



Jun / Juli 2017
ELV JOURNAL
Mehr Wissen in Elektronik

ELV JOURNAL

www.elvjournale.de
Mehr Wissen in Elektronik

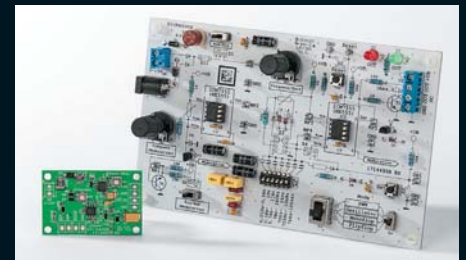
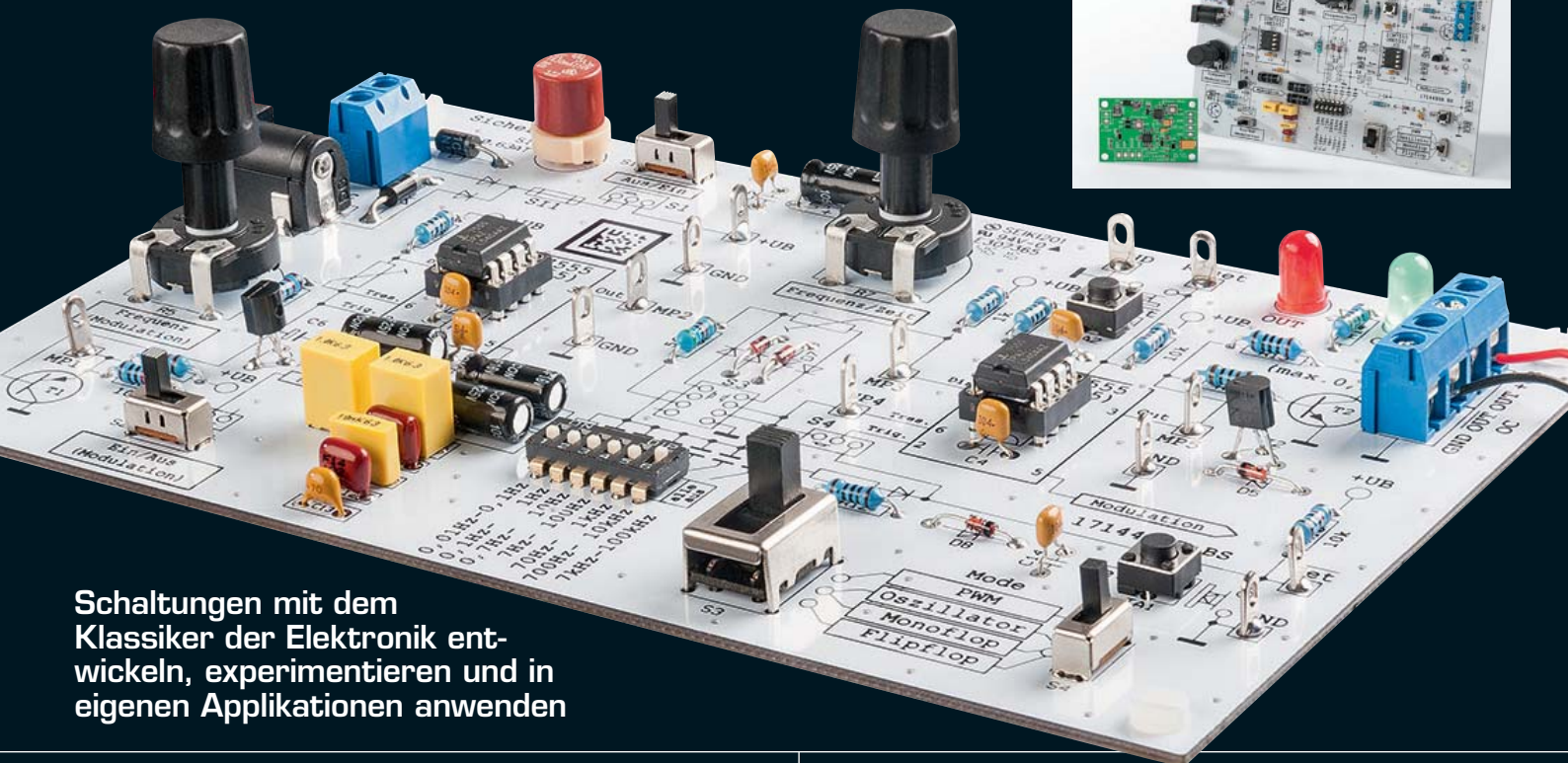
**MITMACHEN
& GEWINNEN**

**LESER TESTEN
UND GEWINNEN!**

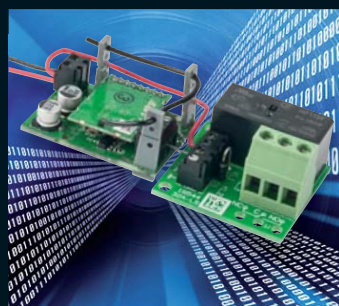


10x LED-Taschenlampe ELV X-BLAZE 800 und 5x Batavia Wowstick

Alleskönner der Elektronik NE555-Experimentier- und -Anwenderboard



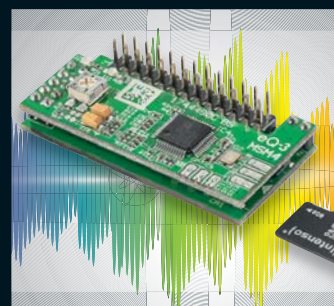
Schaltungen mit dem
Klassiker der Elektronik ent-
wickeln, experimentieren und in
eigenen Applikationen anwenden



Hausautomation

Homematic IP Schaltplatine

Kompakter 1-Kanal-Schaltaktor
mit Open-Collector- und Relais-
Schaltausgang für die einfache
Realisierung von Haustechnik-
Eigenbauprojekten.



Audioteknik

MP3-Soundmodul MSM4

Vielseitig einsetzbarer Mini-Player
mit integriertem Verstärker, ein-
facher Titelwahl, Stereo-Ausgang
und Einschaltklick-Unterdrückung.

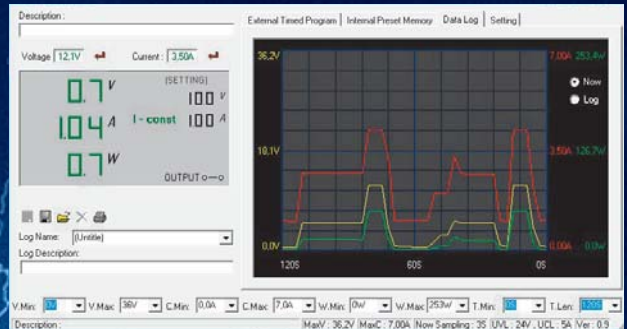
High-Power-Labornetzgeräte

Ausgangsstrom bis 10 A

Manson

Leichte und kompakte Bauform

Einfach programmierbar über
USB-Anschluss per PC-Software



Preis-Hit ab

€ 144,95

Das per USB fernprogrammierbare und per Spannungs-/Widerstandseingang fernsteuerbare Labornetzgerät liefert bis zu 60 Vdc bei einem Ausgangsstrom von bis zu 5 A.

- 3 frei programmierbare Speicher für Strom-/Spannungseinstellungen, ab Werk auf 5 V–13,8 V–25 V (bzw. 5 V–13,8 V–55 V bei Modell CN-10 67 67) und max. Strom eingestellt
- Gleichzeitige Anzeige von Spannung, Strom und Status über helle LED-Anzeigen
- Fernsteuerung für Strom-/Spannungseinstellung und Schalten des Ausgangs (0–5 V bzw. Poti)

- Aktive PFC
 - Integrierter temperaturgesteuerter Lüfter und Selbsttestfunktion
 - Automatische Tastensperre bei Softwarebetrieb
 - Galvanisch getrennter Ausgang, doppelter Anschluss vorn/hinten
 - Überlastschutz, Überspannungs- (OVP), Übertemperatur- und Kurzschluss-Schutz
 - Über USB und mitgeliefertes PC-Programm bis zu 20 Voreinstellungen für Strom, Spannung, Zeitintervalle (bis 100 min) und Zyklen (max. 999) programmierbar
 - Betriebssysteme: Windows 9x/2000/XP/Vista/7/8/8.1/10 (32/64 Bit)
- Abm. (B x H x T): 200 x 80 x 208 mm, Gewicht: 2,4 kg
Lieferung inkl. Software-CD, Netzkabel, USB-Kabel und Bedienungsanleitung

Modell	HCS-3102	HCS-3202	HCS-3204	HCS-3404
Eingangsspannung	230 V/50 Hz	230 V/50 Hz	230 V/50 Hz	230 V/50 Hz
Ausgangsspannung	1–36 Vdc	1–36 Vdc	1–60 Vdc	1–60 Vdc
Ausgangsstrom	0–5 A	0–10 A	0–5 A	0–10 A
Ausgangsleistung	180 VA max.	360 VA max.	300 VA max.	600 VA max.
Restwelligkeit (RMS/Spitze-Spitze)	5/50 mV	5/50 mV	5/50 mV	5/50 mV
Netzspannungsausregelung: Lastvariation: 0–90 %	50 mV	50 mV	50 mV	50 mV
Netzspannungsvariation: (90–264 VAC)	20 mV	20 mV	20 mV	20 mV
Stromausregelung: Last: 10–90 %	100 mA	100 mA	100 mA	100 mA
Anschluss	USB-B	USB-B	USB-B	USB-B
Best.-Nr.	CN-10 67 65	CN-10 67 66	CN-10 67 67	CN-12 52 18
Preis	€ 144, ⁹⁵	€ 169, ⁹⁵	€ 179, ⁹⁵	€ 259, ⁹⁵

Produktvideo



Flexibel laden – Multi-USB-Lader mit 8 USB-Ladeports und Ladeanzeige

NEU

526 V 000 A
36 17 mAh



CN-12 26 35

€ 49,95

Der Multi-USB-Lader ermöglicht das gleichzeitige Laden von bis zu 8 mobilen Geräten mit USB-Ladeanschluss. Die Ladeports sind in der Lage, den vom angeschlossenen Gerät benötigten Ladestrom automatisch zu detektieren und automatisch anzupassen.

4 Ports sind als Vorrangports ausgeführt, sie liefern stets jeweils 2 A, während 4 weitere Ports flexible Ladeströme entsprechend der Anforderung des angeschlossenen Geräts bereitstellen. Ein Display zeigt den Ladestrom, die Spannung und die Lademenge für jeden Port an. Das Gerät ist auch als allgemeine USB-5-V-Spannungsquelle bis 2 A einsetzbar.

- 60-W-Ladegerät mit 8x 5-V-USB-Ladeport, jeweils bis zu 2 A
- Automatische Anpassung an den Ladestrombedarf des angeschlossenen Geräts
- 4 Vorzugs-Ladeports mit je 2 A Ladestrom, 4 Ports mit flexiblem Ausgangsstrom
- Anzeige von Spannung, Ladestrom und Lademenge in mAh über beleuchtete LCD-Anzeige
- Ladeportabschaltung nach Ladeende
- Automatischer, stromsparender Stand-by-Modus nach Ladeschluss
- Kurzschluss-, Überspannungs-, Übertemperatur-, Überstrom-, Überlastschutz, Zurückfahren der Ausgangsleistung bei Überlast

Abm. (B x H x T): 120 x 85 x 34 mm

Lieber Elektronik-Freund,

verfolgen Sie auch mit Spannung, wie die verschiedenen Haustechniksysteme immer weiter zusammenwachsen? Offen gelegte Schnittstellen und findige Techniker machen es möglich, dass man heute als Anwender faktisch die freie Wahl hat, für eine Teillösung genau die gewünschte und passende Technik einsetzen zu können und die verschiedenen Lösungen in einem einzigen Bedien-Interface zusammenzufassen. Die rasante Entwicklung der Hausautomation hat niemandem die Zeit gelassen, einheitliche Normen zu schaffen. So bewältigen eben die Techniker und Softwareentwickler die Aufgabe – wie etwa die openHAB-Foundation mit der freien Software openHAB. Genau auf dieser Basis ist wieder ein Meilenstein gesetzt, nämlich die Zusammenführung des beliebten Alexa-Voice-Services mit unserem Homematic System. Wie dies funktioniert, lesen Sie in diesem ELV Journal.

Wenn es um den Bereich Smart Home geht, taucht auch immer wieder das Thema Sicherheit auf, vor allem in verschiedenen Medien mehr oder weniger qualifiziert und oft in reißerischer Weise diskutiert. Lassen Sie sich davon nicht anstecken. Wir haben in diesem ELV Journal das Thema „Sicherer Betrieb und sicherer Zugriff“ aufgegriffen und zeigen, wie man sein Homematic System wirklich sicher betreibt.

Unsere Elektronikprojekte haben es auch wieder in sich. Vor allem das neue MP3-Soundmodul, das dem äußerst erfolgreichen MSM2 folgt, wird wieder viele neue Freunde finden, zumal es nun noch praxisingerechter zu handhaben ist.

Das NE555-Experimentiersystem rückt einen scheinbaren Veteranen der Elektronik wieder in den Mittelpunkt. Zwar kann man faktisch alles, was der NE555 kann, auch mit einem kleinen Mikroprozessor lösen, aber erst einmal muss man begreifen, wie die zugrunde liegenden Schaltungsteile funktionieren – und genau da setzt unser NE555-Experimentiersystem an, das sich auch sehr gut für die Ausbildung eignet.

Ausbildung: Auch in diesem ELV Journal setzen wir unsere kleine Reihe zur Hochschulbildung fort. Dieses Mal geht es um eine hochaktuelle Ausbildungsrichtung: Energieeffizienz als Berufsfeld.



Viel Spaß beim Lesen und Nachbauen – und bleiben Sie neugierig!

Heinz-G. Redeker

Prof. Heinz-G. Redeker

ELV Newsletter abonnieren und € 5,- Bonus* sichern!

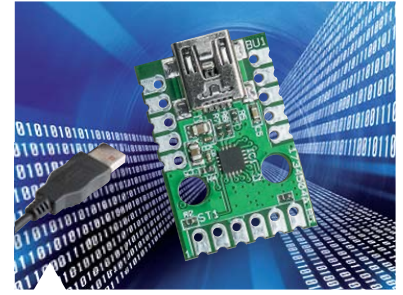
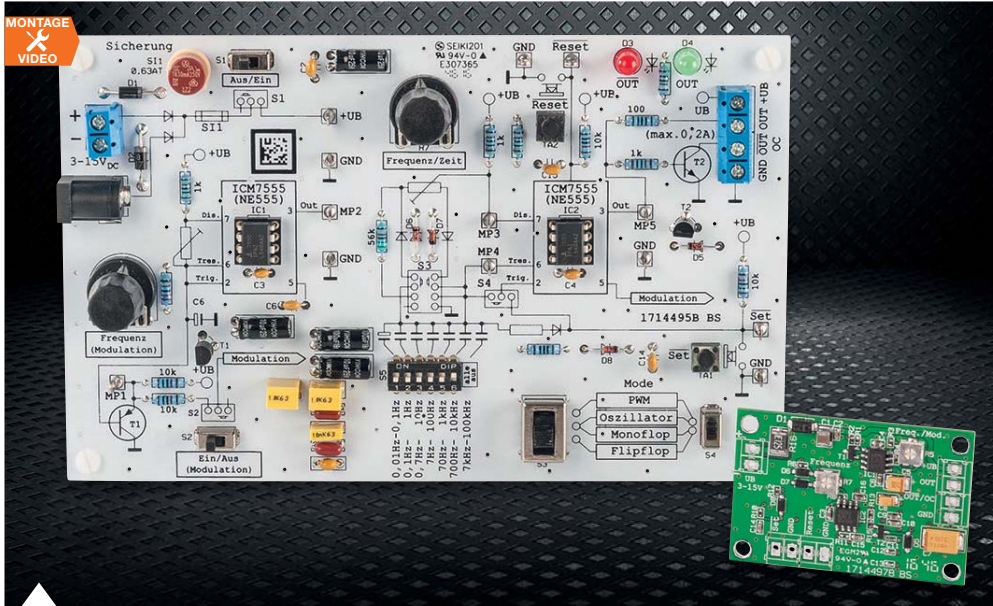
- ▶ **Neueste Technikrends**
- ▶ **Sonderangebote**
- ▶ **Tolle Aktionen und Vorteile**
- ▶ **Kostenlose Fachbeiträge**

und vieles mehr ...

*Sie erhalten einmalig € 5,- Bonus auf Ihre Bestellung, ab einem Warenwert von € 25,-. Der Gutschein gilt nicht in Verbindung mit anderen Aktionen und kann nicht ausbezahlt werden. Fachhändler und Institutionen, die bereits Sonderkonditionen erhalten, sind von diesem Bonus ausgeschlossen. Eine Auszahlung/Verrechnung mit offenen Rechnungen ist nicht möglich.



www.newsletter.elv.de ...at ...ch



69 Mini-USB-UART-Umsetzer

Der Nachfolger des UM2102 – mit modernem UART-USB-Schnittstellenwandler und 5-/3,3-V-Spannungsversorgung

54 NE555-Experimentier- und -Anwenderboard

Ein komplettes Experimentiersystem für das Erarbeiten von Schaltungslösungen mit dem vielseitig einsetzbaren NE555 und das Umsetzen der erarbeiteten Lösung auf einem kompakten SMD-Anwendungsboard

Audiotechnik

- 6 MP3-Soundmodul**
Soundfiles und Playlisten komfortabel abspielen

Hausautomation

- 29 Homematic IP Schaltplatine**
Universeller Aktor für Kleinspannung mit Relais- und Transistor-Schaltausgang

Messtechnik

- 54 NE555-Experimentier- und -Anwenderboard**
NE555 entdecken und anwenden

PC-Technik

- 69 Mini-USB-UART-Umsetzer**
Kompakter und vielseitiger Schnittstellenwandler mit CP2102N

So funktioniert's

- 15 Energieeffizienz als Beruf(ung)**
der Bachelor-Studiengang „Energieeffizienz“
- 20 Homematic Scriptprogrammierung**
Teil 4: Das CCU-Add-on CUXD und Scripte



- 84 Solarstromanlagen**
Kleinanlagen für jedermann



- 95 Arduino verstehen und anwenden**
Teil 22: Der I²C-Bus



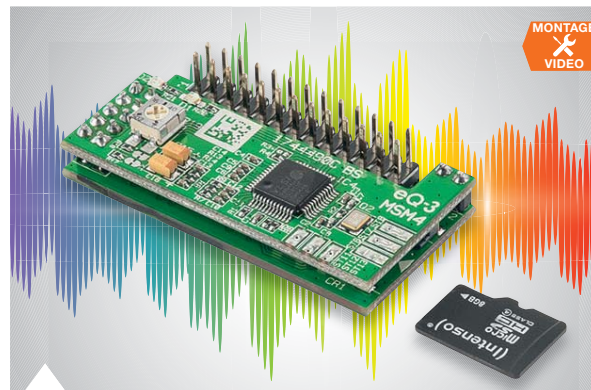
- 38 Raspberry Pi**
Teil 4: Ultraschallsensoren



29 Homematic IP Schaltplatine
 Kompakter Schaltaktor für den universellen Einsatz in eigenen Applikationen – mit Relais- und Open-Collector-Schaltausgang

Kostenlos

* Lieber Leser, den kompletten Teil 22 der Artikelserie „Arduino“ können Sie auf www.elv.de im Bereich des ELV Journals mit dem Webcode #10095 kostenlos herunterladen.



