde früher als eher unwichtig für die Pflanzenentwicklung betrachtet. Neuere Untersuchungen zeigen jedoch, dass abgeschattete Pflanzen bei der „Schattenvermeidungsreaktion“ besonders auf diese Lichtfarbe reagieren.
- UV-Licht (280-400 nm) ist bei der Pflanzenzucht noch im Experimentierstadium. UV-Licht ist zwar mutagen, jedoch sind einige Pflanzen (z.B. Salat oder Tomaten) gegenüber diesem Spektralbereich viel resistenter. UV-Licht scheint für die Bildung bestimmter Schutzmoleküle wie Antioxidantien und Phenole verantwortlich sein, die für die menschliche Ernährung wichtig sind.

Im Gegensatz zu technologisch älteren Lichtquellen können LEDs so hergestellt werden, dass sie Licht mit sehr spezifischen Spektren abstrahlen. Dies ermöglicht die Abstimmung des Spektrums (mit verschiedenen Farben von LEDs) auf die spezifischen Bedürfnisse von Pflanzen. Würth Elektronik bietet hierfür die LED-Familie „WL-SMDC SMD Mono-color Ceramic LED Waterclear“ an. Das Lichtspektrum der LEDs wurde hier auf die Wellenlängen 450 nm (tiefblau), 660 nm (hyperrot) und 730 nm (nahes Infrarot) erweitert, so dass sie mit den Absorptionsspektren von Photosynthese-Pigmenten übereinstimmen. Genau mit diesen drei LED-Varianten plus einigen kaltweißen LEDs mit einer Farbtemperatur von 6000 K wurde die hier beschriebene LED-Pflanzenleuchte realisiert.

INFOS ZUM PROJEKT

	LED PWM ESP32
	Pflanzenleuchte
	Gartenbau
	Einsteiger
	Fortgeschrittene
	Experte
	ca. 2 Stunden
	Arduino IDE (optional)
	ca. 160 €

Der ESP32-Mikrocontroller in der Pflanzenleuchte verfügt über eine spezielle Web-Schnittstelle, mit der man die Intensität jedes der vier Kanäle unabhängig voneinander sowie die Gesamtintensität der vier Kanäle per WLAN mit einem PC, Smartphone oder Tablet steuern kann. Man kann also das Licht der Pflanzenleuchte einfach ändern und die resultierenden Effekte austesten.

Unter den vielen weiteren Vorteilen der Verwendung von LEDs in Pflanzenleuchten sind deren Größe, Wirkungsgrad und Haltbarkeit besonders zu erwähnen.

Eigenschaften

- 4 Farbkanäle: rot, nahes IR, blau, weiß
- Hochleistungs-LEDs WL-SMDC von Würth Elektronik
- LED-Strom auf 300/350/450 mA einstellbar
- 4 PWM-Treiber
- ESP32-gesteuert
- Vollständig einstellbare Farbmischung und Helligkeit
- Timer zur Simulation des Tag/Nacht-Zyklus
- Beleuchtete Fläche mit Radius von 55 cm
- WLAN-gesteuert (Webinterface / MQTT)
- Beliebige Netzteil mit 24 V und 50 W
- Treiberplatine und LED-Platine vorbestückt erhältlich