6 MP3-Soundmodul
 Äußerst universell einsetzbare Kombination aus MP3-Dekoder, vielseitigem Player und Verstärker



48 openHAB 2 mit neuen Möglichkeiten
 Smart Home Geräte über Amazon Alexa steuern

👍 So funktioniert's

- 38 Raspberry Pi**
Teil 4: Hören, sehen und fühlen mit Sensoren – Ultraschallsensor am Raspberry Pi
- 48 openHAB 2 mit neuen Möglichkeiten**
Alexa steuert Homematic und MAX!
- 74 Serielle Schnittstellen**
Teil 4: Die 1-Wire-Schnittstelle
- 84 Große Chance mit kleinen Solarstromanlagen**
Warum der Anschluss von Mini-Anlagen immer einfacher wird
- 88 Homematic Know-how**
Teil 21: Systemsicherheit für das Smart Home
- 95 Arduino verstehen und anwenden**
Teil 22: Der I²C-Bus – Grundlagen und Anwendungen*

★ Spezial

- 36 Experten antworten**
- 81 Leser testen und gewinnen**

☰ Rubriken

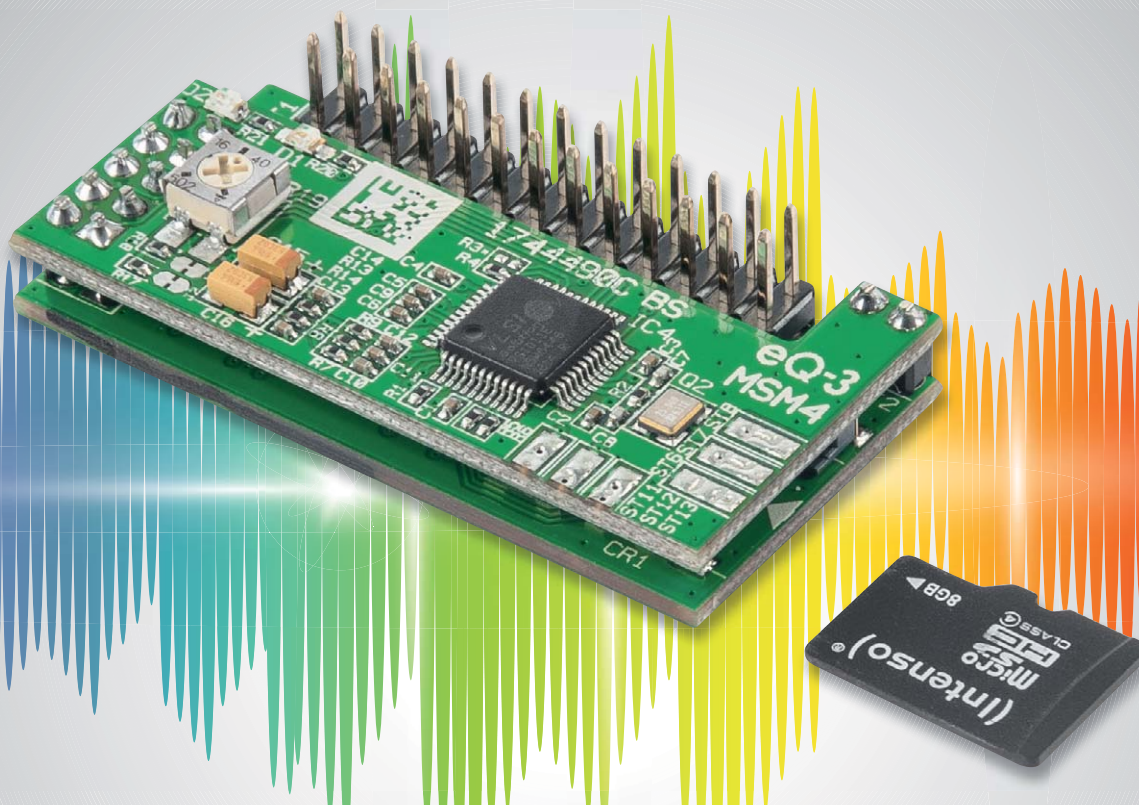
- 96 Die Neuen**
- 112 Service**
- 113 Bestellhinweise, Impressum**
- 114 Vorschau**



15 Energieeffizienz als Beruf(ung)
 Studiengang „Energieeffizienz“



36 Experten antworten
 Unsere Kundenberatung im Dialog



Noch besser – MP3-Soundmodul MSM4

Infos zum Bausatz

im ELV Shop

#10053

Das MP3-Soundmodul MSM2 ist einer der erfolgreichsten Bausätze von ELV – mit seiner Einsatzvielfalt und den vielen praktischen Features bei einem günstigen Preis ist es kaum schlagbar. Das neue MSM4 ist der direkte Nachfolger mit vielen neuen Features: vereinfachte Ansteuerung zur problemlosen Anbindung an eigene Schaltungen, verbesserte Speicherkartenunterstützung und ein Cinch-Anschlussmodul machen den Einsatz einfacher. Dazu kommen eine neue Softwarebibliothek mit verbessertem Funktionsumfang, neue Tastenmodi und eine Onboard-Klickunterdrückung für den NF-Ausgang.

Perfektionierter Mini-Player

Eine der Aufgaben bei der Entwicklung war es, dass das neue Modul in Form und Abmessungen dem MSM2 entsprechen sollte, so kann bei Bedarf ein direkter Austausch und damit eine Steigerung von Bedienkomfort und Leistungsfähigkeit gegenüber dem Vorgänger leicht erreicht werden. Bei den Abmessungen des Moduls wurde besonderer Wert auf ein möglichst schmales Design (nur 23 mm) gelegt, so ist es einfach in nahezu jede denkbare Applikation integrierbar, und es ist auf einfachste Weise die Wiedergabe von bis zu 10 unterschiedlichen Sound-Dateien oder Play-Listen möglich.

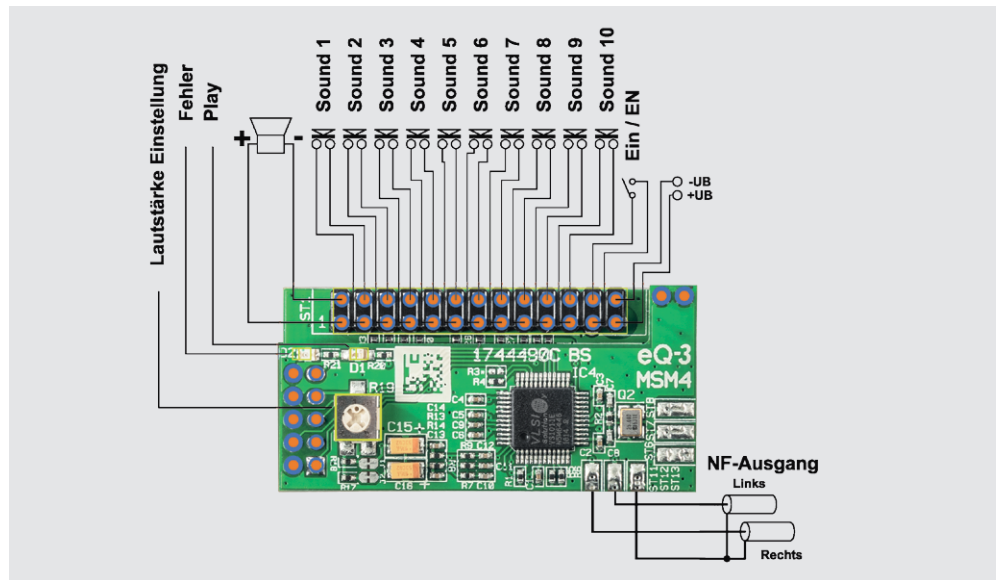
Die mögliche Länge der Sound-Dateien wird nur durch die Größe der verwendeten SD-Karte begrenzt, sodass auch die Wiedergabe von sehr langen Soundfiles kein Problem ist. Gerade auch diese Eigenschaft hat schon den Vorgänger so beliebt gemacht, ob als abrufbare oder automatische Erklärbox im Museum, als Memory mit Soundeffekten für Kleinkinder, Hintergrund-Musikplayer, Einschlaflied-Player oder Soundmodul für den Modellbau – alles möglich.

Wie auch beim MSM2 wird die Dekodierung von MP3-Files und die Erzeugung des analogen Audio-Signals durch einen eigenen MP3-Decoderchip ausgeführt, sodass die Vorbereitung der Sound-Dateien auf einem PC denkbar einfach ist. Unterstützt werden dabei Sound-Dateien mit dem Dateiformat MPEG 1.0 Layer3 (CBR, VBR, ABR) und bis zu max. 320 kbit/s oder WAV mit PCM- oder IMA-ADPCM-Codierung. Die Dateien selbst bedürfen keinerlei Vorbehandlung, sie müssen lediglich auf der Speicherkarte sortiert werden. Die Zuordnung der einzelnen Sound-Dateien zu den Tastenfunktionen des Moduls (Taste 1 bis 10) erfolgt sehr einfach durch eine simple Nummerierung am Anfang des Dateinamens (001-Dateiname, 002-Dateiname usw.).

Sämtliche Ein- und Ausgänge des Moduls stehen an einer 2-reihigen Stiftleiste mit insgesamt 26 Anschlusspins zur Verfügung. Zur Wiedergabe der verschiedenen MP3-Dateien können an die Stiftleiste direkt bis zu 10 Taster, die einfach gegen Masse schalten, angeschlossen werden. Über einen „Enable“-Eingang an dieser Stiftleiste, d. h. eben-



Bild 1: Bedien- und Anzeigeelemente sowie die Anschlussbelegung des MP3-Soundmoduls



falls Schalterkontakt nach Masse, wird auch der komplette „MP3-Player“ eingeschaltet, und zwei weitere Anschlusspins dienen zum Anschluss der Versorgungsspannung, die im Bereich von 4,5 bis 5,5 V liegen muss. Zwei weitere Anschlusspins dieser Stiftleiste dienen zum Anschluss eines 8-Ω-Lautsprechers an den auf dem Modul vorhandenen Mono-Leistungsverstärker, welcher 0,5 W an 8 Ω liefert.

Zusätzlich sind Löt pads vorhanden, an denen ein Stereo-Audio-Signal für einen externen Verstärker abgegriffen werden kann. Ein Trimm-Potentiometer und eine wählbare Grundverstärkung für den Onboard-Leistungsverstärker erlauben die einfache Lautstärke- bzw. Verstärkungseinstellung für den Onboard-Verstärker, für den Stereo-Audio-Ausgang ist zusätzlich die Lautstärkeeinstellung über die Bedientasten möglich. An den Anschluss für den externen (Stereo-) Verstärker ist das mitgelieferte Cinch-Buchsen-Board direkt oder per Kabelverbindung anschließbar, so ist auch dieser Anschluss vereinfacht.

Mit einer speziellen Schaltung wird bei diesem Modul der Einschaltklick wirksam unterdrückt und damit diese lästige Fehlergeräuschquelle eliminiert.

Die elektronische Steuerung der Tastereingänge kann neben direktem Anschluss von Tastern z. B. mithilfe von Transistoren, MOSFETs oder Optokopplern (z. B. CNY17) erfolgen. Alternativ kann man die Tastereingänge direkt mit Mikrocontroller-Ausgängen ansteuern, welche als

OpenDrain geschaltet sein sollten, denn auf dem Modul sind die notwendigen Pull-up-Widerstände zu 3,3 V bereits enthalten.

Achtung: An die Eingänge dürfen nicht mehr als 3,3 V angelegt werden.

Damit ist die Ansteuerung der Tastereingänge wesentlich einfacher realisierbar als beim MSM2, das ja bekanntlich über eine potentialfreie Tastenmatrix anzusteuern ist. So ist es u. a. nun auch möglich, eine Kombination aus Drehschalter zur File-Auswahl und Start-Taster einzusetzen.

In Bild 1 ist die gesamte Beschaltung des Moduls zusammengefasst. Auch bei den Bedien- und Abspielmöglichkeiten ist Neues hinzugekommen, so u. a. temporäres Abspiel nur, solange man eine Taste drückt, oder Zufalls-/Reihenmodus. Dies betrachten wir im folgenden Kapitel näher.

Bedienung

Zum Betrieb des MP3-Soundmoduls wird eine standardformatierte microSD-Karte (FAT32) benötigt und die gewünschten MP3-Sound-Dateien (Format siehe

Technische Daten	Geräte-Kurzbezeichnung:	MSM4
	Wiedergabeformat:	MPEG 1.0 Audio Layer 3 (MP3: CBR, VBR, ABR), WAV (PCM+IMA ADPCM)
	Bitrate:	bis 320 kbit/s
	Speichermedium:	microSD-Karte
	NF-Ausgangsleistung:	0,5 W an 8 Ω
	Wiedergabe:	auf Tastendruck bis zu 10 unterschiedliche MP3-Dateien
	Play-Listen:	auf Tastendruck bis zu 10 Listen mit jeweils bis zu 255 Dateien
	Tasteranschlüsse:	max. 10 – Achtung max. 3,3 V
	Statusanzeigen:	2 Leuchtdioden (rot und blau)
	Lautstärkeeinstellung:	Trimm-Potentiometer
	Lautstärkeabsenkung:	6 dB über Lötbrücke möglich
	Leitungslängen:	max. 1 m
	Versorgungsspannung:	4,5–5,5 Vdc
Stromaufnahme Betrieb:	50–500 mA (je nach Lautstärke)	
Stromaufnahme Stand-by:	2,5 mA (Idle-Modus), 50 µA (Sleep-Modus)	
Umgebungstemperatur:	-10 bis +55 °C	
Abmessungen:	46,5 x 23 x 12 mm	
Gewicht:	11 g	



vorhergehendes Kapitel) werden im ersten Schritt auf diese Karte kopiert.

Es werden max. 10 Dateien vom Soundmodul direkt über die Tasten unterstützt, wobei die Zuordnung zu den Tasten über eine am Anfang des Dateinamens stehende Nummer erfolgt.

Es können dabei durchaus beliebige Dateinamen vergeben werden, allerdings müssen Dateien, die abgespielt werden sollen, mit drei aufeinanderfolgenden Zahlen, 001-Dateiname bis 010-Dateiname bzw. 999-Dateiname, beginnen. Alle anderen MP3-Dateien auf der Karte, die durchaus vorhanden sein dürfen, werden ignoriert.

Eine kurze Betätigung der Taste 1 (< 2 s) führt zur Wiedergabe der Sound-Datei 001-Dateiname, und bei einer kurzen Betätigung der Taste 2 wird die Sound-Datei 002-Dateiname abgespielt usw.

Play-Listen

Eine Besonderheit ist die Möglichkeit, auf langen Tastendruck (> 2 s) hin komplette „Play-Listen“ abzuspielen. Dies sind Listen von Soundfiles, die in der gewünschten Reihenfolge in eine Textdatei eingetragen und nacheinander abgespielt werden. Dabei können bis zu 10 Play-Listen mit jeweils bis zu 255 MP3-Sound-Dateien je Liste angelegt und automatisch wiedergegeben werden.

Die Play-Listen haben die Namen „playlst0“ bis „playlst9“, wobei die Zuordnung der Tasten von TA1 bis TA10 entspricht, und den Dateityp „.txt“.

In den Play-Listen wird die Ziffernfolge der Sound-Dateien mit Semikolon „;“ getrennt aufgelistet. Wichtig ist dabei, dass in der Liste für jeden Eintrag ein Abschluss mit einem Semikolon erfolgt, da sonst die Datei nicht korrekt abgespielt wird. Über die Play-Listen können auch Dateien abgespielt werden, die nicht direkt über einen kurzen Tastendruck erreichbar sind (die Nummerierung ist möglich von 001 bis 999).

Beispiele mit Beschreibung:

001;

002 meine Beschreibung;

Zusätzlich sind auch Kommentareinträge möglich in der Form:

#Kommentar;

Weiterhin steht eine Wiederholfunktion (Repeat) als Endlosschleife zur Verfügung. Dies erfolgt durch Einfügen des Zeichens „<“ nach dem letzten Titel der Liste.

Ein Beispiel dazu:

001;

002;

<; #Dauerschleife von 001 und 002;

Ebenso kann man den Player veranlassen, Sprünge in einer Play-Liste auszuführen, indem hierzu das Zeichen „>“, gefolgt vom Sprungziel (1...255) an der gewünschten Stelle einzutragen ist.

Beispiel:

001;

002;

003;

004;

>2; #Springe zum 2. Eintrag in der Playlist (002);

005; #wird nie abgespielt;

Es werden die Soundfiles „001“ bis 004“ beim ersten Durchlauf wiedergegeben und danach werden die Soundfiles „002“ bis „004“ in einer Endlosschleife wiederholt. Ein weiteres Beispiel zeigt, wie man auch Files überspringen kann:

001;

002;

>5; #Springe zum 5. Eintrag -> 005;

003; #wird übersprungen;

004; #wird übersprungen;

005;

Schließlich kann man auch direkt zu einer anderen Play-Liste springen lassen. Dies wird durch Eintrag des Tilde-Zeichens (~) und Angabe der Ziel-Liste (1 bis 999) erreicht:

001;

002;

~1; #Öffnet playlst1.txt;

Natürlich besteht auch die Möglichkeit, in jeder Play-Liste nur die Nummer einer einzigen MP3-Datei zu speichern. Dann sind 10 MP3-Dateien über einen kurzen Tastendruck und 10 weitere MP3-Dateien über einen langen Tastendruck wiederzugeben.

Autoplay

Eine Besonderheit ist die Autoplay-Funktion. Ist diese aktiviert, wird eine „Autoplay-Liste“ automatisch abgespielt, sobald das MSM4 eingeschaltet wird. Dazu sind die entsprechenden Dateien wie bereits beschrieben in eine Play-Liste einzutragen, die mit „autoplay.txt“ bezeichnet ist. Hier können auch die beschriebenen Sonderfunktionen benutzt werden.

Eine interessante Anwendung der Autoplay-Funktion wäre z. B. das ferngesteuerte Einschalten über einen Funkschalter oder einen Melder, der das Netzteil zuschaltet oder den Enable-Eingang des MSM4 gegen Masse schaltet. So kann man z. B. einen Text oder Musik bei Auslösen eines Präsenzmelders starten. Das Abspielen ist dann so lange aktiv, wie der Melder eingeschaltet bleibt.

Eine andere Anwendung wäre zeitgesteuertes Fernschalten, z. B. via FS20 ZSU und FS20 ST oder Homematic Aktor, z. B. die Zwischenstecker-Schaltsteckdose für das Schalten des Netzteils oder den superkompakten und damit ebenso einfach in eigene Applikationen integrierbaren HM-LC-Sw1-PCB für das Schalten des Enable-Eingangs.

Random und Row

Ist dieser Modus aktiviert, werden über zwei weitere Listen „random.txt“ und „row.txt“ ein Zufallsmodus oder eine Wiedergabe der Reihe nach ermöglicht.

Beim Zufallsmodus wird durch Betätigen der Taste 1 aus der Zufallsliste ein Zufallseintrag ausgewählt.

Beim Reihenmodus wird durch Betätigen der Taste 2 der nächste Eintrag der Reihenliste abgearbeitet, wobei die Wiedergabe nach einem Sound jeweils auf eine erneute Betätigung wartet.

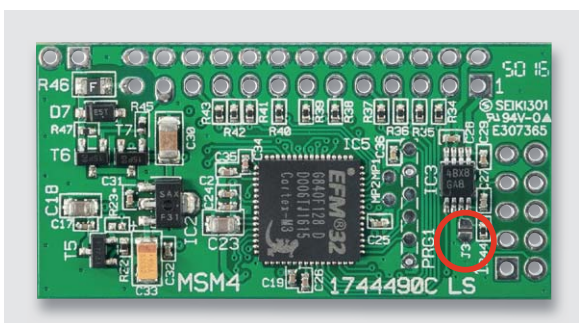


Bild 2: Die Lage der Lötbrücke J3 für die Verstärkungseinstellung der Endstufe



Tastenverhalten		
Tastenmodus	kurzer Tastendruck < 2 s	langer Tastendruck > 2 s
Mode 0 Normal (MSM2)	wenn gleicher Sound aktiv: stoppen wenn Playlist aktiv: nächsten Eintrag aus Playlist	Playlist starten wenn gleiche Playlist aktiv: stoppen
Mode 1 Retrigger	wenn gleicher Sound aktiv: neu starten wenn Playlist aktiv: nächsten Eintrag aus Playlist	Playlist starten wenn gleiche Playlist aktiv: Playlist neu starten
Mode 2 PlayWhilePressed	Taste drücken, Sound startet Taste loslassen, Sound stoppt Sound wird nur einmal abgespielt	
Mode 3 PlayWhilePressedLoop	Taste drücken, Sound startet Taste loslassen, Sound stoppt Sound wird am Ende wiederholt	
Mode 4 PlayToEnd	Sound einmal komplett abspielen, dabei keine Reaktion auf weiteres Betätigen Ausnahme: mehrere Tasten lange drücken	Playlist einmal komplett abspielen, dabei keine Reaktion auf weiteres Betätigen Ausnahme: mehrere Tasten lange drücken
Mode 5 RandomAndRow (nachtrIGGERbar)	TA1 startet einen zufälligen Sound aus der random.txt TA2 startet den nächsten Eintrag aus der row.txt TA3-TA9 startet zugehörigen Sound	TA10 Setzt die row.txt auf den ersten Eintrag zurück
Mode 6 RandomAndRowTo End	TA1 spielt einen zufälligen Sound aus der random.txt komplett ab TA2 spielt den nächsten Eintrag aus der row.txt komplett ab TA3-TA9 spielt zugehörigen Sound komplett ab keine Reaktion auf weiteres Betätigen Ausnahme: mehrere Tasten lange drücken	TA10 Setzt die row.txt auf den ersten Eintrag zurück

Tabelle 1

Über einen langen Tastendruck > 2 s der TA10 kann die Reihenliste wieder auf den ersten Eintrag zurückgesetzt werden.

Die Sprungbefehle „<“ „>“ und „~“ aus den normalen Play-Listen dürfen hier nicht verwendet werden.

Lautstärkeinstellung

Die Lautstärkeinstellung erfolgt für die integrierte Endstufe auf dem Modul mithilfe des Trimm-Potentiometers R19.

Darüber hinaus besteht die Möglichkeit, bei der Verstärkung der Endstufe zwischen zwei Einstellungen zu wählen (6 oder 12 dB). Die Verstärkung der Endstufe ist standardmäßig auf 12 dB eingestellt. Zur Auswahl der 6-dB-Einstellung sind die Flächen der Lötbrücke J3 (Bild 2) mit Lötzinn zu verbinden. In der 6-dB-Einstellung ist die maximale Ausgangsleistung der Endstufe um die Hälfte geringer. Das hat zur Folge, dass zwar die Gesamtlautstärke geringer ist, jedoch auch der maximale Stromverbrauch drastisch sinkt. Sollen aber Soundfiles wiedergegeben werden, deren Lautstärkepegel von sich aus sehr gering ist, empfiehlt es sich hier, die Einstellung 12 dB zu wählen.

Zusätzlich kann die Lautstärke des MP3-Decoders verändert werden. Bei Verwendung des NF-Ausgangs und Verwendung eines externen Verstärkers ist dies die einzige Einstellmöglichkeit auf dem Modul. Dies geschieht mit den Tasten 8, 9 und 10:

Tastenkombination 10 und 8 = leiser bzw.

Tastenkombination 10 und 9 = lauter

Durch langes Drücken beider Tasten (> 1 s) wird die Lautstärke schrittweise bis zum Loslassen der Tasten erhöht/verringert. Bei jedem Schritt der Lautstärkeänderung blitzt die rote LED kurz auf. Wenn dabei aktuell kein Sound abgespielt wird, gibt das Modul zusätzlich einen kurzen Bestätigungssound aus, um auch eine akustische Rückmeldung zu erhalten. Ist die höchste/geringste Einstellstufe erreicht, blitzt die rote LED zweimal auf.

In Tabelle 1 sind die bis hier besprochenen Tastenbetätigungen und -kombinationen zusammengefasst.

Die weiteren Tastenfunktionen

Über die folgend beschriebenen Tastenkombinationen (siehe auch Tabelle 2) lassen sich weitere Funktionen aktivieren. Die entsprechenden Tastenkombinationen sind dabei länger als 10 s gleichzeitig gedrückt zu halten, wobei die Übernahme jeweils durch kurzes Aufleuchten der roten LED und teilweise durch akustische Rückmeldungen quittiert wird.

Powermodi

Idle-Modus (Bereitschaft) & Sleep-Modus aktiviert

Tastenkombination: 1 und 5

Nach Abspielen einer Sound-Datei wird 10 s im Idle-Modus (blinkende blaue LED) auf eine neue Eingabe gewartet, bevor automatisch der Sleep-Modus (keine LED leuchtet) aktiviert wird.

Sleep-Modus aktiviert

Tastenkombination: 1 und 6

Nach Abspielen einer MP3-Datei wird sofort der Sleep-Modus aktiviert. Dessen Ruhestromaufnahme beträgt ca. 50 µA.

Idle-Modus aktiviert

Tastenkombination: 1 und 7

Im Idle-Modus wird nach Abspielen einer MP3-Datei auf neue Eingaben gewartet. MP3-Decoder und Verstärker werden im Idle-Betrieb abgeschaltet. Hier beträgt die Ruhestromaufnahme ca. 2,5 mA.

Idle-Modus deaktiviert

Tastenkombination: 1 und 9

MP3-Decoder und NF-Verstärker sind dauerhaft aktiv.

⚠ erhöhte Ruhestromaufnahme ca. 30 mA

Betriebsmodi

NF-Verstärker aktivieren/deaktivieren

Tastenkombination: 1 und 8

Bei Deaktivieren ist der NF-Verstärker in allen Powermodi komplett abgeschaltet. Dies ist für die Anwendung an einem externen Verstärker gedacht. Im aktivierten Zustand wird der Verstärker wie in den Powermodi beschrieben bei Bedarf an und abgeschaltet.

Tastenmodi:

Normales Abspiel (MSM2-Modus)

Tastenkombination: 2 und 3

Erneuter Tastendruck derselben Taste stoppt die Wiedergabe.

**Normal mit direktem Nachtriggern**

Tastenkombination: 2 und 4
Erneuter Tastendruck derselben Taste startet die Wiedergabe von vorn.

PlayWhilePressed

Tastenkombination: 2 und 5
Tastendruck: Sound einmal abspielen, solange die Taste gedrückt gehalten wird

PlayWhilePressedLoop

Tastenkombination: 2 und 6
Tastendruck: Sound in Dauerschleife abspielen, solange die Taste gedrückt gehalten wird

PlayToEnd

Tastenkombination: 2 und 7
Tastendruck: Sound/Playlist einmal komplett abspielen, dabei keine Reaktion auf weiteres Betätigen

Ausnahme: mehrere Tasten lange drücken

RandomAndRow

Tastenkombination: 2 und 8
Tastendruck TA1: Zufallssound aus der Liste random.txt
Tastendruck TA2: Sound aus der Liste row.txt nacheinander bei erneuten Tastendruck
Zurücksetzen auf Anfang über TA10
TA3–TA9: starte zugehörigen Sound
Erneute Bestätigung der Taste während des Abspielens startet den Sound neu bzw. startet den nächsten Sound.

RandomAndRowToEnd

Tastenkombination: 2 und 9
Wie Random und Row, nur mit dem Unterschied, dass während des Abspielens keine weiteren Tastendrucke angenommen werden.
Ausnahme: mehrere Tasten lange drücken

Entsprechend [Bild 1](#) ist die Betriebsspannung von 4,5 Vdc bis 5,5 Vdc an Pin 25 und Pin 26 der Stiftleiste ST1 anzuschließen. Eingeschaltet wird das Modul an Pin 23, d. h. Pin 23 ist im Betriebszustand mit

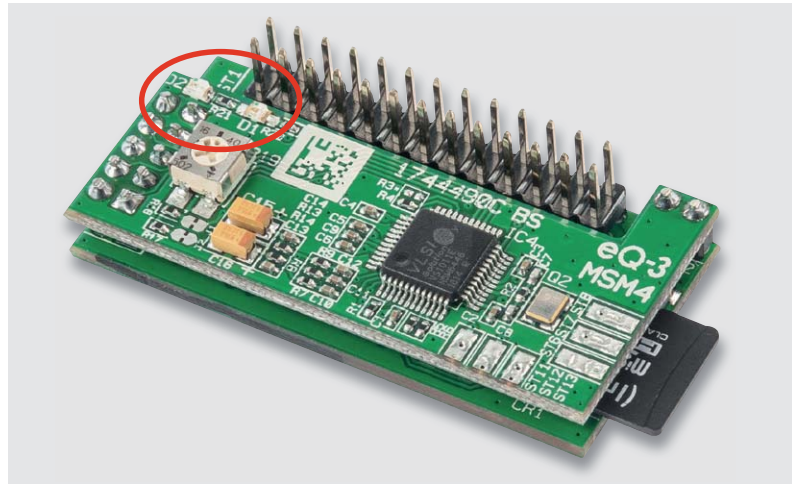


Bild 3: Die Lage der beiden LEDs D1 und D2

Pin 24 (Schaltungsmasse) zu verbinden. Natürlich kann das Einschalten auch über einen externen Open-Collector-Transistor an Pin 23 erfolgen. Die Taster zum Aufrufen der gewünschten Sound-Dateien sind an Pin 3 bis Pin 22 anzuschließen und an Pin 1 und 2 ist direkt ein 8-Ω-Lautsprecher anzuschließen.

LED-Anzeigen

Die beiden SMD-Leuchtdioden D1 und D2 (siehe Kennzeichnung in [Bild 3](#)) dienen zur Anzeige der verschiedenen Betriebszustände des MP3-Soundmoduls.

Beim Starten des Moduls leuchten beide LEDs gleichzeitig kurz auf.

Die dauerhaft leuchtende blaue LED D1 zeigt die Wiedergabe einer Audio-Datei (Play-Modus) oder, langsam blinkend, den Idle-Modus an. Schnelles blaues Blinken signalisiert, dass keine SD-Karte eingesetzt ist

Die rote LED D2 dient der Fehleranzeige und für Quittierungssignale, wie nachfolgend beschrieben.

Fehlercodes:

Ein längeres Blinken der roten LED signalisiert zunächst den Start der Fehleranzeige. Dem folgt ein- oder mehrmaliges Aufblitzen, wobei die Anzahl der Blitze den Fehlercode beschreibt:

Tastenkombinationen bei Drücken > 10 s

Tasten-kombination	Funktion	Tasten-kombination	Funktion
1+2	beim Starten gedrückt -> Update		
1+3	akustische Versionsnummernausgabe	2+3	Tastenmodus 0 Normales Abspiel (MSM2-Modus)
1+4	Autoplay ein/ausschalten	2+4	Tastenmodus 1 Normal mit direktem Nachtriggern
1+5	Powermode Idle + Sleep	2+5	Tastenmodus 2 PlayWhilePressed
1+6	Powermode Sleep	2+6	Tastenmodus 3 PlayWhilePressedLoop
1+7	Powermode Idle	2+7	Tastenmodus 4 PlayToEnd
1+8	Audioverstärker Ein/ausschalten	2+8	Tastenmodus 5 RandomAndRow
1+9	Powermode Idle deaktiviert (High-Power mode)	2+9	Tastenmodus 6 RandomAndRowToEnd
1+10	Update starten	2+10	Werkseinstellungen wiederherstellen

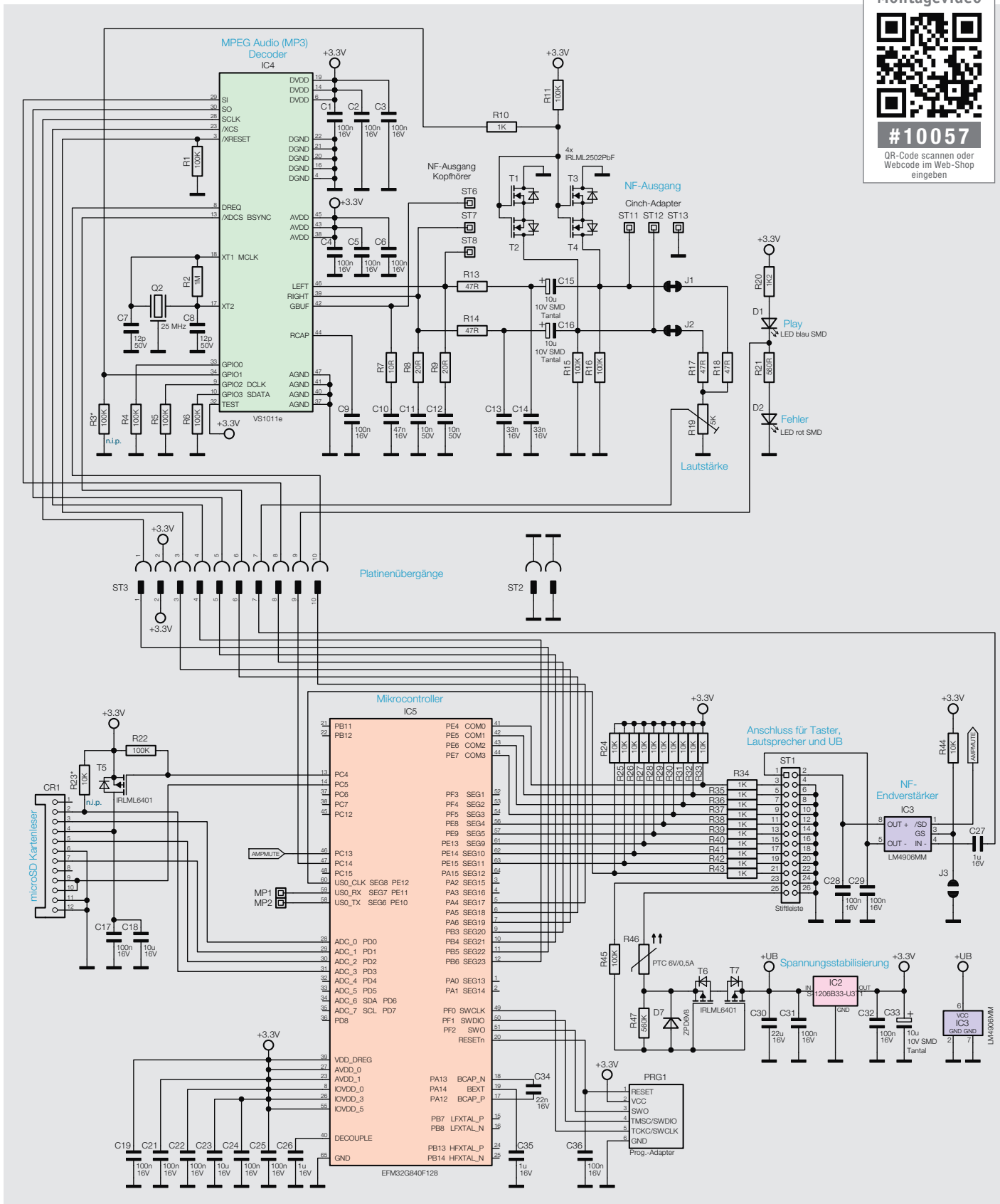


- Blinken 1x lang, 1x kurz: SD-Karte nicht gefunden, Dateisystem fehlerhaft
 - Blinken 1x lang, 2x kurz: Sound-Datei nicht gefunden oder fehlerhaft
 - Blinken 1x lang, 3x kurz: Play-Liste nicht gefunden
 - Blinken 1x lang, 4x kurz: Playlist-Eintrag ungültig
- Diese Sequenzen werden 3x wiederholt und können mittels Tastendruck abgebrochen werden.

Schaltung

Das Schaltbild (Bild 4) besteht aus zwei Teilen, welche auch die Anordnung auf den jeweiligen Platinen widerspiegeln.

Im oberen Teil sind der MP3-Decoder IC4 mit der Klick-Unterdrückung, der Lautstärkeeinstellung und die beiden LEDs zu sehen.



Montagevideo



#10057

QR-Code scannen oder
Webcode im Web-Shop
eingeben

Bild 4: Das Schaltbild des MSM4

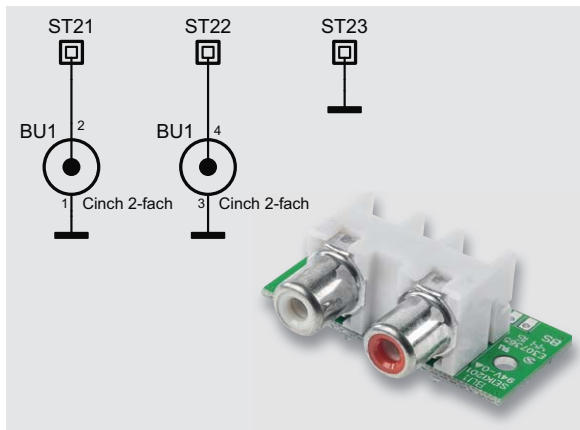


Bild 5: Über die Cinch-Buchse BU1 ist ein einfacher Anschluss eines externen Verstärkers möglich.

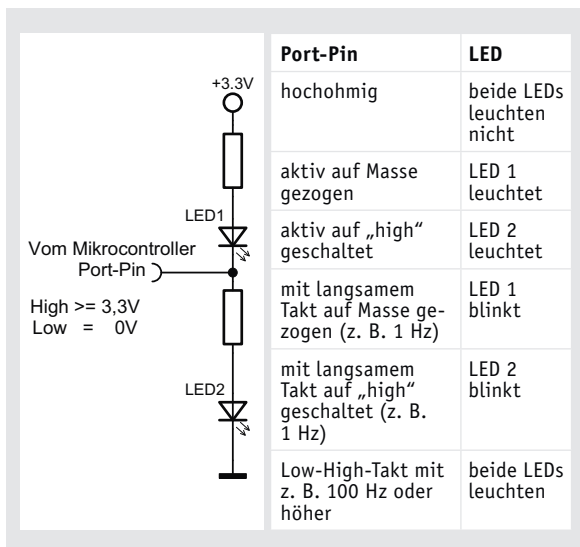


Bild 6: So erfolgt die Ansteuerung zweier LEDs über nur einen Portpin.

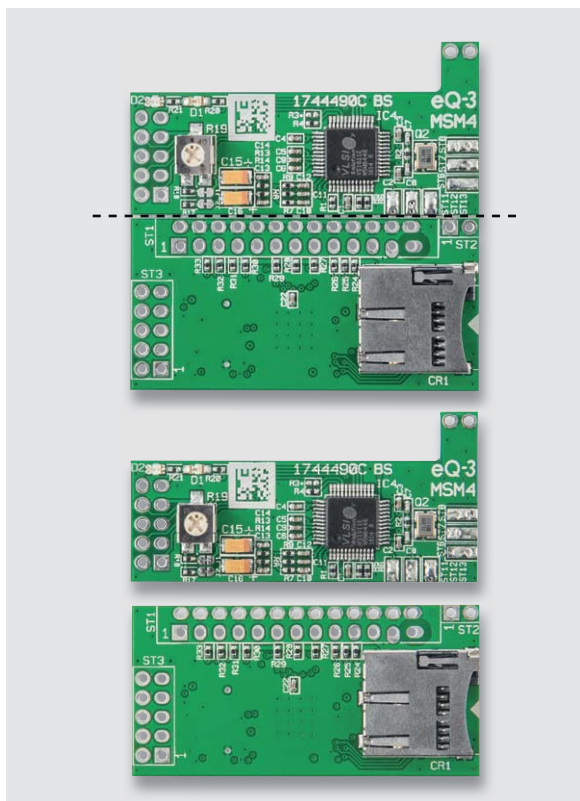


Bild 7: Die Platine ist entlang der perforierten Linie in zwei Teile zu brechen.