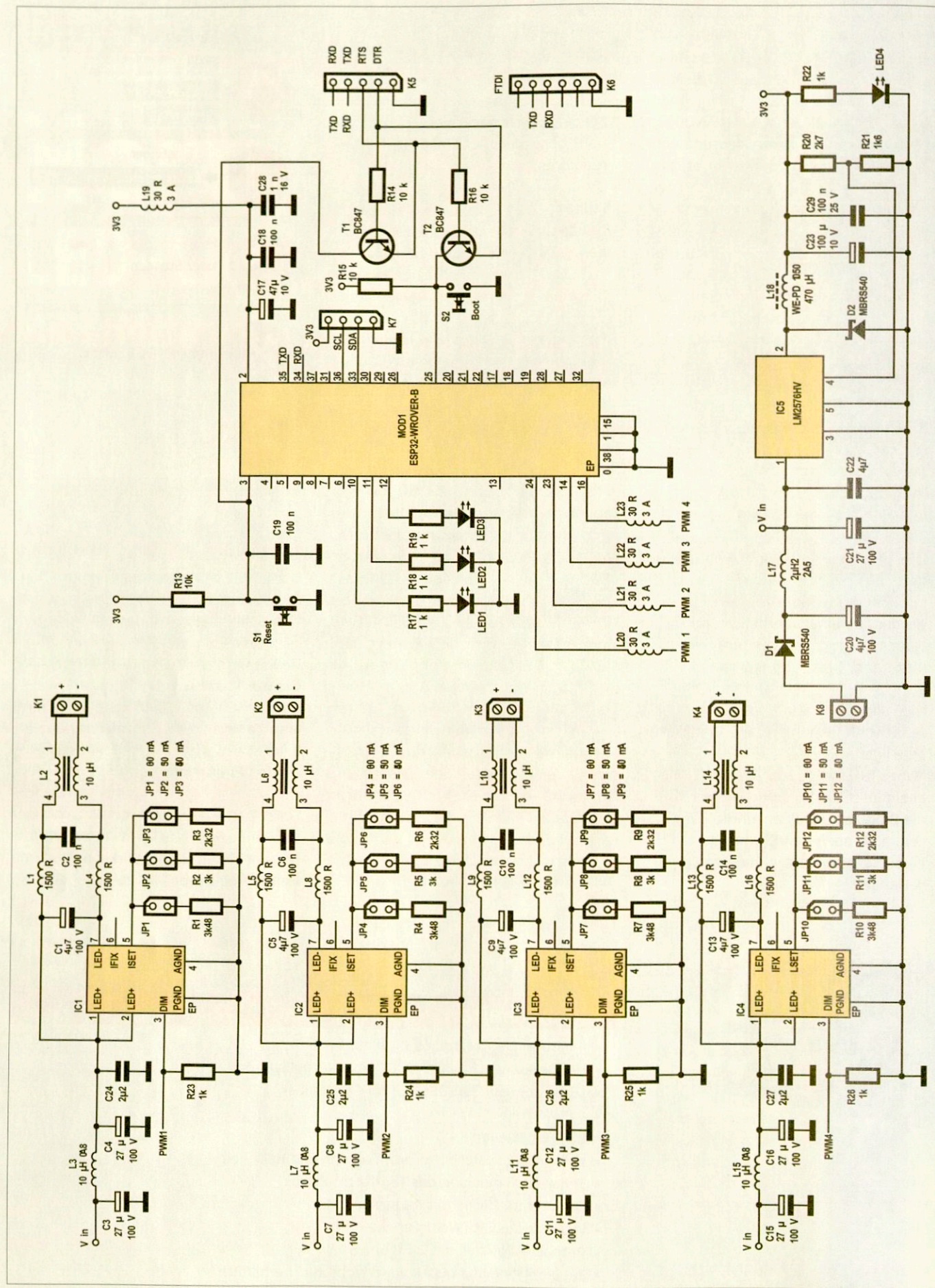


Bild 1. Schaltung der Steuerplatine der Pflanzenleuchte. Hier herrscht das ESP32-Modul.

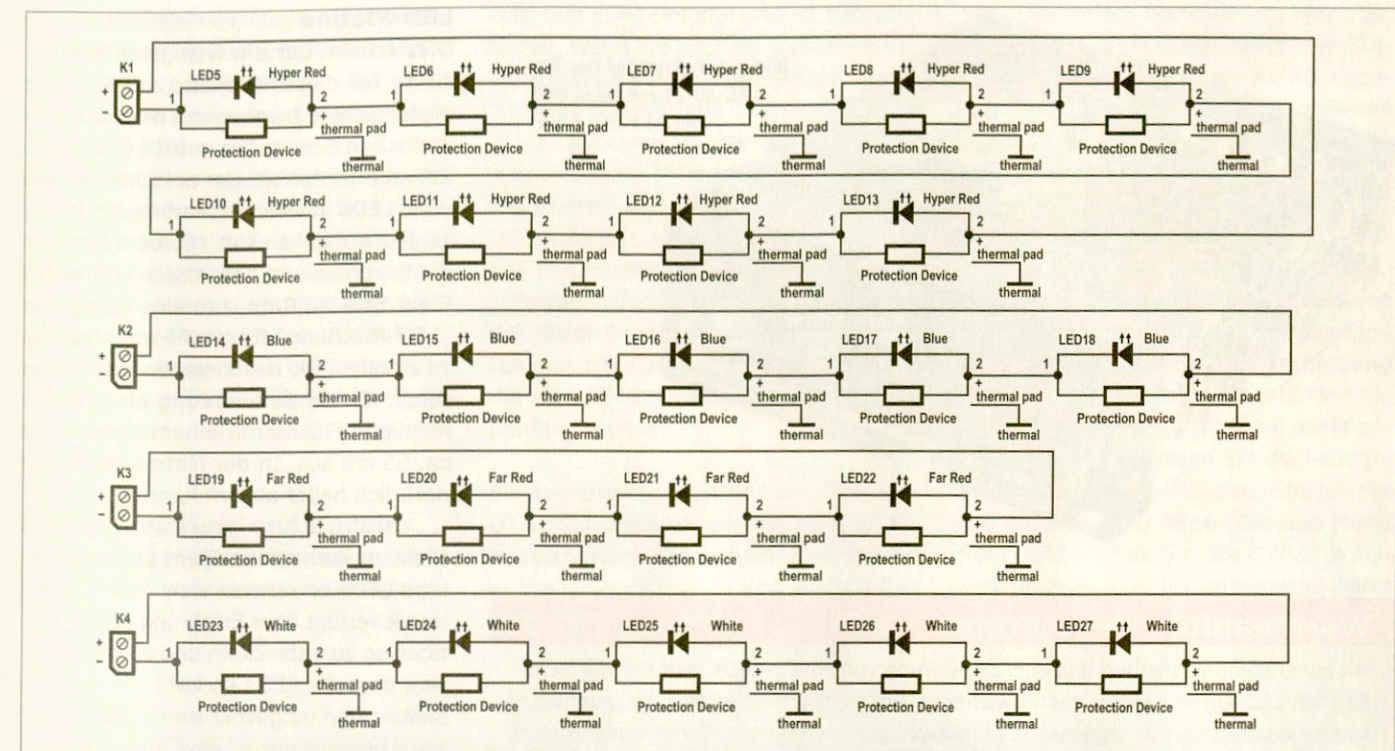


Bild 2. Schaltung der LED-Platine.

Treiber-Platine

Der größte Teil der Schaltung entstammt einem Referenz-Design von Würth Elektronik, das einen vierkanaligen PWM-LED-Treiber bietet. Die Elektor-Entwickler Luc Lemmens und Mathias Claußen ersetzten den PIC-Mikrocontroller samt Bluetooth-Modul des Original-Designs durch das beliebte ESP32-Modul „ESP32-WROVER-B“. Anstelle des Versorgungsmoduls des Referenz-Designs entschied sich Luc für den DC/DC-Wandler LM2576 zur Versorgung des ESP32.

Die Schaltung von **Bild 1** zeigt die Ergebnisse dieser Überarbeitung. Dank der enthaltenen Induktivitäten und Kondensatoren werden die EMV-Grenzwerte der Normen EN55015 und CISPR32 auch beim Dimmen eingehalten. In der modifizierten Ausführung bestehen die vier Treiberstufen deren Pre-Compliance-Tests problemlos. Neben den Schaltungen werden hier auch die zugehörigen Platinen abgebildet. Der ESP32 selbst machte mit seinem 80-MHz-Takt samt Oberwellen zuerst ernsthafte Probleme. Es war sogar eine Neugestaltung der Platine erforderlich. Die Ground-Plane musste teilweise neu geroutet werden, und einige Ferrit-Perlen wurden hinzugefügt.