Der MP3-Decoder arbeitet durch den Quarz Q2 mit einem Takt von 25 MHz. An den Kontakten ST6 bis ST8 steht das Stereo-Audio-Signal, wie vom MSM2 bekannt, für einen Kopfhöreranschluss ohne zusätzliche Beschaltung zur Verfügung.

Jedoch sollten zum Anschluss an externe Verstärker die neuen Kontakte ST11 bis ST13 verwendet werden, hierfür wird zusätzlich eine Adapterplatine auf Cinch-Kontakte (Bild 5) mitgeliefert, diese sollte über ein kurzes geschirmtes Kabel mit dem MSM4 verbunden werden.

Diese Kontakte besitzen im Gegensatz zum ST6 bis ST8 keinen Gleichspannungsanteil, sodass zusätzliche Beschaltungen mit externen Kondensatoren überflüssig werden. Bei Verwendung des Stereo-Signals für externe Verstärker sollten man die Lötjumper J1 und J2 mit einem Cutter auftrennen.

Der nachfolgende Verstärker kann, wie bereits im Kapitel „Bedienung“ beschrieben, dann zur Stromersparnis auch komplett deaktiviert werden.

Zusätzlich werden diese Ausgänge im Einschaltmoment über die MOSFETs T1 bis T4 gegen Masse geschaltet, sodass das vom MP3-Decoder produzierte Einschaltknacken unterdrückt wird.

Da es sich um ein analoges Audio-Signal handelt, ist ein störungsfreies Schalten nur mittels zweier antiseriell geschalteter FETs möglich, da ansonsten die parasitären Dioden Einfluss auf das Signal nehmen würden.

Das Stereo-Signal wird über die Widerstände R17 und R18 zusammengeführt und dann über das Poti R19 zur Lautstärkenanpassung zum NF-Verstärker IC3 im unteren Schaltungsteil geführt.

Die LEDs D1 und D2 sind in Reihe geschaltet, die Flussspannungen ergeben zusammen einen so großen Wert, dass die LEDs im Normalzustand bei 3,3 V Versorgungsspannung nicht leuchten, sie leuchten erst durch Schalten des daran angeschlossenen Controllerpins von Tristate (hochohmig) auf aktiv high (rot leuchtet) oder aktiv low (blau leuchtet) auf. Um beide LEDs leuchten zu lassen, wird schnell zwischen aktiv low und high gewechselt, siehe Bild 6. Mit diesem Schaltungstrick lassen sich zwei LEDs über einen Controllerpin ansteuern.

Im unteren Teil der Schaltung sind der microSD-Kartenschacht, der Hauptcontroller IC5, der NF-Verstärker IC3, die Spannungsversorgung sowie die Anschlussleiste zu sehen.

Der PTC R46 begrenzt den Stromfluss im Fehlerfall, die nachfolgenden MOSFETs T6 und T7 übernehmen zwei Funktionen, zum einen den Verpolungsschutz, sodass sie bei verpolt angeschlossener Betriebsspannung sperren, zum anderen die Deaktivierung des gesamten Moduls, solange der EN-Pin nicht gegen Masse geschaltet wird. So nimmt das nicht aktivierte Modul keinen Strom auf.

Die Spannungsversorgung der microSD-Karte wird über den MOSFET T5 geschaltet, wobei die Kondensatoren C17 und C18 zur Pufferung dienen.

Der Verstärker IC3 wird direkt aus der Versorgungsspannung betrieben, sein Ausgang liegt an den Pins 1 und 2 von ST1.

Der Controller IC5 und die restliche Schaltung werden mit 3,3 V versorgt, dies übernimmt der Linearregler IC2.

Die Tasteneingänge sind mit Pull-up-Widerständen gegen die 3,3-V-Versorgungsspannung versehen. Die Reihenwiderstände R34 bis R43 begrenzen den Stromfluss gegen die extern anzuschließende Schaltung.

An den Eingängen sollten nur OpenDrain/OpenCollector-Ausgänge, Relais oder Taster angeschlossen werden, denn sie sind nur 3,3-V-tolerant.

Nachbau

Im Auslieferungszustand wird eine Platine geliefert, die entlang der eingekerbten Sollbruchlinie einfach in zwei Teile zu brechen ist (Bild 7). Die Platine wird bereits komplett mit bestückten SMD-Bauteilen geliefert, und da, abgesehen von den Stiftheisten, ausschließlich SMD-Komponenten zum Einsatz kommen, bleiben nur noch wenige Aufbau-schritte bis zum fertigen MP3-Soundmodul. Der weitere Aufbau wird durch die Platinenfotos und die Bestückungspläne entsprechend Bild 8 unterstützt.

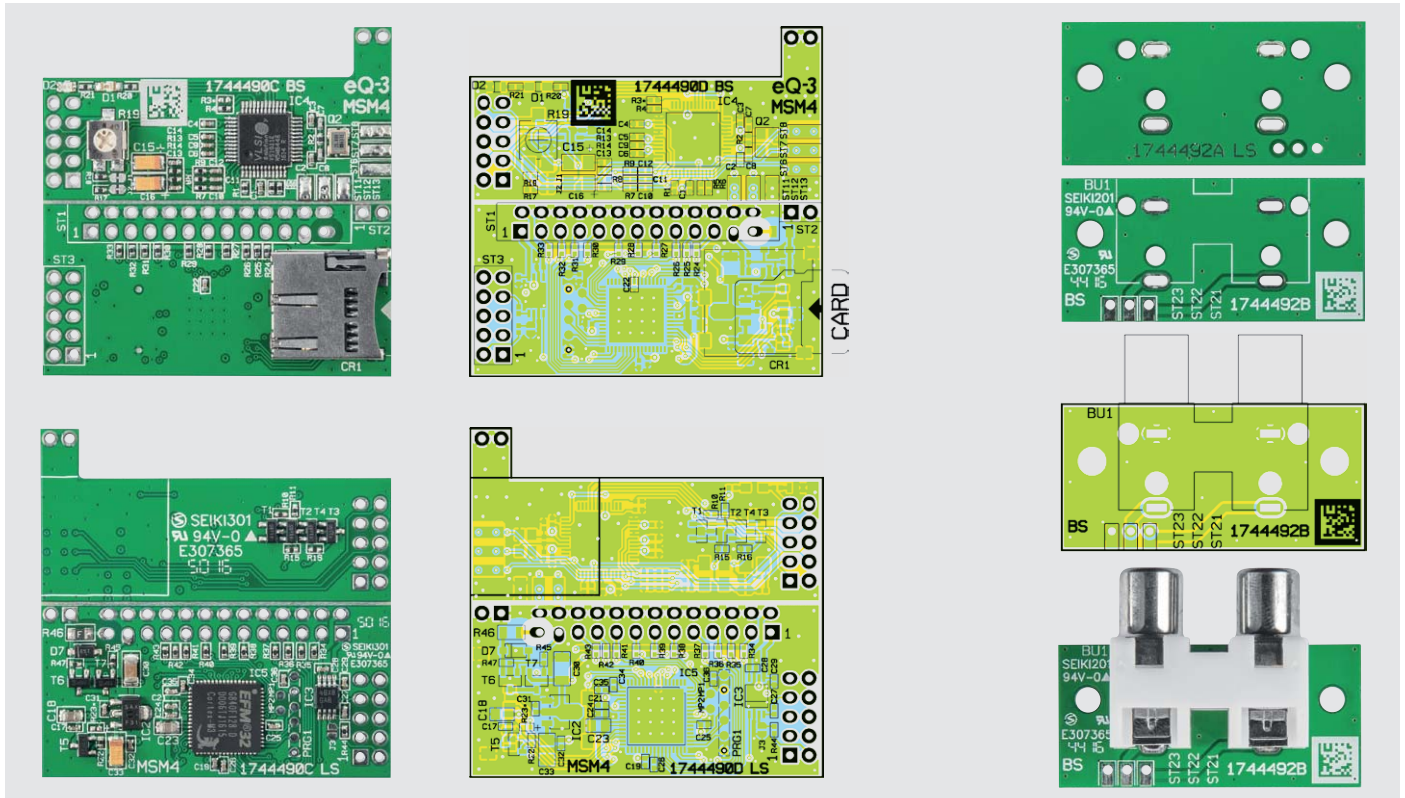


Bild 8: Die Platinenfotos mit den zugehörigen Bestückungsplänen für die MP3-Player-Platinen und die Cinch-Adapterplatine



Wichtiger Hinweis:

Für einen ausreichenden Schutz vor elektrostatischen Entladungen ist der Einbau in ein geeignetes Gehäuse erforderlich, damit die Schaltung nicht durch eine Berührung mit den Fingern oder Gegenständen gefährdet werden kann.

Nach dem Trennen der Platine erhalten wir dann eine Prozessorplatine mit microSD-Kartenschacht und eine Decoderplatine mit Potentiometer.

Auf der Prozessorplatine werden nun eine 26-polige, eine 10-polige und eine 2-polige Stiftleiste bestückt und an der SMD-Seite sorgfältig verlötet. Beim Verlöten ist darauf zu achten, dass die Kunststoffgehäuse der Stiftleisten plan auf der Platinenoberfläche aufliegen müssen. Bild 9 zeigt die fertig bestückte Prozessorplatine mit Blick auf den microSD-Kartenleser.

Entsprechend Bild 10 werden die beiden Platinen zusammengesetzt und verlötet, wobei auf eine gerade Ausrichtung (gleichmäßiger Abstand der beiden Platinen zueinander) zu achten ist.

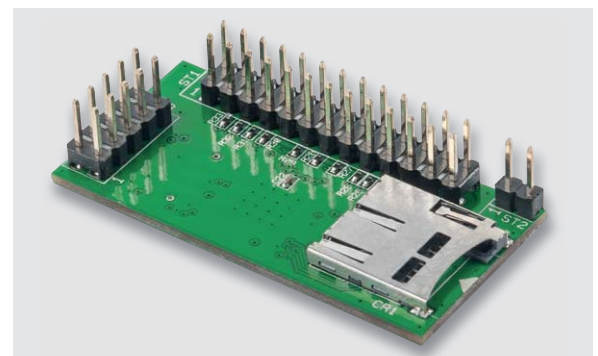


Bild 9: Die mit den Stiftleisten bestückte Platine des MSM4



Wichtiger Hinweis:

Zur Gewährleistung der elektrischen Sicherheit muss es sich bei der speisenden Quelle um eine Sicherheits-Schutzkleinspannung handeln.

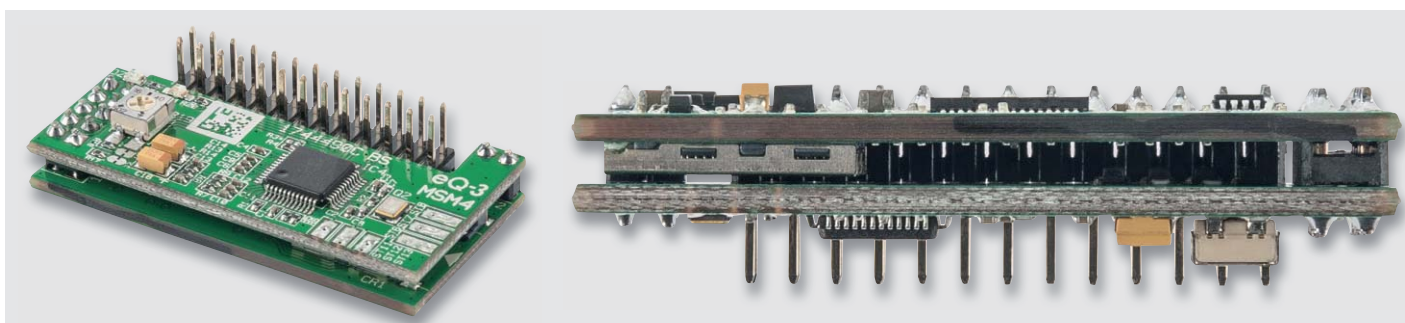


Bild 10: Beim Zusammenlöten der beiden Platinen über die Stiftleisten ist auf eine gerade Ausrichtung zu achten. Vor dem Verlöten ist es ratsam, die Pins der Stiftleisten ST2 und ST3 zu kürzen.

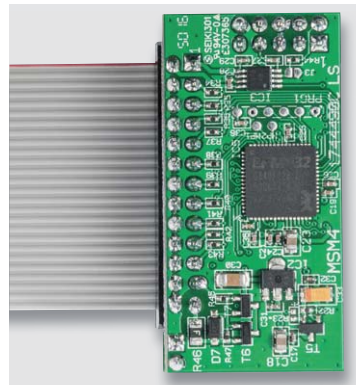


Bild 11: Sämtliche Anschlüsse des MP3-Soundmoduls können über einen Flachbandkabel-Steckverbinder hergestellt werden.

Dazu ist es ratsam, die Stifteleisten ST2 und ST3 etwas zu kürzen.

Das MP3-Soundmodul ist damit bereits vollständig aufgebaut. Wie in Bild 11 zu sehen, können die Verbindungen zur Stifteleiste ST1 einfach mithilfe eines entsprechenden Flachbandkabels hergestellt werden. Der Cinch-Adapter besteht nur aus der Platine und der Cinch-Buchse, welche plan auf die Platine zu löten ist (Bild 12).

Der Cinch-Adapter sollte über eine geschirmte Leitung angeschlossen werden, aber auch die direkte Verbindung über eine Stifteleiste ist möglich, dabei müssen die Pins der Stifteleiste auf der längeren Seite gekürzt werden, da sie sonst Kurzschlüsse auf den Platinen verursachen könnten. **ELV**

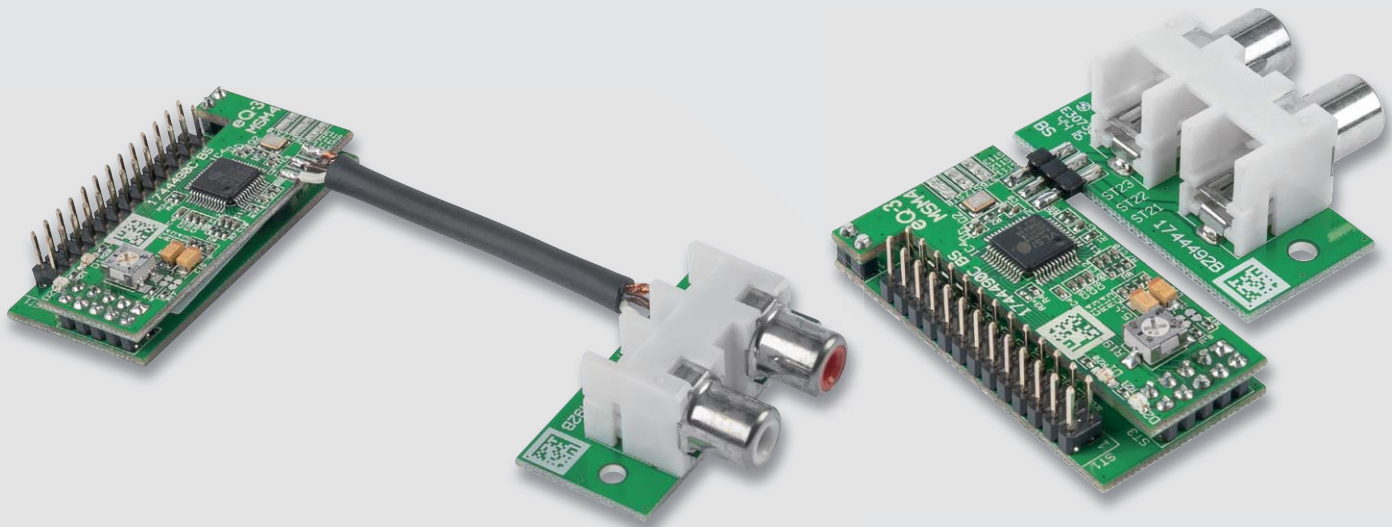


Bild 12: Die Cinch-Adapterplatine kann entweder über ein geschirmtes Kabel oder direkt an das MSM4 angeschlossen werden.

Widerstände:

10 Ω/SMD/0402	R7
20 Ω/SMD/0402	R8, R9
47 Ω/SMD/0402	R13, R14, R17, R18
560 Ω/SMD/0402	R21
1 kΩ/SMD/0402	R10, R34–R43
1,2 kΩ/SMD/0402	R20
10 kΩ/SMD/0402	R24–R33, R44
100 kΩ/SMD/0402	R1, R4–R6, R11, R15, R16, R22, R45
560 kΩ/SMD/0402	R47
1 MΩ/SMD/0402	R2
SMD-Cermet-Trimмер/5 kΩ	R19
PTC/0,5 A/6 V/SMD/0805	R46

Kondensatoren:

12 pF/50 V/SMD/0402	C7, C8
10 nF/50 V/SMD/0402	C11, C12
22 nF/16 V/SMD/0402	C34
33 nF/16 V/SMD/0402	C13, C14
47 nF/16 V/SMD/0402	C10
100 nF/16 V/SMD/0402	C1–C6, C9, C17, C19, C21, C22, C24, C25, C28, C29, C31, C32, C36
1 μF/16 V/SMD/0402	C26, C27, C35

10 μF/10 V	C15, C16, C33
10 μF/16 V/SMD/0805	C18, C23
22 μF/16 V/SMD/1206	C30

Halbleiter:

S1206B33U3T1/SOT89-3	IC2
LM4906MM/SMD	IC3
VS1011e	IC4
ELV161534/SMD	IC5
IRLML2502PbF/SMD	T1–T4
IRLML6401/SMD	T5–T7
MMSZ5235B/SOD-123	D7
LED/blau/SMD	D1
LED/rot/SMD/0805/super hell	D2
Quarz/25.000 MHz/SMD	Q2

Sonstiges:

microSD-Kartenhalter	CR1
Stifteleiste, 2x 13-polig, gerade, print	ST1
Stifteleiste, 1x 2-polig, gerade, print	ST2
Stifteleiste, 2x 5-polig, gerade, print	ST3
Cinch-Anschlussplatte, 2-polig, liegend, winkelprint	BU1



Nachhaltig und effizient

Energieeffizienz als Beruf(ung)



*Die Windenergiebranche ist nur eine von vielen, die sich mit dem Thema Energieeffizienz befasst.
© panthermedia.net/WDGPhoto*

Energieeffizienz – was ist das?

Mehr aus Energie machen. Das ist, kurz gesagt, das Ziel der Energieeffizienz. Gemeint ist damit eine rationellere Verwendung von Energie – in einer Zeit weltweit steigenden Bedarfs ein zukunftsträchtiges Arbeitsfeld. Optimierte Prozesse sollen dabei helfen, den energetischen Nutzen zu erhöhen und zugleich den Energieeinsatz zu minimieren.

An der Hochschule Emden/Leer ist das Thema Energieeffizienz bereits seit Jahren Bestandteil von Lehre und Forschung. Und nicht nur das: Am Fachbereich Technik wird auch der Bachelor-Studiengang „Energieeffizienz“ angeboten. Dieser bündelt verschiedene ingenieurwissenschaftliche Kompetenzen und bereitet auf ein stetig wachsendes Berufsfeld vor.

Ein zukunftsweisendes Studium

„Wie wir die Energieversorgung der Zukunft gestalten wollen, ist eine sehr wichtige Frage, sowohl regional als auch global“, sagt Prof. Dr. Iván Herráez. Er koordiniert den Studiengang Energieeffizienz an der Hochschule Emden/Leer seit Übernahme seiner Professur im September 2016. Herráez hat selbst einige Jahre als Entwicklungsingenieur

bei einem Windenergieanlagenhersteller in Hamburg und am Zentrum für Windenergieforschung ForWind in Oldenburg gearbeitet. Seine Schwerpunkte lagen unter anderem im Bereich der Strukturdynamik und Aerolastik. Von seinem Wissen aus der Tätigkeit in Industrie und Forschung profitieren nun seine Studierenden.