Ansonsten ist der Schaltplan recht einfach. Das ESP32-WROVER-Modul ist zentral für die Treiberplatine, denn es

generiert die PWM-Signale, mit der die Lichtintensität gesteuert wird. Die MCU kümmert sich sowohl um die Benutzeroberfläche als auch um die Verbindung zum WLAN. Der Abwärtswandler LM2576 (IC5) stellt die 3,3-V-Versorgung für das ESP32-Modul (MOD1) zur Verfügung. Die blaue LED4 signalisiert den Status der Stromversorgung. Die Eingangsspannung könnte bis zu 45 V betragen. Um jedoch die Verlustleistung auf der Platine bei eingeschalteten LEDs zu begrenzen, wurde die Eingangsspannung auf 24 V begrenzt.

Auf den UART des ESP32 kann zwecks serieller Kommunikation über K5 oder K6 zugegriffen werden. Auch bei der Programmierung des ESP im Bootloader-Mode kommt er zum Zuge. Die Verbindungen zu DTR und RTS auf K5 ermöglichen es der Arduino-IDE, den ESP32 automatisch über T1 und T2 in den Bootloader-Mode zu schalten. Hierzu benötigt man aber eine USB/UART-Schnittstelle, die diese beiden Handshake-Signale bereitstellt. Geeignet sind z.B. die Typen *USB/Serial Bridge/BoB* oder der *USB-RS232-Konverter „FT231X BoB“* von Elektor. Sie können auch ein normales 3,3-V-FTDI-Kabel verwenden und es an den K6 anschließen. In diesem Fall müssen Sie aber, wie vielleicht von anderen ESP-basierten Elek-

tor-Projekten bekannt, die Tasten S1 und S2 verwenden, um den ESP32 manuell zu resetten und in den Bootloader-Mode zu versetzen.

Wenn Sie die fertig bestückten Platinen der Pflanzenleuchte im Elektor-Store kaufen, brauchen Sie sich nicht um die Firmware-Programmierung zu kümmern, aber ein Firmware-Update kann „OTA“ (Over The Air) durchgeführt werden. Es wird dann also kein Kabel oder Interface für ein Update benötigt.

Die I²C-Bussignale SDA und SCL stehen mit 3,3-V-Pegel für zukünftige Entwicklungen an K7 zur Verfügung. Es können Pull-up-Widerstände beim Anschluss von Peripherie an diesen Bus erforderlich sein. Bei 5-V-Peripherie braucht es eventuell 5/3,3-V-Pegelkonverter, da die ESP32-I/O-Pins keine 5 V vertragen! Die PWM-Signale des ESP32 sind an die DIM-Eingänge der LED-Treiber IC1... IC4 geführt. Deren Ausgangsstrom wird durch eine von drei Brücken (pro Treiber) auf 300, 350 oder 450 mA begrenzt. Bei Verbindung mit einem WLAN leuchtet LED1 grün, wenn der ESP32 in den Station-Mode geschaltet ist. Sie blinkt, wenn der ESP32 als Access Point betrieben wird. LED2 (rot) leuchtet, wenn ein OTA-Firmware-Update durchgeführt wird. Die gelbe LED3 zeigt an, dass eine MQTT-Verbindung hergestellt wurde.

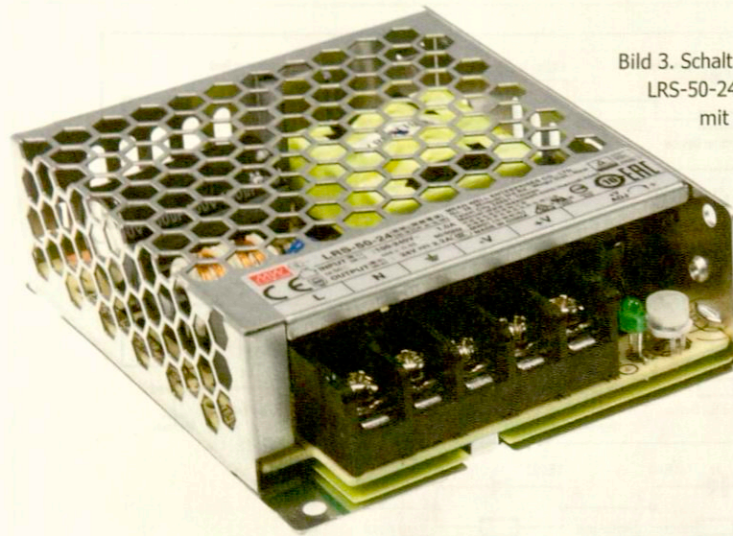


Bild 3. Schaltnetzteil des Typs LRS-50-24 von Mean Well mit 24 V und 2,2 A.

LED-Platine

Dies scheint der am wenigsten komplizierte Teil dieses Projekts zu sein, aber dieser Schein trügt, wenn man sich den einfachen Schaltplan in **Bild 2** ansieht. Zuerst mussten wir uns entscheiden, wie viele LEDs in welchen Farben die Pflanzenleuchte haben soll. Hierbei zogen wir die Expertise und das Wissen von Würth Elektronik zu Rate, um eine geeignete „LED-Mischung“ für die Pflanzenleuchte zu erhalten. Die resultierende LED-Platine reicht für die Beleuchtung einer kreisförmigen Fläche mit einem Radius von ca. 55 cm aus. In der Mitte ist es dabei natürlich heller als am Rand.

Nicht zu vernachlässigen: Leistungsfähige LEDs erwärmen sich. Würth Elektronik verfügt über Erfahrung damit, Platinen so zu entwickeln und zu produzieren, dass die LEDs so kühl wie möglich bleiben und möglichst wenig altern. Der Trick besteht darin, eine dünne Platine auf einen Aluminiumkühlkörper zu kleben - mehr dazu im **Kasten Wärmeabfuhr bei Platinen**. Diese technisch perfekte Lösung kostet natürlich etwas mehr als eine „normale“ Platine. Um dieses Projekt bezahlbar zu halten, wurde eine mit nur 0,5 mm Stärke besonders dünne Platine ohne Aluminium eingesetzt. Stattdessen hat sie große Kupferflächen auf beiden Seiten. Die LEDs der WL-SMDC-Serie verfügen über elektrisch neutrale Wärmepads, die auf der Bestückungsseite mit der Kupferebene verlötet werden, welche über thermische Durchkontaktierungen mit der Kupferebene auf der anderen Seite verbunden ist. Standardmäßig sind die Jumper auf der Treiberplatine für einen Strom pro Kanal von 300 mA durch die LEDs eingestellt. Auf diese Weise kommt die Pflanzenleuchte ohne Kühlkörper aus, und zudem kann die Helligkeit für manche Pflanzen auch reduziert werden.

Aufbau der Pflanzenleuchte

Die vorbestückten Platinen aus dem Elektor-Store begrenzen das Selbermachen auf das Konfigurieren der Jumper, das Verbinden der Platinen und den Anschluss einer Stromversorgung mit 24 V und 2,5 A Belastbarkeit. Es gibt zahlreiche fertige und preiswerte Netzteile, die passen. Das Prototyping erfolgte mit einem Labornetzteil, doch die EMV-Pre-Compliance-Tests wurden mit einem Netzteil des Typs BI60-240250-E2 von Egston durchgeführt. Von Mean Well gibt es ein

gut funktionierendes, elegantes und kleines Schaltnetzteil des Typs LRS-50-24 mit 24 V und 2,2 A (**Bild 3**).

Wenn Sie die Platinen selbst bestücken wollen, empfehlen wir, mit Lötpaste und einer Heißluftlötstation zu arbeiten. Ein LötKolben kann ausreichen, doch dann sollten Sie die Bauteile in der richtigen Reihenfolge platzieren, da größere Bauteile den Zugang zu den Pads der kleineren blockieren können. SMD-Bauteile kommen zuerst an die Reihe. Erst nach einer gründlichen Überprüfung der SMD-Lötstellen kommen die bedrahteten Bauteile!

Programmierung und Updates

Die vorbestückte Treiberplatine aus dem Elektor Store kommt mit fertig program-

miertem ESP32-Modul und ist betriebsbereit. Wenn Sie aber die System-Firmware ändern möchten, benötigen Sie die Arduino-IDE mit Unterstützung für den ESP32 auf ihrem PC. Das genaue Vorgehen wird unter [2] ausführlich erläutert. Sie müssen dann auch die Firmware für dieses Projekt von der ElektorLabs-Webseite herunterladen [3]. Die Kommentarzeilen am Anfang des Haupt-Sketches mit der Bezeichnung *Firmware.INO* enthalten eine Liste der Arduino-Bibliotheken, die für eine Kompilation des Quellcodes benötigt werden.

Ein eventueller USB-UART mit RTS- und DTR-Handshake-Signalen wird an K5 angeschlossen. Die Arduino-IDE ist dann in der Lage, Firmware und Sketch-Da-

ten automatisch hochzuladen. Wenn Sie einen UART ohne diese Handshake-Signale verwenden (z.B. ein an K6 angeschlossenes FDTI-Kabel), müssen Sie den ESP32 mit den Tasten S1 und S2 in den Bootloader-Mode versetzen, bevor die IDE Daten übertragen kann: S2 gedrückt halten, während Sie gleichzeitig S1 drücken und loslassen, dann S2 loslassen. In der Standard-Firmware ist Arduino OTA aktiviert, d.h. Sie können das System über WLAN ohne USB-Verbindung oder Tastendruck flashen. Verbinden Sie sich im Access-Point-Modus mit der Pflanzenleuchte oder lassen Sie die Leuchte mit Ihrem WLAN verbinden. Starten Sie die Arduino-IDE, öffnen Sie das Menü *Tools* und wählen Sie das *ESP32 WROVER Module* aus. Im Untermenü *Ports*

Wärmeabfuhr bei Platinen

Mit zunehmender Leistung und Miniaturisierung von Power-LEDs wird die Wärmeabfuhr bei LED-Anwendungen immer wichtiger. Um die thermische Belastung der LEDs innerhalb zulässiger Grenzen zu halten, braucht es ein effektives Wärmemanagement. Ziel muss es sein, die Temperatur der LED-Dies so niedrig wie möglich zu halten. Vertikale Wärmeleitung mittels sogenannter „Thermovias“ direkt unter der Kühlfläche der LED kann helfen. Ebenso wichtig ist die horizontale Wärmeleitung durch große Kupferflächen sowohl auf der oberen als auch auf der unteren Platinenseite. Beim Konzept von Würth Elektronik wird die so verteilte Wärme großflächig mit einem Transferkleber auf einen Aluminiumkühlkörper übertragen. Der besondere Vorteil gegenüber Metallkern-Platinen ist die Möglichkeit, Multilayer-Schaltungen aufzubauen. Thermovias sind Durchkontaktierungen, die der Wärmeübertragung durch die Platine dienen. Im Idealfall sollten diese Durchkontaktierungen direkt unter der Wärmequelle platziert werden, was eine direkte Wärmeabfuhr ermöglicht. Der Lochdurchmesser beträgt typischerweise 0,30 mm und die Wanddicke der Kupferhülse im Loch mindestens 25 µm. Auch größere Durchmesser sind möglich. Da Thermovias mit einem „Kupferdeckel“ verschlossen werden, wird das Einstromen von Lot verhindert und Bauteile können problemlos auf die Pads gelötet werden. Gute Eindrücke vom thermischen Widerstand und den Temperaturen auf und in der Platine bieten Simulationen oder FEM-Modelle. Neben der vertikalen Wärmeleitung wird die Wärme horizontal über eine größere Kupferfläche auf der Platine verteilt. Eine große Fläche ist auch eine wichtige Voraussetzung für eine gute vertikale Wärmeübertragung durch thermisch weniger leitfähige Materialien, etwa Isolationsschichten zwischen Platine und Kühlkörper oder zwischen Platine und Umgebung. Üblich sind die Verteilung der Wärme in Kupferschichten und die Befestigung der Platine auf einem Kühlkörper. Kühlkörper aus Aluminium haben sich auch bei Platinen bewährt. Eine gängige Methode ist das Pressen mit FR4-Prepreg (mit Glasgewebe verstärkte Kleberschichten) oder die Verwendung von Transferklebstoff. Der Vorteil der Verwendung des Klebstoffs gegenüber einem Prepreg ist das Verhalten beim Löten. Die unterschiedlichen Ausdehnungskoeffizienten von Aluminium und Platinenmaterial können so besser kompensiert werden. Bei der von Würth Elektronik eingesetzten Technologie wird der Kühlkörper unter Vakuum kalt mit der Platine verklebt. Diese Verbindung mittels Transferkleber wird als TWINflex-Technologie bezeichnet. Im Idealfall wird die Rückseite der Platine mit einer vollflächigen Kupferschicht versehen. Die Wärme wird auf dieser Ebene verteilt und anschließend durch den Transferkleber auf den Kühlkörper übertragen. Der thermische Widerstand der Anwendung wird durch die größere verfügbare Fläche reduziert. Ziel ist es, eine minimale Leiterplattendicke bei maximaler Kühlkörperoberfläche zu erreichen.