Diese sollen als angehende Ingenieure befähigt werden, bereits bestehende Umwandlungsprozesse der konventionellen und regenerativen Energiequellen in Nutzenergie zu verbessern und neue Prozesse zu entwickeln. Damit möchte die Hochschule zum einen den Einstieg in einen zukunftsweisenden Beruf ermöglichen, zum anderen sollen so das Gelingen der Energiewende und eine nachhaltige globale Entwicklung unterstützt werden. Attraktiv ist der Studiengang offenkundig für beide Geschlechter: „Wir haben einen Frauenanteil von rund 40 Prozent“, so Herráez.



Prof. Dr. Iván Herráez



Ein breites Spektrum

Der Studiengang Energieeffizienz ist ein grundständiger Bachelor-Studiengang und als siebensemestriges Vollzeitstudium angelegt. Angesiedelt ist er in der Abteilung „Naturwissenschaftliche Technik“ des Fachbereichs Technik, wird aber gemeinsam mit der Lehrinheit „Maschinenbau“ umgesetzt.

Das Spektrum an Kompetenzen und Befähigungen, das das Energieeffizienz-Studium umfasst, ist vielschichtig. Wissenschaftliche Befähigungen erlangen die Studierenden vor allem in Modulen des vierten bis sechsten Semesters. Dies erfolgt beispielsweise in Projekten und wissenschaftlicher Dokumentation, bei der Diskussion der Ergebnisse, in den vertiefenden Wahlpflichtfächern und während der Erstellung der Bachelorarbeit.

Basiskompetenzen werden im Bereich der Mathematik, in den naturwissenschaftlichen Fächern, in Ingenieurwissenschaften und der Prozesstechnik sowie als Grundwissen im Bereich der konstruktiv nutzbaren Werkstoffe, der Anwendungsbereiche der technischen Mechanik und der Datenverarbeitung vermittelt. Zu den technologischen Kompetenzen, die während des Studiums erworben werden, gehören unter anderem

- Verständnis für physikalisch-chemische Zusammenhänge oder
- verfahrenstechnische Zusammenhänge oder Prozesse der Energieumwandlung.

Als vertieftes Grundwissen kann beispielsweise zwischen

- Anlagentechnik,
 - Konstruktion oder
 - Produktion und Fertigung
- gewählt werden. Zu den fachübergreifenden Kompetenzen zählen zudem
- Grundkenntnisse in Betriebswirtschaftslehre und Recht,
 - das Halten von Präsentationen oder
 - Grundkenntnisse in der Planung und Umsetzung von Projekten.

Die Praxisphase im letzten Semester wird in der Regel in der Industrie oder einem Forschungsinstitut absolviert. Dies kann auch im Ausland geschehen. Neben der Begleitung durch Betreuer vor Ort ist auch eine intensive Betreuung durch Lehrende der Hochschule gewährleistet.

Glänzende Berufsaussichten

„Der Studiengang bietet wirklich glänzende Berufsaussichten“, sagt Prof. Dr. Iván Herráez. Den Absolventen des Studiengangs Energieeffizienz steht ein breites Berufsfeld in der Industrie und der Energiewirtschaft offen. Dazu gehören Tätigkeiten in der Forschung und Entwicklung neuer Energieträger und neuer Umwandlungsprozesse, die Optimierung bestehender industrieller Prozesse sowie beratende Funktionen bei energieerzeugenden Unternehmen.

Potenzielle Arbeitgeber sind hier beispielsweise die energieerzeugenden Unternehmen, Unternehmen der chemischen Industrie, des Maschinen- und Anlagenbaus, die verarbeitende Industrie, Ingenieurbüros, Forschungseinrichtungen sowie der öffentliche Dienst. Die Aufgaben der Absolventen umfassen die Planung, Forschung und Entwicklung und Überwachung von Prozessen und Maschinen zur Gewinnung, Nutzung und Umwandlung primärer Energieträger sowie die Beratung der Energiewirtschaft

„Etwas Sinnvolles machen“

Henrike Wahrmond studiert im sechsten Semester Energieeffizienz. „Ich habe nach einem vielseitigen technischen Studium gesucht, wollte dabei



Studierende im Hörsaal. Foto: Michael Stephan



aber auch etwas Sinnvolles machen“, erzählt die 23-Jährige. Über die Homepage der Hochschule Emden/Leer stieß sie auf den Studiengang Energieeffizienz. „Der Titel und die Beschreibung haben mich sofort angesprochen, da für mich die regenerative Erzeugung und effiziente Nutzung von Energie unabdingbar für einen nachhaltigen Umgang mit dem Thema Energie sind“, sagt sie.

Besonders gefällt Wahrmund, dass der Studiengang interdisziplinär ist und Studierende so einen umfassenden Einblick erhalten. Für eine Spezialisierung im späteren Berufsleben hat sich die 23-Jährige noch nicht entschieden. „Ich würde gerne einen Beitrag zur Energiewende leisten, und wenn es nur ein kleiner ist“, erklärt sie. In ihrer Praxisphase will sie in die Windenergiebranche reinschnuppern, da Windkraft für sie ein sehr spannendes Feld darstellt, wie sie sagt. Auch die Entwicklung innovativer Energiekonzepte oder die Beschäftigung mit Energiespeichern seien interessante Themenfelder. „Denn ohne die wird der Umstieg auf erneuerbare Energien nicht funktionieren.“

Die Inhalte und Qualifikationsziele des Studiengangs sind unter <http://energieeffizienz.hs-emden-leer.de> zu finden.

Interessant auch: „Regenerative Energien“ online, mehr unter: re.hs-emden-leer.de

Smarte Haushalte und Fabriken

Mit der Jahrtausend- und später der von der Bundesregierung beschlossenen Energiewende wurde nicht nur den Ausbau der regenerativen Energielieferanten angeschoben. Auch die Idee, Produktionsverfahren umwelt- und ressourcenschonender zu gestalten – und dabei zugleich wirtschaftlicher zu machen –, hat die Forschung und Entwicklung in diesem Bereich weiter vorangetrieben. Energieeffizienz kann so die Wettbewerbsfähigkeit von Unternehmen steigern, Verbraucher von höheren Energiekosten entlasten und die Umwelt schonen.

Ob Privathaushalt oder Fabrik: Beides soll „smarter“ werden, also durch den Einsatz hochmoderner und intelligenter Technologie Energie einsparen. So lassen sich in Wohnhäusern Geräte wie Heizung, Lampen oder Waschmaschine per Smartphone, Tablet oder PC steuern und kontrollieren. Und auch die Vernetzung von Alltagsgegenständen und die Automatisierung im häuslichen Umfeld ist inzwischen weit vorangeschritten. Das neue Ziel: Energieeffizient, aber zugleich auch nachhaltig werden. Waren es anfangs noch hauptsächlich Heizungsanlagen und Jalousien, die beispielsweise im Ferienhaus aus der Ferne steuerbar sein sollten, so entwickelte sich der Trend auch in Richtung Kontrolle von Alarmanlagen und Küchengeräten – Stichwort Sicherheit. Und auch die „smarte“ Fabrik gehörte zu den neuen Forschungstrends.

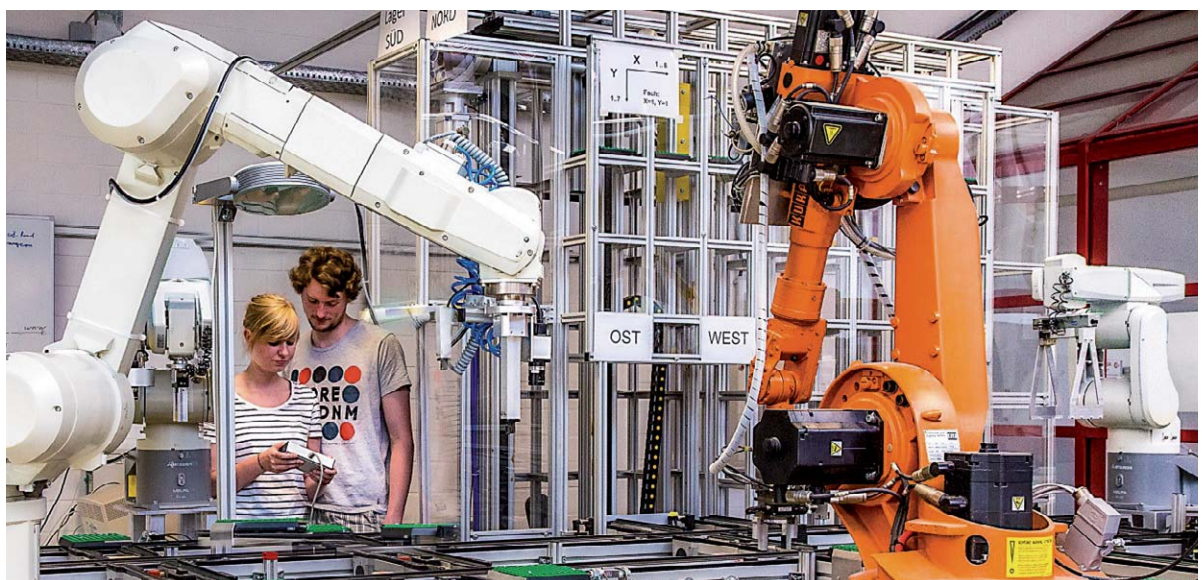
Henrike Wahrmund



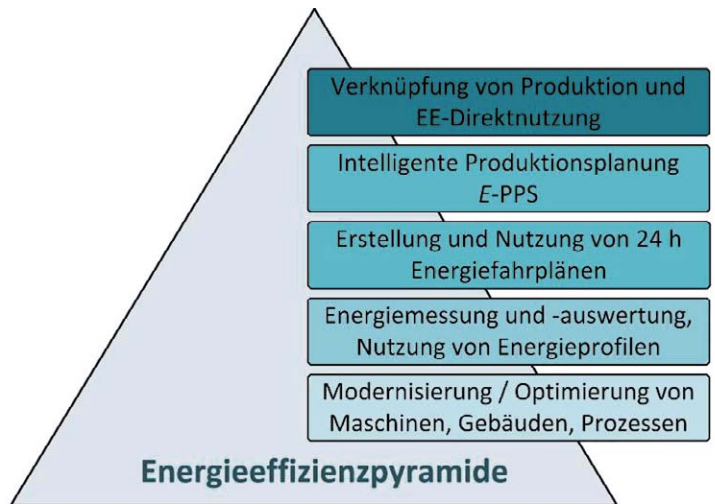
Von der Produktionsplanung zur „grünen“ Fabrik

An der Hochschule Emden/Leer gibt es verschiedene Forschungsprojekte, die sich dem Thema „Energieeffizienz in der Produktion“ widmen. Als Prof. Dr. Ing. Agnes Pechmann im Jahr 2008 an die Hochschule kam, war ihr dieses Thema ein besonderes Anliegen. „Ich habe damals die Professur für Produktionsplanung und -steuerung angenommen und hatte von Anfang an den Wunsch, Energiethemen mit einzubringen“, sagt Pechmann.

Aus dieser Motivation heraus entstand in Zusammenarbeit mit dem deutschen IT-Unternehmen Transfact das Projekt E-PPS (Energieeffiziente Weiterentwicklung der Produktionsplanung und -steuerung). Dieses beinhaltete die Entwicklung eines Systems, das die Produktion auch in Bezug auf Energieeffizienz optimiert. Ziel war es, durch den Einsatz den Energieverbrauch in der Produktion zu senken und Spitzenlasten bei der Leistungsanspruchnahme zu vermeiden. „Spitzenlasten machen einen großen Teil der Energiekosten eines Unternehmens aus“, weiß Pechmann. Bei der Produktionsplanung sollte zudem der Einsatz erneuerbarer Energien berücksichtigt werden.



Ob ganze Haushalte oder einzelne Maschinen – alles wird „smarter“



Ganzheitlicher Ansatz zur Steigerung der Energieeffizienz in der Produktion

Beteiligt waren neben Pechmann als Projektleiterin wissenschaftliche Mitarbeiter und Studierende der Hochschule. „Einige der Studierenden haben inzwischen eine Masterarbeit in diesem Bereich verfasst und arbeiten in der Produktionsplanung“, berichtet Pechmann.

Basierend auf den Erkenntnissen des E-PPS-Projekts wurde dann die Arbeit an zwei weiteren Projekten aufgenommen: Bei PREmdeK (Prognose und Realisierung der Energiebereitstellung mittels dezentraler Kraftwerke) wurde untersucht, wie kleine und mittelgroße Produktionsunternehmen CO₂-neutral und gleichzeitig wirtschaftlich fertigen können. „Wir konnten damit anhand realer Daten aufzeigen, wie die ‚grüne Fabrik‘ konkret für einzelne Unternehmen möglich sein kann“, erklärt Pechmann.

Ein dezentrales Kraftwerk ist eine Zusammenfassung von mehreren regenerativen Energieerzeugern wie Windkraft- oder Photovoltaikanlagen. Diese sollen den Energiebedarf eines Unternehmens vollständig oder teilweise abdecken. Die Produktion und das Kraftwerk bilden dabei ein lokales Energiesystem. Die Ist-Situation wurde durch Energiemessungen erfasst und anschließend analysiert. Anhand der Analyseergebnisse wurde dann die Produktion energetisch optimiert. Die „großen“ Verbraucher wurden dabei durch eine speziell an der Hochschule Emden/Leer erarbeitete Vorgehensweise identifiziert. Kernstück des For-

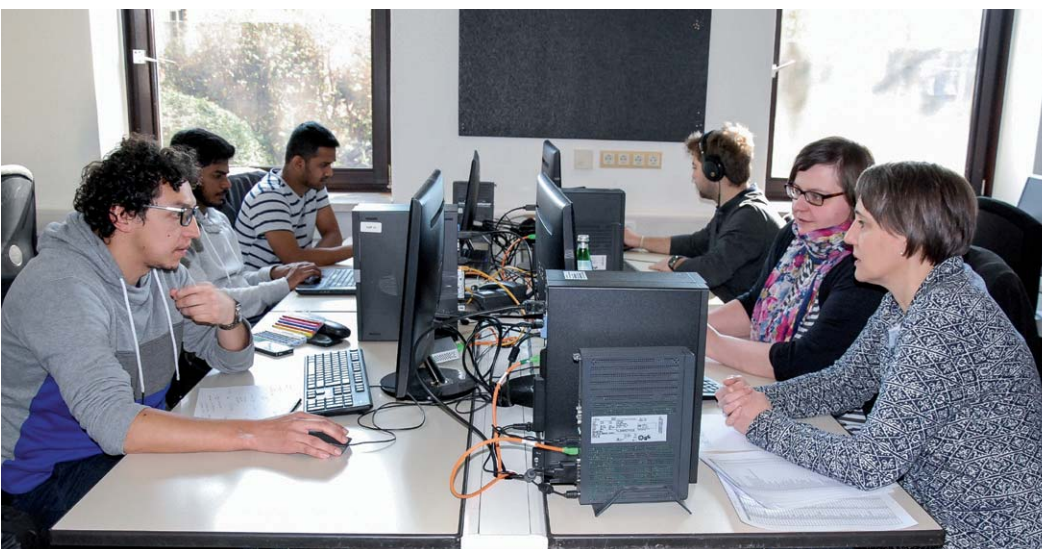
schungsprojektes war die Entwicklung einer Simulationssoftware, die basierend auf Wetterprognosen den Ausgleich zwischen dem Energieverbrauch eines Unternehmens und dem Bezug erneuerbarer Energie bzw. Energie vom Netz vorhersagen kann.

Das Projekt REN ProV (Regenerative Energie in der Produktion – Organisatorische Voraussetzungen und Implementierung) wiederum zielte darauf ab, den Anteil erneuerbarer Energien bei der Stromversorgung eines Unternehmens zu erhöhen. Zudem sollte die technische Leistungsgrenze durch Nutzung selbst generierter, erneuerbarer Energien erhöht werden. Regenerativ erzeugte Energie ist jedoch wetterabhängig, und die Speichermöglichkeiten bisher vor allem für kleine und mittlere Unternehmen eine recht kostspielige Investition.

Das Team des Projektes REN ProV entwickelte eine Produktionsplanungs- und -steuerungssoftware, die – mittels eines Algorithmus zur Kapazitätsplanung und unter einer Limitierung der vorhandenen Energiemenge – eine bestmögliche Produktion hinsichtlich Menge und Lieferfähigkeit von Energie sicherstellt. Die entwickelte Software bietet dementsprechend unterschiedliche Möglichkeiten: Die Ausweitung der Produktionsfenster – beispielweise bei Stromausfällen –, die Erhöhung der technischen Leistungsgrenze durch Nutzung selbst generierter, erneuerbarer Energien und eine höchstmögliche Produktionsmenge bei kompletter netzunabhängiger Nutzung von erneuerbarer Energie.

Als „Testfabrik“ für die Komplettversorgung aus erneuerbaren Energien diente ein Unternehmen aus der Region. Das Ergebnis: „Es ist technisch möglich, ein Unternehmen vollkommen mit erneuerbaren Energien zu versorgen. Ob das Ganze auch wirtschaftlich sinnvoll ist, muss natürlich ebenfalls abgewogen werden“, sagt Pechmann.

Genau mit dieser Frage beschäftigen sich Pechmann und ihr Team an der Hochschule derzeit im Projekt WeEmU (Wirtschaftlichkeit von erneuerbaren Energieerzeugern für mittelständische Unternehmen). Dabei soll das Verhältnis zwischen dem Eigennutzen der „grün“ produzierten Energie und dem Anteil an extern bezogener Energie optimiert werden.



Prof. Dr. Agnes Pechmann mit Studierenden bei einer Laborübung



Dazu wurden die aktuellen geltenden Rahmenbedingungen durch das „Erneuerbare Energie Gesetz“ (EEG 2017) analysiert und Geschäftsmodelle zur Betreibung eines erneuerbaren Energie Kraftwerkes von kleinen und mittleren Unternehmen definiert. Für Unternehmen, die in erneuerbare Energien investieren wollen, wurde nun mit der Simulationssoftware aus dem Projekt PREmdeK und historischen Daten für den Energieverbrauch des Unternehmens geprüft, welches Geschäftsmodell bei welcher Auslegung der erneuerbaren Energieerzeuger wirtschaftlich für das Unternehmen am sinnvollsten ist.

Generell ist die die Zusammenarbeit mit Unternehmen für Prof. Dr. Ing. Agnes Pechmann ein wichtiger Ansatz, den sie in ihren Forschungsprojekten konsequent verfolgt, um das gewonnene Wissen anzuwenden. Die Ergebnisse der Projekte wurden direkt bei gewerblichen Projektpartnern umgesetzt, so beispielsweise bei den Unternehmen Maschinenbau Grimm GmbH in Apen, den Leda-Werken in Leer oder beim alteingesessenen Traditionsunternehmen Schüt-Duis und Auricher Süssmost in Aurich. Diese Zusammenarbeit mit regionalen Unternehmen spiegelt sich auch in zahlreichen studentischen Abschluss- und Projektarbeiten wider.

Dachmarke für die Region: greentech OSTFRIESLAND

„Wir möchten uns in Zukunft noch stärker im Bereich ‚Grüne Technologien und gesellschaftliche Verantwortung‘ profilieren“, erklärt Hochschulpräsident Prof. Dr. Gerhard Kreutz. Die Hochschule stelle schon jetzt einen besonderen Bezug zum Thema Nachhaltigkeit her. Dies biete sich zum einen aufgrund der vielfachen Nutzung regenerativer Energien in der Region und der Nähe zum Weltnaturerbe Wattenmeer an, spiegele sich aber auch in den Lehrinhalten, Projekten und Partnerschaften des Hauses wider.