STÜCKLISTE TREIBERPLATINE

Widerstände:

(wenn nicht anders angegeben: 1%, 0,1 W, SMD 0805)
 R1,R4,R7,R10 = 3k48
 R2,R5,R8,R11 = 3k
 R3,R6,R9,R12 = 2k32
 R13..R16 = 10k, 5%
 R17..R19,R22..R26 = 1k, 5%
 R20 = 2k7, 5%
 R21 = 1k6

Spulen:

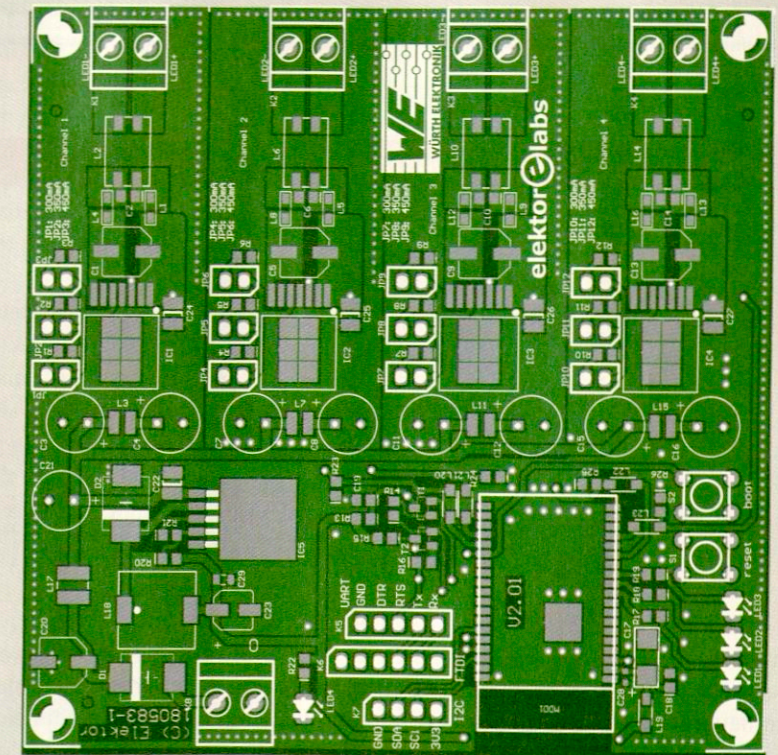
L1,L4,L5,L8,L9,L12,L13,L16 = EMV-Unterdrückungsferrit, 1,5 kΩ @ 100MHz, SMD 0805, WE-CBF
 L2,L6,L10,L14 = Gleichtakt-Netzfilter, 10 µH, 1,6 A, SMD, WE-SL2
 L3,L7,L11,L15 = Spule, 10 µ, 0,8 A, SMD 3521WE-PD2
 L17 = Spule, 2µ2, 2,5A, SMD 4532, WE-PD2
 L18 = Spule, 470 µ, 0,6 A, SMD 1050, WE-PD
 L19..L23 = Ferritperle, 31 Ω, 3 A, SMD 1206

Kondensatoren:

C1,C5,C9,C13,C20 = 4µ7, 100 V, 7,7 x 6,3 mm
 C2,C6,C10,C14,C18,C19,C29 = 100 n, 100 V, X7R, SMD 0805
 C3,C4,C7,C8,C11,C12,C15,C16,C21 = 27 µ, 100 V, 20%, 8 mm, radial
 C17 = 47 µ, 10 V, SMD 2312
 C22 = 4µ7, 50 V, X7R, SMD 1210
 C23 = 100 µ, 10 V, 5,5 x 5,5 x 5,5 mm
 C24..C27 = 2µ2, 100V, X7R, SMD 1210
 C28 = 1 n, 16 V, X7R, SMD 0603

Halbleiter:

D1,D2 = MBR5540, 40 V, Vf=550 mV @ If=5 A
 LED1 = LED, grün, 3 mm
 LED2 = LED, rot, 3 mm
 LED3 = LED, gelb, 3 mm, 3 mm
 LED4 = LED, blau, 3 mm
 T1,T2 = BC847C
 IC1..IC4 = Mag13C, LED-Abwärtsregler, Würth 172946001



60% der wahren Größe

IC5 = LM2576HVS-ADJ, Abwärtsregler
 MOD1 = ESP-32-WROVER-B

Außerdem:

S1,S2 = Taster, 12 V, 5 mA, 6x6 mm
 K1..K4,K8 = 2-pol. Schraubklemme, RM 0,2"
 K5 = 5-pol. Stiftleiste, vertikal, RM 0,1"
 K6 = 6-pol. Stiftleiste, vertikal, RM 0,1"
 K7 = 4-pol. Stiftleiste, vertikal, RM 0,1"
 JP1..JP9 = Jumper, 1x2, RM 0,1"
 4 x Jumper, 1x2, RM 0,1"
 Platine 180583-1 V2.01