Prof. Dr. Gerhard Kreutz

Im Jahr 2014 hat die Hochschule gemeinsam mit der Industrie- und Handelskammer für Papenburg (IHK), der Stadt Emden mit der Zukunft Emden GmbH und der Handwerkskammer für Ostfriesland (HWK) die Netzwerkinitiative greentech OSTFRIESLAND gegründet. Ziel aller Beteiligten ist es, das breite Spektrum grüner Technologien in der Region zu bündeln, eine gemeinsame Dachmarke zu entwickeln und so mittel- bis langfristig neue Arbeitsplätze in der Region entstehen zu lassen. Konkret soll greentech OSTFRIESLAND als Plattform dienen, um innovative Ideen und Produkte zu entwickeln und entsprechende Konzepte umzusetzen. Gerade kleineren und mittelständischen Betrieben bietet sich damit die Chance, zukunftsfähige Nischen zu besetzen.

Unabdingbare Voraussetzung für diese Ziel, so Kreutz, sei die Entwicklung fachlich passender Studienangebote durch die Hochschule, die damit verbundene Ausbildung sehr gut qualifizierter potenzieller Mitarbeiter und somit der Aufbau entsprechenden Know-hows in der Region. Schon jetzt gibt es viele innovative, „grüne Projekte“ innerhalb und außerhalb der Hochschule. Auf dem Gebiet angewandter Forschung stellen, wie bereits erwähnt, besonders regionale Fragestellungen der Energie- und Umwelttechnik ein stark beachtetes Themenspektrum dar.

Nachdem greentech OSTFRIESLAND vor gut zwei Jahren mit einer erfolgreichen Auftaktveranstaltung an den Start ging, kommen „grüne“ Unternehmer – und solche, die es werden wollen – mittlerweile regelmäßig zusammen, um beim „Regionalforum greentech OSTFRIESLAND“ Kontakte zu knüpfen, neue Technologien kennenzulernen und sich auszutauschen. Bei den halbjährlichen Zusammenkünften wird stets ein anderer Themenschwerpunkt wie etwa Ressourceneffizienz oder nachhaltige Wasserwirtschaft in den Mittelpunkt gerückt. Dr. Stephan Kotzur, Leiter des Hochschul-Instituts Logistik (HILOG), sieht in der



Dr. Stephan Kotzur

Bündelung der grünen Aktivitäten eine große Chance. „Ostfriesland ist weit über Niedersachsen hinaus bekannt als Region der regenerativen Energien“, sagt er. Dass es jedoch weit mehr als nur viele Windparks oder Solaranlagen gebe, wolle die Initiative gezielt bekannter machen. „Die Region hat weit mehr zu bieten und somit beste Voraussetzungen, sich zu einer Modellregion für Nachhaltigkeit in Wirtschaft und Wissenschaft zu entwickeln.“

Unter einer Dachmarke sollen vielversprechende Ansätze zusammengeführt, von allen gemeinsam getragen und vorangebracht werden. Dank der Institutionalisierung der Zusammenarbeit soll ein gemeinsames Fundament geschaffen werden, das Impulse zur Wirtschaftsförderung gibt. Niedersachsenweit leisten die Initiatoren damit Pionierarbeit: Bislang gibt es keine vergleichbaren Ansätze.



Nachhaltigkeit an der Hochschule

Nachhaltigkeit und Energieeffizienz haben viele Gesichter. Die Hochschule Emden/Leer hat in ihrem Leitbild verankert, eine nachhaltige Entwicklung zu fördern und damit auch zugleich Impulsgeberin für die Region zu sein. Dies wird in Lehre und Forschung, aber auch in der Umsetzung eines betrieblichen Gesundheitsmanagements, einer familienfreundlichen Betriebsphilosophie oder der Auszeichnung als Fairtrade-Hochschule deutlich.

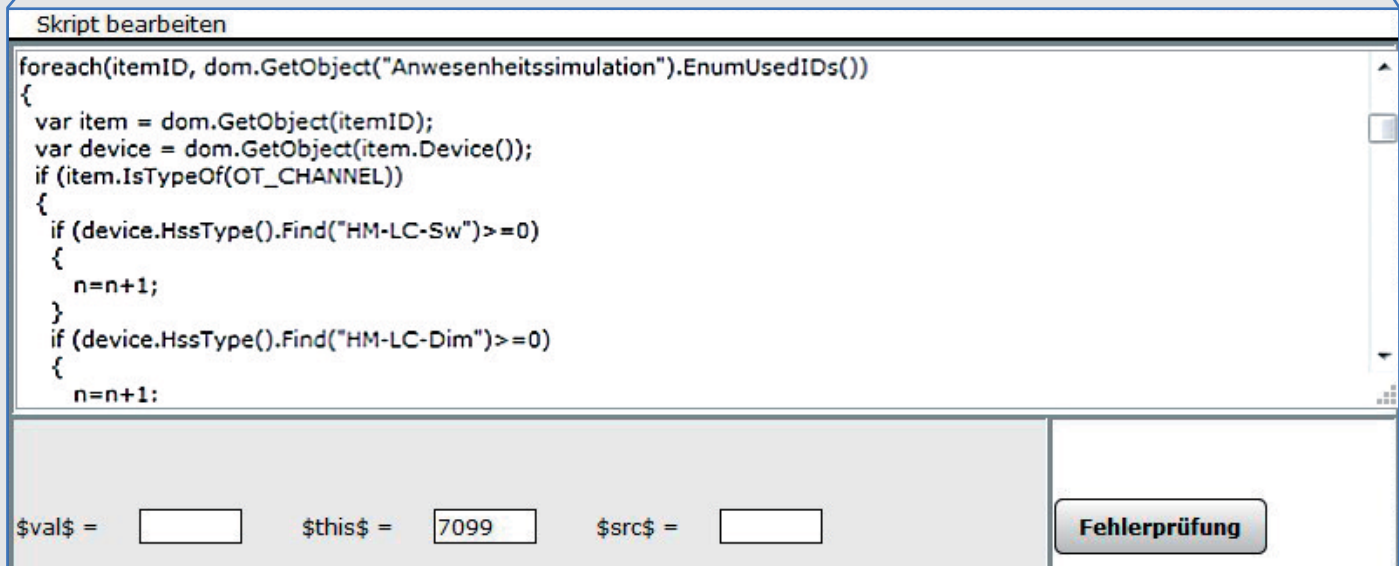
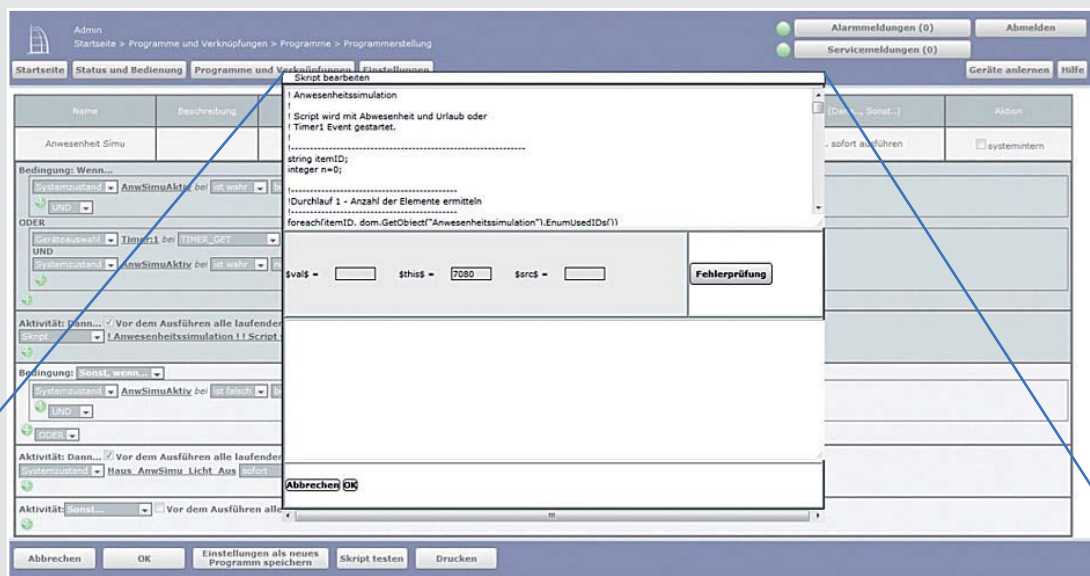
Zudem wird die Hochschule seit Dezember 2016 zum Teil über ein Blockheizkraftwerk (BHKW) mit Wärme und Strom versorgt. „Auch dies unterstreicht unser Bestreben, stets nachhaltig zu agieren“, sagt Kreutz. Lehrinhalte, Partnerschaften und Projekte auf „grüner“ Basis haben sich mit der Zeit entwickelt und werden stetig ausgebaut.

Fazit: Energieeffizienz ist ein typisches Querschnittsthema und bietet hoch interessante und äußerst vielfältige Möglichkeiten. Ein Studium in diesem Themenfeld ist genau das Richtige für Menschen, die Verantwortung übernehmen wollen und Energieeffizienz als Beruf(ung) sehen.



Homematic Scriptprogrammierung

Teil 4 – Das CCU-Add-on CUxD und Scripte



Im vierten Teil der Artikelserie über die Scriptprogrammierung beschäftigen wir uns mit dem CCU-Add-on CUxD und schreiben weitere Scripte.



Das CCU-Add-on CUxD

Für die Homematic CCU gibt es mittlerweile eine ganze Reihe von sogenannten Add-ons, also von Softwarepaketen, die auf der Homematic Zentrale (CCU) installiert werden können und neue Funktionalitäten zur Verfügung stellen, die nicht Teil der offiziellen Firmware sind.

Der CUxD-Daemon (CUxD) ist eine universelle Schnittstelle zwischen der Homematic Zentrale (CCU) und entsprechend kompatiblen Komponenten von EnOcean, FS20, ELV-Wetterstationen und vielen mehr. Durch Einbindung dieser (eigentlich inkompatiblen) Produkte lässt sich der Umfang eines Homematic Systems also über die Grenzen hinweg erweitern.

Weiterhin bietet CUxD Funktionen, die nicht im Umfang der Homematic Scriptprogrammierung vorhanden sind.

Die Einrichtung und Verwendung des Add-ons ist in der dazugehörigen Dokumentation sehr gut beschrieben. Im Folgenden ein paar Beispiele mit Funktionen aus dem CUxD-Paket :

SYSTEM_EXEC

Die wohl bekannteste Anwendung von CUxD-Möglichkeiten ist der Ersatz des undokumentierten und nicht unproblematischen SYSTEM_EXEC-Script-Befehles durch den korrespondierenden CUxD-Befehl.

Hier ein Beispiel für den Versand einer E-Mail mithilfe des E-Mail-Add-ons und des CUxD-System-Exec-Befehls:

Nicht verwenden:

1	<code>string stderr;</code>
2	<code>string stdout;</code>
3	<code>system.Exec("/etc/config/addons/email/email 03 '#MailText#'",&stdout,&stderr);</code>

Stattdessen:

1	<code>dom.GetObject("CUxD.CUX2800001:10.CMD_EXEC").State("/etc/config/addons/email/email 03 '#MailText#'");</code>
---	--

CUxD.CUX2800001:10 muss natürlich entsprechend den eigenen Gegebenheiten abgeändert werden!

Zufallszahlen und Timer

Im Zusammenhang mit einem weiteren Script, das wir vorstellen möchten, wird die Generierung von Zufallszahlen mithilfe des CUxD-Add-ons gezeigt.

Bei dem Script handelt es sich um eine Anwesenheitssimulation, die zufällig gewählte Lampen im Haus für zufällig gewählte Zeiten ein- und dann wieder ausschaltet.

Wir verwenden dazu folgende CUxD-Befehlsstruktur:

<code>dom.GetObject("CUxD.CUX2801001:2.RAND").State(xx);</code>
<code>integer RNDTime=dom.GetObject("CUxD.CUX2801001:2.RAND").State().ToInteger();</code>

CUxD.CUX2801001:2 ist der Name und der Kanal des CUxD-Elementes, die Einrichtung ist in der CUxD-Dokumentation sehr umfangreich beschrieben.

Die erste Programmzeile wählt den Bereich an, aus dem die Zufallszahl stammt, der Bereich ist:

$0 \leq \text{Zufallszahl} \leq \text{xx}$

D. h., bei $\text{xx} = 90$ wird mit der nächsten Zeile in die im Script deklarierte Variable RNDTime eine zufällige Zahl aus dem Bereich $0 \dots 90$ gespeichert.



Voraussetzung für den Betrieb des Scriptes ist nun lediglich noch das Anlegen eines Gewerkes „Anwesenheitssimulation“, dem alle Lampen zugeordnet werden, die innerhalb der Simulation ein- und ausgeschaltet werden sollen. Das folgende Zentralenprogramm startet dann das Script:

Name	Beschreibung	Bedingung (Wenn...)
Anwesenheit Simu		Systemzustand: AnwSimuAktiv bei bei Änderung auslösen ist wahr
Bedingung: Wenn...		
Systemzustand AnwSimuAktiv bei ist wahr bei Änderung auslösen		
UND		
ODER		
Geräteauswahl Timer:1 bei TIMER_GET im Wertebereich kleiner oder gleich 0.00 s bei Änderung auslösen		
UND		
Systemzustand AnwSimuAktiv bei ist wahr nur prüfen		
Aktivität: Dann... Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).		
Script !Anwesenheitssimulation !! Script wird mit Abwesenheit und... sofort		
Bedingung: Sonst, wenn...		
Systemzustand AnwSimuAktiv bei ist falsch bei Änderung auslösen		
UND		
ODER		
Aktivität: Dann... Vor dem Ausführen alle laufenden Verzögerungen für diese Aktivitäten beenden (z.B. Retriggern).		
Script !alle Lampen Gewerk AnwSimu ausschalten var i = dom.GetObject... sofort		

Gestartet wird die Anwesenheitssimulation in Form eines ersten Scriptes durch Setzen der Systemvariablen (CCU-Variablen) *AnwSimuAktiv* auf *true*.

Zum Ausschalten der Anwesenheitssimulation wird ein zweites Script, das eventuell noch leuchtende Lampen ausschaltet, mit dem Zustand *false* der Systemvariablen *AnwSimuAktiv* gestartet.

Anmerkung: Vor allem das erste Script lässt sich natürlich vereinfachen, mit der gezeigten Struktur lassen sich aber hervorragend (weitere) Versuche machen.





Das erste Script:

```

1  ! Anwesenheitssimulation
2  !
3  ! Script wird z.B. mit Abwesenheit oder Urlaub gestartet.
4  ! Das Script muss zur ordnungsgemaessen Funktion auch mit dem Timer1 Event gestartet
5  ! werden.
6  !-----
7  string itemID;
8  integer n=0;
9  !-----
10 !Durchlauf 1 - Anzahl der dem Gewerk Anwesenheitssimulation zugeh. Elemente ermitteln
11 !-----
12 foreach(itemID, dom.GetObject("Anwesenheitssimulation").EnumUsedIDs())
13 {
14     var item = dom.GetObject(itemID);
15     var device = dom.GetObject(item.Device());
16     if (item.IsTypeOf(OT_CHANNEL))
17     {
18         if (device.HssType().Find("HM-LC-Sw")>=0)
19         {
20             n=n+1;
21         }
22         if (device.HssType().Find("HM-LC-Dim")>=0)
23         {
24             n=n+1;
25         }
26     }
27 }
28 ! kein Element fuer Anwesenheitssimulation gefunden
29 if (n==0) {
30     quit;
31 }
32 ! Eine Zufallszahl (RND_Nr) im Bereich von 0 bis zur gefundenen Anzahl von Lampen im
33   !Gewerk (n) ermitteln
34 dom.GetObject("CUxD.CUX2801001:1.RAND").State(n);
35 integer RND_Nr = dom.GetObject("CUxD.CUX2801001:1.RAND").State().ToInteger();
36 !fuer Durchlauf 2 n=-1 setzen
37 n=-1;
38 !-----
39 !Durchlauf 2 - Zufalls-Element schalten
40 !-----
41 foreach(itemID, dom.GetObject("Anwesenheitssimulation").EnumUsedIDs())
42 {
43     var item = dom.GetObject(itemID);
44     if (item.IsTypeOf(OT_CHANNEL))
45     {
46         var device = dom.GetObject(item.Device());
47         !-----
48         !Einschalten bzw. auf 100 Prozent dimmen
49         !-----
50         if (device.HssType().Find("HM-LC-Sw")>=0)
51         {
52             n=n+1;
53             if (n == RND_Nr) {
54                 !-----
55                 !Einschaltzeit (RNDTime) innerhalb von 0...90 Minuten Zufallszahl
56                 !-----
57                 dom.GetObject("CUxD.CUX2801001:2.RAND").State(90);
58                 integer RNDTime=dom.GetObject("CUxD.CUX2801001:2.RAND").State().ToInteger();
59                 if (RNDTime==0) {RNDTime=1;}
60                 !-----
61                 !Einschaltzeit setzen
62                 !-----
63                 item.DPByHssDP("ON_TIME").State(60*RNDTime);
64
65                 !-----
66                 !Einschalten
67                 !-----
68                 item.State(1);
69             }
70         }
71     }
72     if (device.HssType().Find("HM-LC-Dim")>=0)
73     {
74         n=n+1;
75         if (n == RND_Nr) {

```



76	!-----
77	!Einschaltzeit innerhalb von 0...90 Minuten Zufallszahl
78	!-----
79	dom.GetObject("CUxD.CUX2801001:2.RAND").State(90);
80	integer RNDTime=dom.GetObject("CUxD.CUX2801001:2.RAND").State().ToInteger();
81	if (RNDTime==0) {RNDTime=1;}
82	!-----
83	!Einschaltzeit setzen
84	!-----
85	item.DPByHssDP("ON_TIME").State(60*RNDTime);
86	!-----
87	!Einschalten
88	!-----
89	item.State(1.00);
90	}
91	}
92	}
93	}
94	!-----
95	! CUxD Timer setzen
96	!-----
97	dom.GetObject("CUxD.CUX2800001:1.TIMER_SET").State(60*RNDTime);

CUxD.CUX2800001:1 bzw. CUxD.CUX2801001:2 sind natürlich auf die eigenen Gegebenheiten anzupassen.

Funktion des Skriptes:

Zunächst werden in einer Schleife (Zeilen 12 bis 27) alle Aktoren gezählt, die dem Gewerk „Anwesenheitssimulation“ angehören. Diese Funktionalität lässt sich natürlich auch mit der Funktion .Count() erreichen, dann dürfen aber wirklich nur Relais und Dimmer zum Gewerk „Anwesenheitssimulation“ gehören.

Nun wird aus diesen Aktoren derjenige bestimmt, der eingeschaltet werden soll. Dies geschieht durch Wahl einer Zufallszahl zwischen Null und der Anzahl, die vorher ermittelt wurde (Zeilen 33 bis 35).

Angenommen, es werden 6 Geräte gefunden, die zum Gewerk „Anwesenheitssimulation“ gehören, dann hat n in der Zeile 33 den Wert 6. Somit gilt:

Mit der Zeile

```
dom.GetObject("CUxD.CUX2801001:1.RAND").State(6);
```

wird bestimmt, dass die Zufallszahl im Bereich von 0 bis 6 liegen soll.

Mit der Zeile

```
integer RND_Nr = dom.GetObject("CUxD.CUX2801001:1.RAND").State().ToInteger();
```

steht die Zufallszahl dann in der Variablen *RND_Nr*.

In einem zweiten Durchlauf (Zeile 40 bis 93) wird dann der per Zufallszahl ermittelte Aktor (Relais oder Dimmer) eingeschaltet. Dies geschieht für eine Zeit im Bereich von 0 bis 90 Minuten, die ebenfalls per Zufallszahl ermittelt wird (Zeile 56 bis 58 bzw. 79 bis 81).