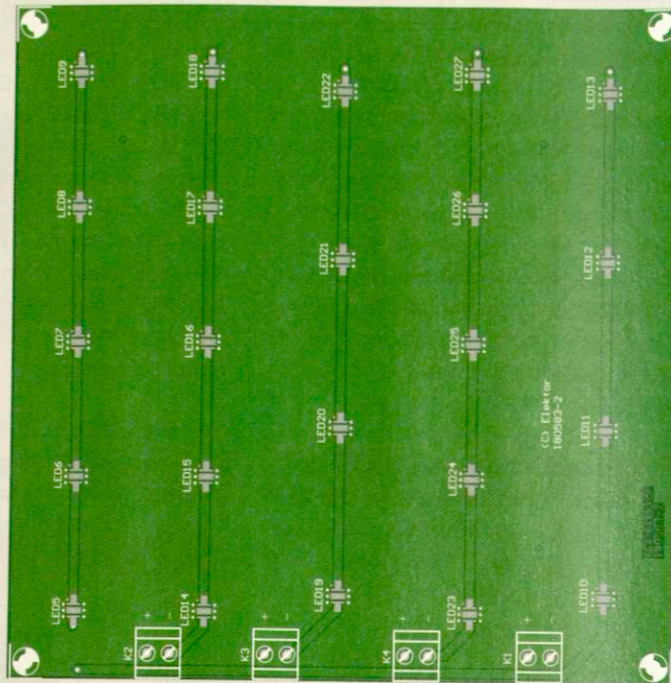
STÜCKLISTE LED-PLATINE

Halbleiter:

- LED5..LED13 = Hochleistungs-LED, rot 660 nm, WL-SMDC, 125°, 700 mA
- LED14..LED18 = Hochleistungs-LED, blau 450 nm, WL-SMDC, 125°, 700 mA
- LED19..LED22 = Hochleistungs-LED, rot 730 nm, WL-SMDC, 125°, 700 mA
- LED23..LED27 = Hochleistungs-LED, weiß, WL-SWTC, 120°, 121lm, 6000 K, 700 mA

Außerdem:

- K1..K4 = 2-pol. Schraubklemme, RM 0,2"
- Platine 180583-2 V1.1



60% der wahren Größe

Vorsicht: Augenschäden!

Um dauerhafte Schäden an Ihren Augen zu vermeiden, schauen Sie niemals direkt in die LEDs der Pflanzenleuchte oder andere LEDs hoher Helligkeit. Die Intensität des LED-Lichts ist auch bei reduzierter Leistung sehr hoch, Es gibt noch mehr zu beachten, wenn die Helligkeit von LEDs per PWM gesteuert wird, wie es bei diesem Projekt der Fall ist. Im Gegensatz zu beispielsweise Glühlampen kann die Intensität des LED-Lichts sehr schnell variieren. Eine scheinbar dunkle LED kann tatsächlich periodisch sehr kurze, aber sehr helle Lichtimpulse abstrahlen, und so auf das menschliche Auge dunkel wirken. Es gibt aber leider noch keine gesicherten Informationen, ob das für die Augen unschädlich ist. Und so lange das so ist, sollte man auf Nummer sicher gehen und einfach nie direkt auf LEDs hoher Leistung schauen, auch wenn sie scheinbar ausgeschaltet sind.

steht *HC-LED-XX-XX-XX-XX* als Port zur Auswahl. Dies ist der ESP32 der Pflanzenleuchte. Nach der Auswahl können Sie auf **UPLOAD** klicken, um den Code auf den Chip zu übertragen. Wenn der Upload funktioniert hat, wird der ESP32 neu gestartet und die neue Firmware ist installiert. Aber seien Sie gewarnt: Wenn Sie die Firmware modifizieren und Arduino OTA beschädigen, müssen Sie das System per Kabel neu flashen. Auch das Hochladen von ESP-Sketch-Daten kann per OTA durchgeführt werden. Hierzu klickt man auf *ESP32 Sketch Data Upload* im Menü *Tools*.

Software

Die parallele Entwicklung von Hard- und

Software bei diesem Projekt ermöglichte einen angenehmen Informationsaustausch, insbesondere hinsichtlich der verwendeten I/O-Pins. Einige der Pins wurden vertauscht, um das Routing der Leiterplatte etwas zu erleichtern. Auch wenn es sich hier nur um eine zu ändernde Codezeile handelt, kann dies viel Ärger beim Board-Layout ersparen. Zurück zur Software: Teuren Lesern dürfte kaum entgangen sein, dass in Elektor schon einige ESP32-Projekte veröffentlicht wurden. Als Entwickler fängt man nicht jedes Mal bei null an, sondern nutzt schon erstellte „Bausteine“ aus anderen Projekten und passt sie gegebenenfalls an. Für die Software der Pflanzenleuchte wurden Grundkompo-

nenten der Monster-LED-Uhr [4] und der ESP32-Wetterstation [5] recycelt, speziell der Webserver und grundlegende Routinen zur WLAN-Konfiguration. Für die Pflanzenleuchte wurden einige zusätzliche Module hinzugefügt. Zunächst wurde mit „ledc“ eine PWM-Einheit mit einigen Ergänzungen zum Ansteuern von LEDs realisiert. Sie ist einfach zu verwenden, indem man die gewünschte PWM-Frequenz und Auflösung vorgibt. Der LED-Treiber von Würth Elektronik basiert auf dem Dimm-Verhältnis. Es gibt eine minimale Einschaltzeit, die zu beachten ist. Wie bei jeder PWM sollte man die resultierende Auflösung bei gegebener PWM-Frequenz kennen. Der PWM-Takt beträgt 40 MHz. Die maximale Auflösung in Bit berechnet sich zu:

$$\log_2(40 \times 10^6 / 250) = 17,2$$

Dies führt zu 17 nutzbaren Bits für die PWM-Auflösung. Aufgrund der minimalen On-Time werden aber nur 8 Bit im Code genutzt. Die PWM-Einheit wird mit folgender Codezeile initialisiert:

```
ledcSetup(CHANNEL, PWM_FREQ, PWM_RES);
```