Sollte als Zufallszahl die 0 ermittelt werden, wird stattdessen der Wert 1 genommen.

Abschließend wird dann ein CUxD-Timer auf die zufällig ermittelte Einschaltzeit gesetzt, damit nach deren Ablauf das Script wieder gestartet wird, um die nächste (zufällig ermittelte) Lampe für eine (ebenfalls zufällig ermittelte) Zeit einzuschalten.

Das zweite Script mit dem Ausschalten der eventuell noch leuchtenden Lampen:

1	!alle Lampen Gewerk Anwesenheitssimulation ausschalten
2	var i = dom.GetObject("Anwesenheitssimulation");
3	string itemID;
4	foreach(itemID, i.EnumUsedIDs())
5	{
6	var item = dom.GetObject(itemID);
7	if (item.IsTypeOf(OT_CHANNEL))
8	{
9	var device = dom.GetObject(item.Device());
10	if ((device.HssType().Find("HM-LC-Sw") >= 0))
11	{
12	item.DPByHssDP("ON_TIME").State(0);
13	item.State(0);
14	}
15	if ((device.HssType().Find("HM-LC-Dim") >= 0))
16	{
17	item.DPByHssDP("ON_TIME").State(0);
18	item.State(0.00);
19	}
20	}
21	}

Das CUxD-Add-on bietet über die gezeigten Funktionen hinaus viele weitere mehr. Probieren Sie es aus!

Beispiel für die Ausgabe von http-Befehlen

Mit dem zunehmenden Wachstum des „Internet of Things“ steigt die Anzahl der Geräte, die über http-Befehle gesteuert werden. Für das Absetzen eines http-Befehls wird ebenfalls die oben schon angesprochenen CUxD-Funktion SYSTEM_EXEC verwendet. Hier ein Beispiel für die Ansteuerung eines Mediola Interfaces (Schnittstelle zu Infrarot-Geräten) mit einem Infrarot-Steuerbefehl:

Laut Dokumentation des Interfaces setzt sich der Sendecode wie folgt zusammen:
http://aaa.bbb.ccc.ddd/command?XC_PASS=password&code=xyz&XC_FNC=Send2

Dabei sind:
aaa.bbb.ccc.ddd = IP-Adresse des Interfaces
password = Passwort des Interfaces
xyz = Infrarot Code

Der Infrarot-Code wird zunächst in eine Variable geschrieben (in diesem Fall z. B. der Code für die Taste ON/OFF einer Fernseh-Fernbedienung):

```
!-----
!Code ON OFF
!-----
string CODE = "19082600003002204046b046b008a01a5008a008a008a2afe00010101020202020101010202020
20102020101020201020101020201010203";
```

ebenso die IP-Adresse des Adapters (Beispiel):

```
string IP_ADR = "192.168.2.155"
```

sowie das Passwort (Beispiel):

```
string PW = "Adapterpasswort"
```

Damit sieht der Sendebefehl beispielsweise folgendermaßen aus:

```
!-----
!Sendecode
!-----
string url="http://" # IP_ADR # "/command?XC_PASS=" # PW # "&code=" # CODE # "&XC_FNC=Send2";
```

Die Einzelstrings werden einfach zum Sendecode gekettet.

Das Absetzen des Sendebefehls erfolgt nun durch den CUxD-Befehl:

```
!-----
!Ausfuehren
!-----
dom.GetObject("CUxD.CUX2801001:1.CMD_EXEC").State("wget -q -O - '" # url # "'");
```

Script: Verknüpfte Informationen

In CCU-Systemvariablen (z. B. Wertelisten oder Textvariablen) lassen sich auch sehr gut Listen mit Informationen unterbringen und verarbeiten. Dies ist vor allem dann sinnvoll, wenn Sie dadurch vermeiden, innerhalb von Scripten Informationen (Daten) zu programmieren, die sich irgendwann einmal ändern können. Auch können Programmteile, die Sie in anderen Scripten ebenfalls verwenden möchten, ohne Änderungen übernommen werden.

Wenn Sie z. B. den Funk-MP3-Gong zur Ausgabe von Sprachinformationen verwenden wollen, ergibt beispielsweise die Zuordnung von Raumnamen zu Textnummern der Soundfiles solch eine Liste:

Raum	Nummer des Soundfiles
Wohnzimmer	100
Esszimmer	101
Küche	102
Badezimmer	103
Schlafzimmer	104
Kinderzimmer	105
Flur	106
WC	107
Garten	108
Terrasse	109
Garage	110

Diese Tabelle lässt sich in einer Systemvariablen unterbringen und in einem Script verwenden.



Erstellung der Systemvariablen (Typ Zeichenkette):

Systemvariable bearbeiten					
Name	Beschreibung	Variablentyp	Werte	Maßeinheit	Kanal-zuordnung
Room_Sound_List		Zeichenkette ▾			<input checked="" type="radio"/> ohne <input type="radio"/> mit <input type="button" value="Kanalauswahl"/>
<input type="button" value="Abbrechen"/> <input type="button" value="OK"/>					

In einem (Zentralen-)Programm wird dann die Variable *Room_Sound_List* einmalig mit dem folgenden String beschrieben:

```
"Wohnzimmer*100,Esszimmer*101,Küche*102,Badezimmer*103,Schlafzimmer*104,
Kinderzimmer*105,Flur*106,WC*107,Garten*108,Terrasse*109,Garage*110"
```

Es handelt sich also um eine *Liste* (Kommatrennung), deren Listenelemente jeweils aus dem Namen des Raumes sowie der zugehörigen Textnummer (immer dreistellig, also auch 001, 002 ...), getrennt durch einen Separator (in diesem Fall *) bestehen.

Im folgenden Script ermitteln die Zeilen 9 bis 15 nun aus dem Raumnamen, der in der Variablen *sRaum* steht, und der Zentralenvariablen *Room_Sound_List* die Nummer des zur Küche gehörenden Textes (MP3-File):

1	string daten;
2	string sRaum = "Küche";
3	string sTextnummer;
4	string sCommand;
5	integer word_position;
6	!-----
7	! Textnummer fuer Raum
8	!-----
9	daten = dom.GetObject("Room_Sound_List").State();
10	word_position = daten.Find(sRaum);
11	if (word_position >= -1)
12	{
13	daten = daten.Substr(word_position + 1, daten.Length() - word_position);
14	word_position = daten.Find("*");
15	sTextnummer = daten.Substr(word_position + 1, 3);
16	sCommand = sCommand # ", " # sTextnummer;

Die Textnummer wird als String ermittelt, da sie ja bei der Ansteuerung des MP3-Funk-Gongs auch im Stringformat an diesen übermittelt werden muss.

In der Zeile 16 wird die Nummer in den Kommandostring für den Funk-Gong eingesetzt.

Hier ein komplettes Beispiel für eine akustische Meldung über den Batteriezustand der batteriebetriebenen Geräte in einer Hausinstallation. Neben der Zentralenvariablen *Room_Sound_List* gibt es eine Variable *Ger_Sound_List*, in der analog zu obigem Beispiel die Zuordnungen zwischen Gerätenamen und Soundfilenummern gespeichert sind.

Im folgenden Beispiel wurden die Gerätenamen wie folgt gewählt:

*Raum*Gerätebezeichnung*laufende Nummer dieses Gerätetyps in diesem Raum*

Also zu Beispiel:

*Wohnzimmer*Türschalter*01*

Die laufende Nummer einfach deshalb, weil es mehrere gleiche Geräte in einem Raum geben kann.

Das Script überprüft den Batteriezustand jedes batteriebetriebenen Gerätes und gibt entweder eine O.K.-Meldung oder die Namen der Geräte aus, deren Batterien zur Neige gehen (max. 4 Stück gleichzeitig).

1	! Batteriemeldungen Sprachausgabe
2	string itemID;
3	string text = "";
4	integer counter = 0;
5	string command;
6	string text_nr;
7	string daten;
8	string d_Raum;
9	string d_Geraet;



10	string d_Nummer;
11	integer word_position;
12	!*****
13	! Variablen (Textnummern) hier bitte anpassen
14	!*****
15	!Textnummer von "Die Batterien der folgenden Geräte sind bald leer"
16	string text_batt_leer = "126";
17	!Textnummer von "Alle Batterien sind voll"
18	string text_batt_ok = "213";
19	!maximale Anzahl von Geräten, die angesagt werden
20	integer max_anz_mldg = 4;
21	!Name des Gewerkes BATTERIE
22	string gew_batt = "Batteriebetrieb";
23	var myAssembly = dom.GetObject(gew_batt);
24	command= "1,1,108000" # " " # text_batt_leer;
25	foreach(itemID,myAssembly.EnumUsedIDs())
26	{
27	var item = dom.GetObject(itemID);
28	var device = dom.GetObject(item.Device());
29	var interface = dom.GetObject(item.Interface());
30	string interface_name = interface.Name();
31	string device_address = device.Address();
32	device_address = device_address.StrValueByIndex(":",0);
33	boolean condition=false;
34	if (device.HssType().Find("HM-CC-RT-DN")>=0) {
35	string channel_name=interface_name # "." # device_address #
36	":4.FAULT_REPORTING";
37	var channel = dom.GetObject(channel_name);
38	if (channel.Value() ==6)
39	{
40	condition = true;
41	}
42	else
43	{
44	condition = false;
45	}
46	else
47	{
48	string channel_name = interface_name # "." # device_address # ":0.LOWBAT";
49	var channel = dom.GetObject(channel_name);
50	if ((channel.State() == true) && (channel.Value() == true))
51	{
52	condition = true;
53	}
54	else
55	{
56	condition = false;
57	}
58	}
59	if (counter < (max_anz_mldg-1))
60	{
61	if (condition == true)
62	{
63	!-----
64	! Raum, Geraet und Nummer ausdecodieren
65	!-----
66	word_position = daten.Find("*");
67	if (word_position>0)
68	{
69	d_Raum = daten.Substr(0, word_position);
70	daten = daten.Substr(word_position+1,daten.Length()- word_position-1);
71	word_position = daten.Find("*");
72	if (word_position>-1)
73	{
74	d_Geraet = daten.Substr(0,word_position);
75	daten=daten.Substr(word_position+1,daten.Length()-word_position-1);
76	d_Nummer = daten;
77	}
78	}
79	!-----
80	! UND
81	!-----
82	if (counter>0)
83	{
84	command = command # " " # "020";
85	}
86	!-----
87	! Textnummer fuer Raum



88	!-----
89	daten = dom.GetObject("Room_Sound_List").State();
90	word_position = daten.Find(d_Raum);
91	if (word_position>=-1)
92	{
93	daten=daten.Substr(word_position+1, daten.Length()- word_position);
94	word_position = daten.Find("*");
95	text_nr = daten.Substr(word_position+1,3);
96	command = command # "," # text_nr;
97	}
98	!-----
99	! Textnummer fuer Geraet
100	!-----
101	daten=dom.GetObject("Ger_Sound_List").State();
102	word_position = daten.Find(d_Geraet);
103	if (word_position>=-1)
104	{
105	daten=daten.Substr(word_position+1, daten.Length()- word_position);
106	word_position = daten.Find("*");
107	text_nr = daten.Substr(word_position+1,3);
108	
109	command = command # "," # text_nr;
110	}
111	!-----
112	! Geraetenummer (laufende Nummer)
113	!-----
114	text_nr = d_Nummer;
115	command = command # "," # text_nr;
116	counter = counter + 1;
117	}
118	}
119	}
120	if (counter > 0)
121	{
122	dom.GetObject("FunkGong:2").DPByHssDP("SUBMIT").State(command);
123	}
124	else
125	{
126	command= "1,1,108000" # "," # text_batt_ok;
127	dom.GetObject("FunkGong:2").DPByHssDP("SUBMIT").State(command);
128	}

Im Script wurden noch folgende Texte (Soundfiles) verwendet:

- Die Batterien der folgenden Geräte sind bald leer - Filenummer 126 – Zeile 16
- Alle Batterien sind voll - Filenummer 213 – Zeile 18
- und - Filenummer 020 – Zeile 84

Die Zahlen von 1 bis 9 liegen im Funk-Gong auf den Speicherplätzen 1 bis 9, die MP3-Files haben also die Namen 001.mp3, 002.mp3 bis 009.mp3.


Gestartet werden kann das Script durch einen Taster, über die Zeitsteuerung oder man stellt in einem Programm zyklisch fest, ob ein Batteriefehler vorliegt, setzt dann eine Zentralenvariable und startet damit die Ausgabe. Das Script bietet viel Spielraum für eigene Versuche.

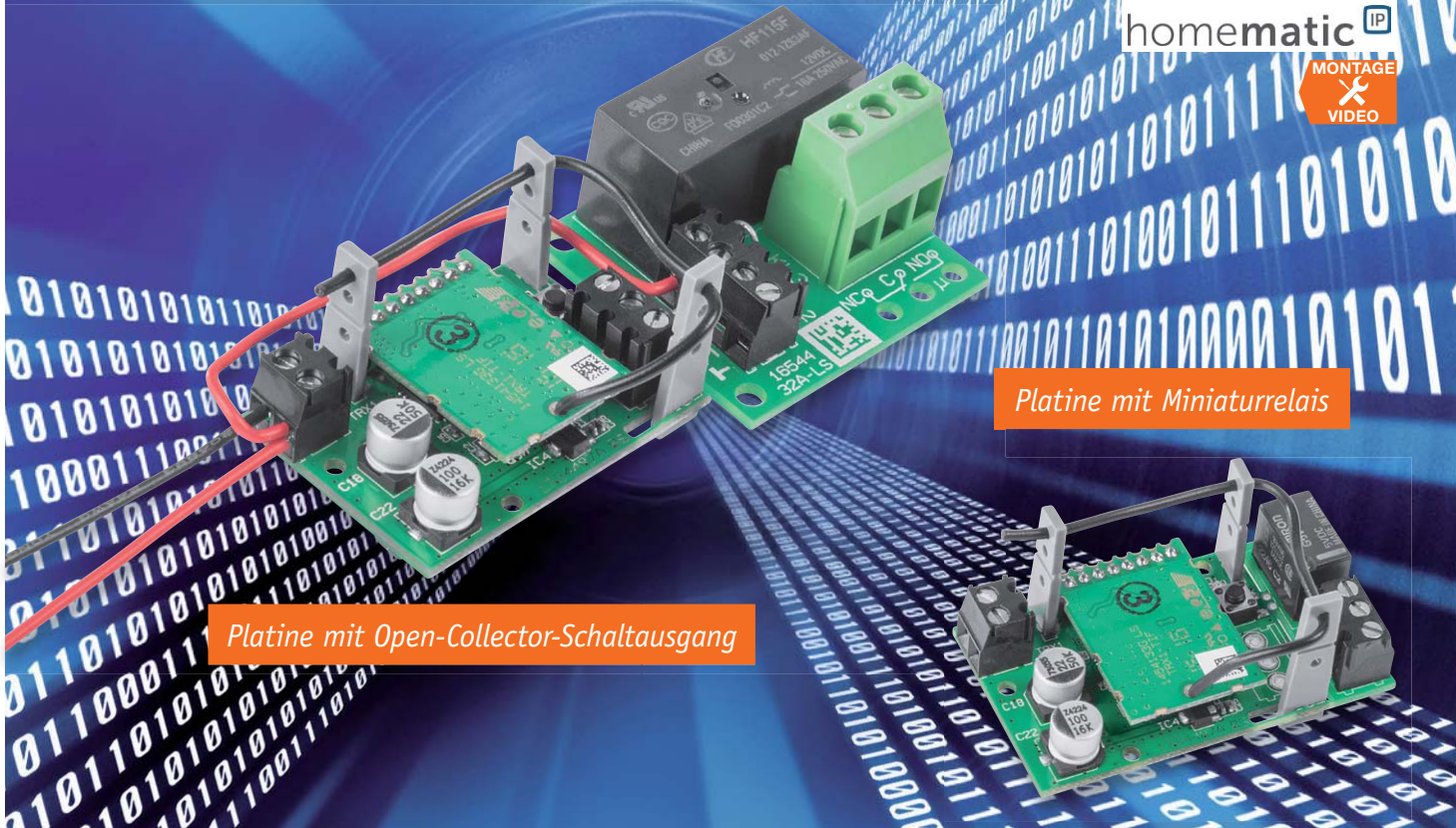
Eine weitere, sehr interessante Anwendung für die Scriptprogrammierung ist das Arbeiten mit Komponenten, die über eine sogenannte API anzusteuern sind.

Eine API ist eine Programmierschnittstelle, genauer gesagt eine Schnittstelle zur Anwendungsprogrammierung (englisch *Application Programming Interface*, wörtlich „Anwendungs-Programmier-Schnittstelle“).

Anwendungsfälle finden wir z. B. in der Ansteuerung von Audio-Anlagen, der Abfrage von Daten wie z. B. Wetterdaten und viele weitere mehr.

Ausblick

Im fünften Teil dieser Reihe werden wir uns also mit der Kommunikation mit Geräten mit API-Schnittstellen beschäftigen und weitere Scripte schreiben. 



Platine mit Open-Collector-Schaltausgang

Platine mit Miniaturrelais

Homematic IP Schaltplatine

Universeller Aktor für Kleinspannung

Infos zum Bausatz

im ELV Shop

#10056

Der kompakte Schaltaktor für das Homematic IP System vereinfacht Eigenbauprojekte im Kleinspannungsbereich, er stellt dabei entweder einen Open-Collector-Schaltausgang oder einen potentialfreien Relais-Umschaltkontakt bereit.

Geräte-Kurzbezeichnung:	HmIP-PCBS
Versorgungsspannung:	5 V _{DC} -25 V _{DC}
Stromaufnahme ohne/mit Relais:	50 mA max./70 mA max.
Relais:	Typ: Wechsler, 1-pol., μ -Kontakt
Lastart:	ohmsche Last
Maximale Schaltspannung:	30 V
Maximaler Schaltstrom:	1 A
Transistor-Schaltausgang:	
Typ:	Open-Collector
Maximale Schaltspannung:	30 V
Maximaler Schaltstrom:	0,5 A
Leitungsart und -querschnitt:	starre und flexible Leitung, 0,75-1 mm ²
Leitungslängen:	50 cm max.
Empfängerkategorie:	SRD-Category 2
Funkfrequenz:	868,3 MHz/869,525 MHz
Typ. Funk-Freifeldreichweite:	350 m
Duty-Cycle:	< 1 % pro h/< 10 % pro h
Schutzklasse:	III
Verschmutzungsgrad:	2
Umgebungstemperatur:	-10 bis +55 °C
Abmessungen (B x H x T):	28 x 48 x 21 mm (mit Relais)
Gewicht:	13 g

Technische Daten

Die eigene Lösung für Homematic IP

Ähnlich wie das Homematic System wird auch das System Homematic IP ständig mit weiteren Komponenten ausgebaut. Schnell entsteht auch hier der Wunsch nach individuell in eigene Applikationen einsetzbaren Komponenten. Genau hier setzt der kleine Universal-Aktor HmIP-PCBS an.

Die kleine Schaltplatine lässt sich mit anderen Homematic IP Sendern steuern und schaltet dabei Ströme bis zu 0,5 A am Open-Collector-Ausgang und bis zu 1 A am potentialfreien Wechslerkontakt des Kleinspannungsrelais.

Reicht hingegen die Schaltleistung des Miniaturrelais nicht für die vorgesehene Aufgabe aus, kann der Aktor auch sehr einfach mit dem im ELV Journal 1/2017 vorgestellten Relais-Schaltmodul RSM1 [1] verbunden werden. Die Klemmen wurden deshalb so positioniert und belegt, dass beide Schaltungen direkt aneinanderpassen.