Damit wird einer von acht verfügbaren LED-Kanälen eingerichtet. Anders als bei anderen Systemen ist dieser Ledc-Kanal

nicht an einen speziellen ESP32-Pin gebunden. Man kann hier jeden möglichen Ausgangs-Pin verwenden. Die folgende Zeile legt diesen Pin fest:

```
ledcAttachPin(GPIO_PIN, CHANNEL);
```

Jetzt ist alles für die Ansteuerung der PWM-Pins bereit. Die folgende Zeile stellt einen Wert ein:

```
ledcWrite(CHANNEL, VALUE);
```

So kann man den aktuellen Wert eines bestimmten ledc-Kanals auslesen:

```
ledcRead(CHANNEL)
```

Das waren die Grundlagen für die Steuerung der *ledc*-Einheit. Im Softwaremodul befinden sich vier physikalische LED-Kanäle und ein fünfter als reine Software zur Steuerung der Gesamthelligkeit aller Kanäle. Im Webinterface wird dies als „intensity“ bezeichnet. Wenn Sie den Wert dieses Kanals ändern, wird er nicht sofort gesetzt. Stattdessen wird alle 100 ms eine Funktion von einem Timer aufgerufen und überprüft, ob der Sollwert gleich dem aktuell verwendeten ist. Wenn nicht, wird sie mit einer vorgegebenen maximalen Schrittweite erhöht oder verringert. Dies führt zu einem gewissen Fading bei der Änderung der Gesamthelligkeit. Beim Start wird es langsam heller. Auch das Zeit-Modul der Monster-LED-Uhr wird leicht modifiziert verwendet. Einige Pflanzen benötigen einen Tag- und Nachtzyklus. Hierfür wurde eine Funktion eingebaut, welche die LEDs für einen bestimmten Zeitraum pro Tag ausschalten kann.

Web-Interface

Wie Sie in den Screenshots in **Bild 4** sehen können, entstammt die UI (User Interface) ebenfalls früheren Elektor-Projekten. Die Hauptseite verfügt über fünf Felder für die Einstellungen der PWM. Die Kanäle 1 bis 4 sind die physikalischen Kanäle und *Intensity* steuert die Gesamthelligkeit. Plus oder Minus ändert einen Wert um 1 %. Alternativ man man auch einen Wert von 0...100 % eingeben. In der Hintergrundkommunikation erfolgt eine Wertänderung via Web-Sockets. Wenn mehr als ein User die Einstellungen ändert, werden alle anderen die neuen Werte im Browser sehen. Dies geschieht im Code mit einem separaten Web-Socket-Server, der JSON-Messages sendet und verarbeitet.

Ein weiterer Teil aktiviert oder deaktiviert den Zeitmodus für die LEDs.

Bei *Time settings* kann die interne Uhr manuell eingestellt werden oder ein NTP-Server ausgewählt werden, um die Systemzeit automatisch zu synchronisieren.

In den WLAN-Einstellungen kann man die Anmelde-Infos Ihres WLANs festlegen. Dies ermöglicht die Steuerung der Pflanzenleuchte mit jedem anderen Gerät, das an dieses LAN angeschlossen ist.

Der MQTT-Teil wurde von der Wetterstation übernommen und bietet eine Möglichkeit, die LED-Einstellungen und die automatische Ein- und Ausschaltzeit via MQTT-Broker zu steuern. Dies geschieht derzeit mit der Bibliotheken *PubSubClient* und *ArduinoJson* für den Datenaustausch. Um neue Einstellungen an das System zu senden, müssen Sie einen JSON-String erstellen, der so aussieht:

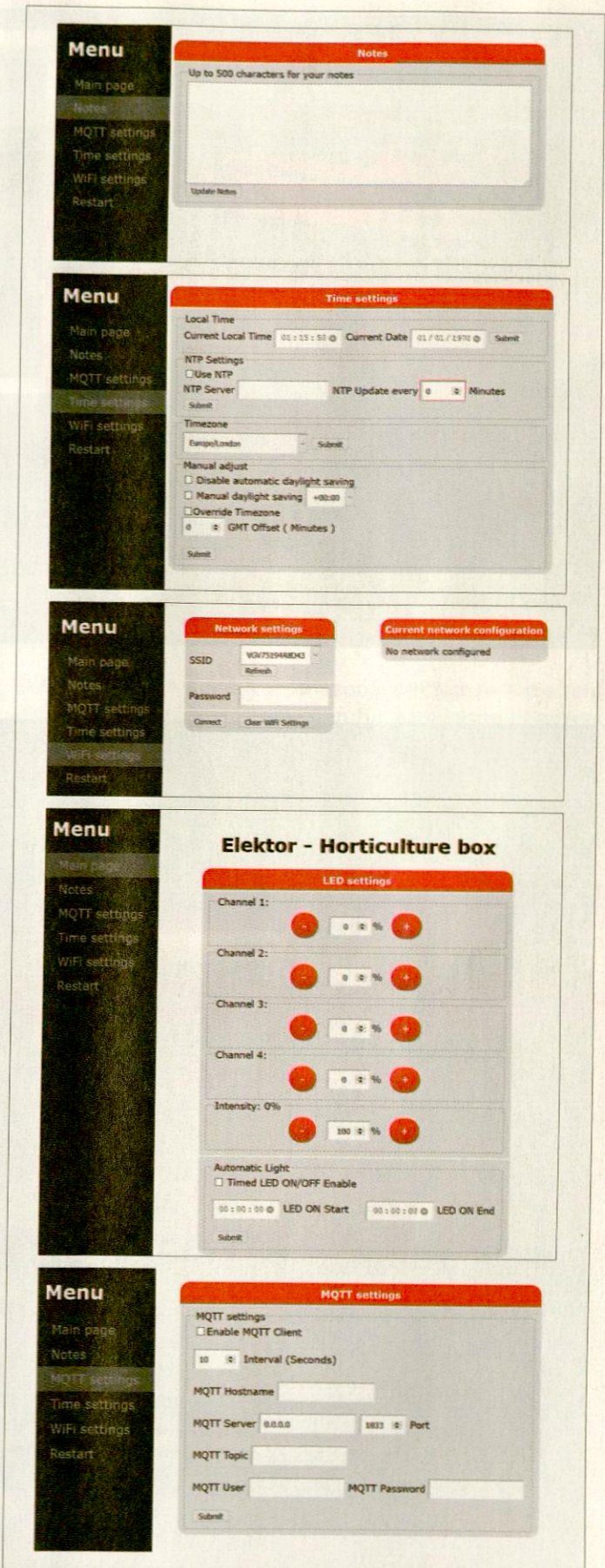


Bild 4. Screenshots der für die Pflanzenleuchte entwickelten Benutzeroberfläche: Hauptseite, MQTT-Seite, Notizen, Zeiteinstellungen und WLAN. Die Benutzeroberfläche ist ein Modul, das für zwei weitere Elektor-Projekte entwickelt wurde.



Bild 5. Das Gehäuse „HortiCoolture“ wurde von Würth Elektronik als Einzelstück zur Demonstration des Projekts auf Messen hergestellt. Der Rahmen wurde aus Sperrholz gefertigt, die Seitenteile sind aus Acrylglas und die Pflanzenschale aus verzinktem Stahl. Das „Dach“ ist abnehmbar und enthält die Elektronik sowie zwei kleine Lüfter zur Zwangskühlung.



Bild 6. Die neueste Version der „Horticulture Box“ kommt aus einem 3D-Drucker (Produkt und Bild: Würth Elektronik).

```
{
  "light": {
    "ch0": 6400,
    "ch1": 0,
    "ch2": 0,
    "ch3": 0,
    "intense": 32000
  },
  "timer": {
    "enable": false,
    "start": {
      "hour": 0,
      "minute": 0,
      "second": 0
    },
    "end": {
      "hour": 0,
      "minute": 0,
      "second": 0
    }
  }
}
```

Für `ch0` bis `intensity` sind vorzeichenlose 16-Bit-Werte erlaubt. Größere Zahlen werden automatisch auf 65535 begrenzt und das Minimum ist 0. Damit kann die Software *Node-RED* die Beleuchtung nicht nur mit einem Tageslichtzyklus, sondern auch über den Tag mit unterschiedlichen Farben steuern. Es müssen nicht alle Felder im String vorhanden sein. Um nur den Wert von `ch0` zu ändern, reicht:

Weblinks

- [1] Application Notes zu Horticulture-LEDs von Würth Elektronik: https://katalog.we-online.de/en/led/WL-SMDC_HORTICULTURE
- [2] Arduino-IDE für ESP32: www.elektormagazine.de/labs/esp32-getting-started
- [3] Firmware-Download zur Pflanzenleuchte: www.elektormagazine.de/180583-02
- [4] Monster-LED-Uhr: www.elektormagazine.de/labs/bedroom-clock-with-out-side-temperature-based-on-esp32
- [5] Wetterstation mit ESP32: www.elektormagazine.de/magazine/elektor-69/42262

```
{
  "light": {
    "ch0": 6400
  }
}
```

Dadurch wird nur der erste Kanal auf 6.400 gesetzt. Hierfür prüft der Code zur Verarbeitung der MQTT-Messages, ob einzelne Objekte enthalten sind. Bei Verwendung von *ArduinoJson* lautet der Code wie folgt:

```
JsonObject light = doc["light"];
JsonVariant light_ch0_var = light["ch0"];
If (false == light_ch0_var.isNull()) {
  .....
```

In der IF-Anweisung können wir das Vorhandensein eines Wertes für einen bestimmten Kanal überprüfen. Wenn er gefunden wird, wird er verarbeitet und ansonsten einfach ignoriert.

Wenn Sie den JSON-String mit eigenem Code verarbeiten müssen, sollten Sie überprüfen, ob sich die benötigten Werte wirklich im String befinden, da sonst unbeabsichtigt Standardwerte verwendet werden.

LED it grow

Unsere Pflanzenleuchte ist in ein Gehäuse eingebaut, das als Paludarium (Aquaterrarium für Sumpfflora) verwendet werden kann. Sie ist aber auch für andere mechanische Konstruktionen geeignet, welche die LED-Platine über Ihren Pflanzen positioniert. **Bild 5** zeigt Ansichten des von Würth Elektronik speziell für Werbeaktionen entwickelten Gehäuses „HortiCoolture“. Es hat zwei kleine Lüfter im oberen Teil, die bei der Kühlung des

LED-Boards helfen. **Bild 6** zeigt eine spätere Version, die von Würth Elektronik aus 3D-Druck-Teilen entworfen und hergestellt wurde.

Wenn der Selbstbau solch eines schönen Gehäuses nichts für Sie ist, können Sie auch auf (preiswertere) Alternativen zurückgreifen:

- Ein Aquarium oder Herbarium in einem Second-Hand-Laden abstauben.
- ein Besuch bei Ikea oder
- Angebote von entsprechend spezialisierten Läden.

Obwohl es eine Reihe von wissenschaftlichen Arbeiten über optimale Lichteinstellungen für bestimmte Pflanzen gibt, möchten wir Sie dazu ermutigen, Ihre eigenen „Licht-Rezepte“ herauszufinden. ◀

180583-02

Anzeige

HortiCoolture.

Let it grow!

#1 LEDITGROW

WE speed up the future

Electronic components
and printed circuit boards
for a successful growth!

www.wedirekt.com
www.we-online.com



IM ELEKTOR-STORE

→ Pflanzenleuchte Treiber und LED-Platine, vorbestückt, inklusive programmiertem Modul ESP-32-W-ROVER-B www.elektor.de/180583-71

→ Pflanzenleuchte Steuer-/Treiberplatine, nur Platine 180593-1 v.2.01 www.elektor.de/180583-1

→ Pflanzenleuchte LED-Platine, nur Platine 180593-2 v1.1 www.elektor.de/180583-2