Wie alle anderen Homematic IP Geräte lässt sich auch der Aktor über die Zentrale des Homematic Systems ansteuern, u. a. mit dem Vorteil, dass hier eine sehr hohe Funkreichweite erzielbar ist.

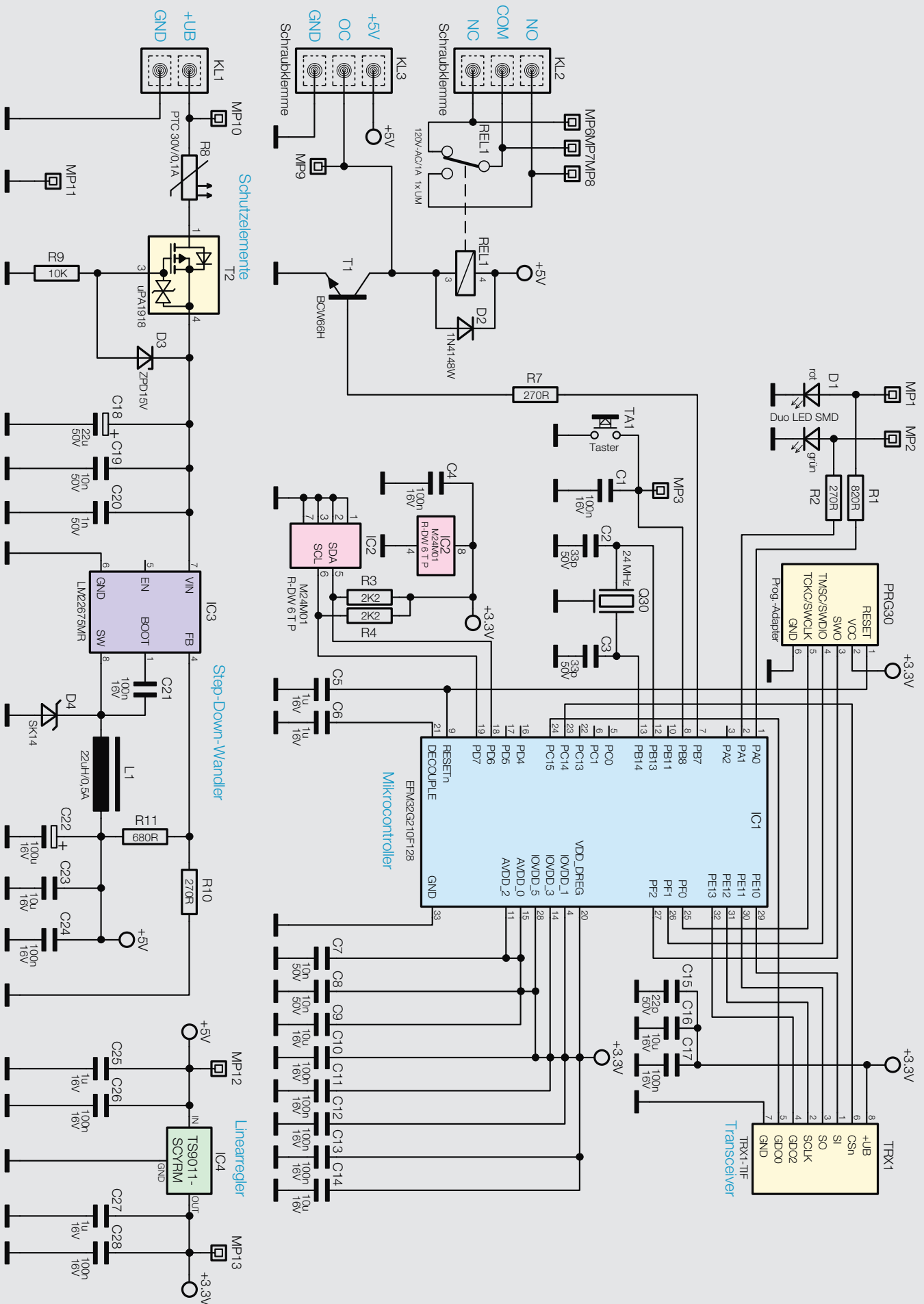


Bild 1: Das Schaltbild des Schaltaktors HmIP-PCBS



Schaltung

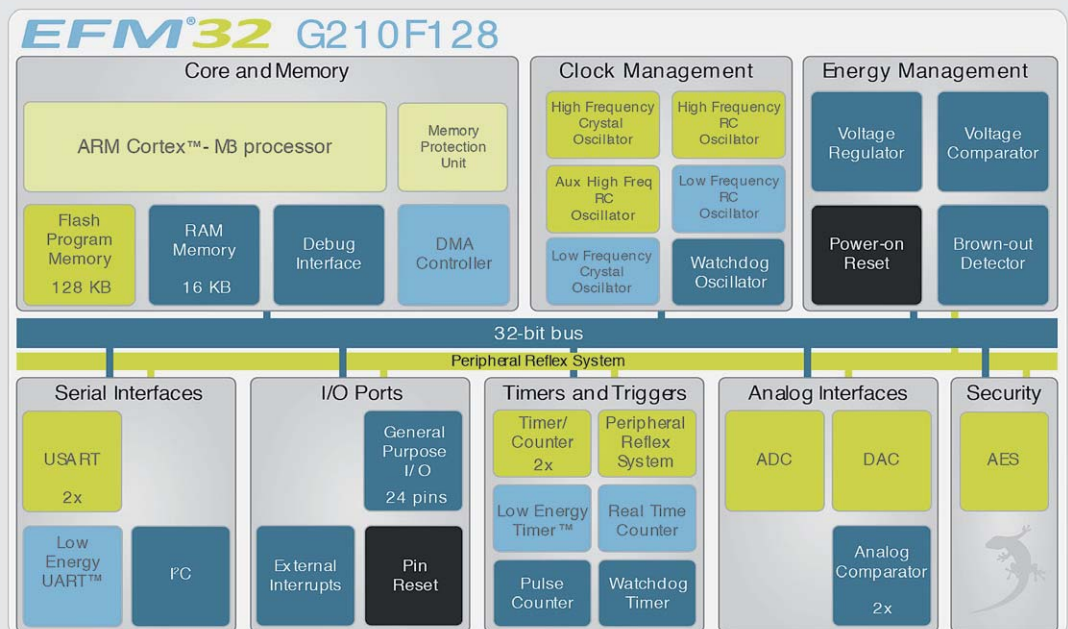
Der Schaltplan (Bild 1) unterscheidet sich, wie auch die Platine, kaum vom HM-LC-Sw1-PCB. Der steuernde Mikrocontroller ist beim HmIP-PCBS allerdings ein EFM32G210F128, der zusätzlich in einem extern per I²C angeschlossenen EEPROM IC2 dauerhaft Daten speichern kann. Mit dem Quarz Q30 wird vom Controller ein Arbeitstakt von 24 MHz erzeugt, mit dem er nicht nur schnell arbeiten kann, sondern auch Timings bei automatischen Abläufen relativ genau einhalten kann. Über die Schnittstelle PRG30 wird der Controller ab Werk mit dem Bootloader (siehe [Elektronikwissen](#)) und der Applikationsfirmware versehen. Per Funk lässt sich letztere später im laufenden Betrieb aktualisieren.

Mit dem Taster TA1 kann sowohl ein Werksreset des Aktors durchgeführt als auch der Schaltausgang lokal bedient werden. Üblicherweise wird der Ausgang aber

natürlich per Funk gesteuert, wozu der Controller per SPI-Schnittstelle mit dem Funkmodul TRX1 verbunden ist.

Am Port PB7 des Controllers ist der Schaltausgang des Aktors angeschlossen, wobei der Transistor T1 zum einen als Open-Collector-Ausgang an KL3 nutzbar ist, wenn der Relais-Platinenteil abgebrochen wird, oder der Transistor T1 treibt das auf der abtrennbaren Platine befindliche kleine Umschaltrelais REL1 an, wodurch dann an KL2 ein potentialfreier Umschaltkontakt zur Verfügung steht. „COM“ bezeichnet dabei den Umschaltkontakt, während „NC“ den im Ruhezustand geschlossenen und „NO“ den im Ruhezustand geöffneten Kontakt bezeichnen.

Zur Spannungsversorgung des Aktors können Gleichspannungen von 5 V bis 25 V an KL1 angeschlossen werden. Über den als selbststrückstellende Sicherung arbeitenden PTC-Widerstand R8 und den



Bootloader

Ein Bootloader ist ein kleines Programm, das sich entweder fest programmiert in einem Mikrocontroller befindet oder in den Flash-Speicher des Mikrocontrollers ladbar ist. Das Programm hat die Aufgabe, das eigentliche Programm (Firmware) auf einfache Weise laden und austauschen zu können. Dies erfolgt üblicherweise über eine der seriellen Schnittstellen des Mikrocontrollers (UART, USB oder I²C), und es wird keine spezielle Programmier-Hardware (Flash-Programmer) für das Laden der Firmware benötigt. So ist es z. B. möglich, die Firmware per Funk (OTA, **O**ver **T**he **A**ir) auszutauschen, nachdem man den Controller in den Programmierzustand versetzt hat.

Der Bootloader befindet sich in einem geschützten Speicherbereich (Protected Memory), der so nicht versehentlich löschtbar ist.

Beim im HmIP-PCBS verwendeten EFM32-Controller, einem 32-Bit-ARM-Cortex-M3-Controller der Gecko-Familie, ist der UART-Bootloader ab Werk vorprogrammiert. Für eigene Applikationen, die den gesamten Speicher des Mikrocontrollers benötigen, kann hier der Bootloader auch überschrieben werden.

Um eine neue Firmware über den vorhandenen Bootloader einzuspielen, ist hier der Controller durch ein Reset, das ausgeführt wird, während der Pin „DBG_SWCLK“ auf High liegt, in den Programmierzustand versetzbar. Danach kann das als Binärfile vorliegende Firmware-Programm über die UART-Schnittstelle aufgespielt werden.

Weitere Hinweise dazu finden sich in der Application Note AN 0003 von Silicon Labs:

<https://www.silabs.com/products/mcu/Pages/32-bit-mcu-application-notes.aspx>



Homematic IP	Homematic IP Gerät mit Internetzugang anlernen Homematic IP Geräte können auch über die CCU2 angelernt werden. Klicken Sie auf den Button "HmIP Gerät anlernen". Der Anlernmodus der CCU2 ist dann für 60 Sekunden aktiv. Aktivieren Sie innerhalb dieser Zeit den Anlernmodus des Homematic IP Gerätes, das angelernt werden soll.	Homematic IP Gerät ohne Internetzugang anlernen Homematic IP Geräte können auch ohne aktiven Internetzugang an die CCU2 angelernt werden. Geben Sie die SGTIN und den KEY ein und klicken Sie auf "HmIP Gerät anlernen (lokal)".
	<input type="button" value="Anlernmodus nicht aktiv"/> <input type="button" value="HmIP Gerät anlernen"/>	SGTIN <input type="text"/> KEY <input type="text"/> <input type="button" value="Anlernmodus nicht aktiv"/> <input type="button" value="HmIP Gerät anlernen (lokal)"/>

Bild 2: Über den Anlernmodus der WebUI kann das Schaltmodul an die CCU2 angelernt werden.

als Verpolungsschutz dienenden Transistor T2 gelangt die Betriebsspannung zum Step-down-Wandler IC3, der mit R10 und R11 so eingestellt ist, dass an seinem Ausgang knapp 5 V erzeugt werden. Diese dienen einerseits der Versorgung des Miniaturrelais auf der Schaltplatine, andererseits erzeugt der Linearregler IC4 hieraus stabilisierte 3,3 V für den Betrieb der Logikkomponenten.

Bedienung

Über die Taste TA1 lässt sich der Aktor mit einem Tastendruck ein- und ausschalten. Der eingeschaltete Zustand wird durch die LED mit grünem Leuchten angezeigt.

Soll an dem Aktor ein Werksreset vorgenommen werden, ist der Aktor mit gedrückter Taste mit der Versorgungsspannung zu verbinden und der Taster weitere 4 Sekunden gedrückt zu halten, bis die LED orange blinkt. Nach kurzem Loslassen des Tasters ist dieser erneut für 4 Sekunden zu betätigen, bis die LED grün leuchtet. Jetzt erfolgt der Reset, und die Taste kann losgelassen werden. Um den Aktor an eine Homematic Zentrale oder an einen Homematic IP Access-Point anzuler-

nen, ist bei dem jeweiligen Zentralelement zuerst der entsprechende Anlernmodus zu starten. In Bild 2 ist der relevante Teilausschnitt aus dem CCU2-Dialog zu sehen. Danach sollte ein kurzer Tastendruck am Aktor vorgenommen werden, wenn dieser bereits länger als 3 Minuten an seiner Versorgungsspannung angeschlossen ist.

Der Screenshot in Bild 3 zeigt die Konfigurationsmöglichkeiten des Aktors in Verbindung mit einer Homematic Zentrale CCU2.

Hier wird auch die im Aktor verwendete Kanalstruktur sichtbar. Geräteübergreifende Parameter sind dem Kanal 0 zugeordnet. Hier lassen sich die zyklischen Statusmeldungen des Aktors deaktivieren oder ihr Intervall anpassen. Ebenso kann hier die Reset-Funktion am Aktor gesperrt werden, damit der Aktor nicht versehentlich oder mutwillig durch Unbefugte in den Werkszustand versetzt werden kann.

Name	Kanal	Parameter
HmIP-PCBS 000453C262633D:0	Ch.: 0	Zyklische Statusmeldung <input checked="" type="checkbox"/> Anzahl der auszulassenden Statusmeldungen <input type="text" value="20"/> (0 - 255) Anzahl der auszulassenden, unveränderlichen Statusmeldungen <input type="text" value="0"/> (0 - 255) Reset per Gerätetaste sperren <input type="checkbox"/>
HmIP-PCBS 000453C262633D:1 Tasterkanal	Ch.: 1	Doppelklick-Zeit (Tastensperre) <input type="text" value="0.00"/> s (0.00 - 25.50) Mindestdauer für langen Tastendruck <input type="text" value="0.40"/> s (0.00 - 25.50) Timeout für langen Tastendruck <input type="text" value="2 Minuten"/>
HmIP-PCBS 000453C262633D:2 Statusmitteilung Relais	Ch.: 2	Eventverzögerung <input type="text" value="1 Sekunde"/> Zufallsanteil <input type="text" value="1 Sekunde"/> Geräte-LED deaktivieren <input type="checkbox"/>
HmIP-PCBS 000453C262633D:3 Schaltaktor	Ch.: 3	Verknüpfungsregel <input type="text" value="OR (höherer Pegel hat Priorität)"/> Aktion bei Spannungszufuhr <input type="text" value="Schaltzustand: Aus"/> Einschaltverzögerung <input type="text" value="Nicht aktiv"/> Einschaltdauer <input type="text" value="Nicht aktiv"/>
HmIP-PCBS 000453C262633D:4 Schaltaktor	Ch.: 4	Verknüpfungsregel <input type="text" value="OR (höherer Pegel hat Priorität)"/> Aktion bei Spannungszufuhr <input type="text" value="Schaltzustand: Aus"/> Einschaltverzögerung <input type="text" value="Nicht aktiv"/> Einschaltdauer <input type="text" value="Nicht aktiv"/>
HmIP-PCBS 000453C262633D:5 Schaltaktor	Ch.: 5	Verknüpfungsregel <input type="text" value="OR (höherer Pegel hat Priorität)"/> Aktion bei Spannungszufuhr <input type="text" value="Schaltzustand: Aus"/> Einschaltverzögerung <input type="text" value="Nicht aktiv"/> Einschaltdauer <input type="text" value="Nicht aktiv"/>

Bild 3: Die Konfigurationsmöglichkeiten des HmIP-PCS in der WebUI der CCU2

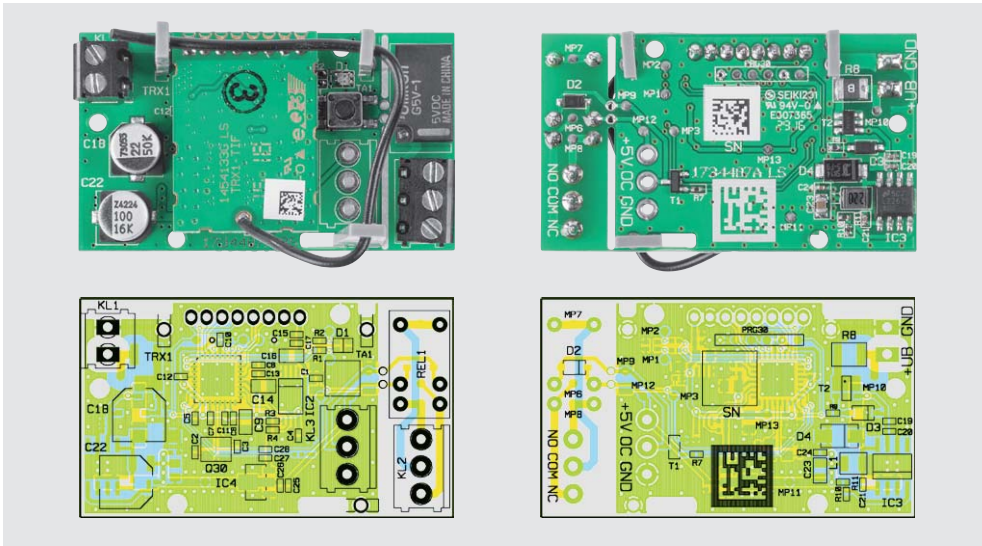


Bild 4: Die Platinenfotos des Schaltaktors (mit Schaltrelais-Bestückung) mit den zugehörigen Bestückungsplänen

Montagevideo

#10060

QR-Code scannen oder Webcode im Web-Shop eingeben

Kanal 1 des Aktors ist die Gerätetaste. Hier lässt sich das Verhalten der kurzen und langen Tastendrucke anpassen.

Der Aktor verwendet zudem die aus einigen Homematic Dimmern bereits bekannten virtuellen Aktorkanäle, die durch konfigurierbare Logikfunktionen ausgefeilte Verhaltensweisen mittels Direktverknüpfungen erlauben (siehe Elektronikwissen „Virtuelle Homematic Aktorkanäle und ihre Verknüpfungslogik“ unter <https://www.elv.de>, Webcode #10061).

Kanal 2 dient lediglich der Statusmeldung des realen Verknüpfungsergebnisses und spiegelt den Zustand des Relais wider. Hier kann das Sendeverhalten bei Zustandsänderungen konfiguriert werden. Weiterhin lässt sich hier die Status-LED des Aktors abschalten.

Kanal 3 bis 5 sind die virtuellen Aktorkanäle, zu denen Direktverknüpfungen mit HmIP Sendern hergestellt werden können. Neben der Verknüpfungslogik kann hier für jeden Kanal festgelegt werden, wie er sich bei Spannungszufuhr verhalten soll. Dabei ist es auch möglich, den Aktorkanal nur zeitlich begrenzt oder verzögert einzuschalten.

Weitere Hinweise zur Bedienung und Einbindung in das Homematic System finden sich in der zu jedem Gerät mitgelieferten Bedienungsanleitung und dem Homematic WebUI-Handbuch. Aktuelle Versionen davon sind immer im Downloadbereich von eQ-3 zu finden.

Nachbau

Wie gewohnt ist die Platine des Aktors bereits komplett mit SMD-Bauteilen vorbestückt, sodass nach einer Bestückungs- und Lötfehlerkontrolle, bei der die Platinenfotos und die Bestückungspläne in Bild 4 herangezogen werden, nur noch wenige bedrahtete Bauteile sowie das Transceivermodul zu bestücken sind.

Bevor man jedoch damit beginnt, sollte man entschieden haben, ob man das Modul mit dem internen Miniaturrelais oder ohne dieses (Nut-

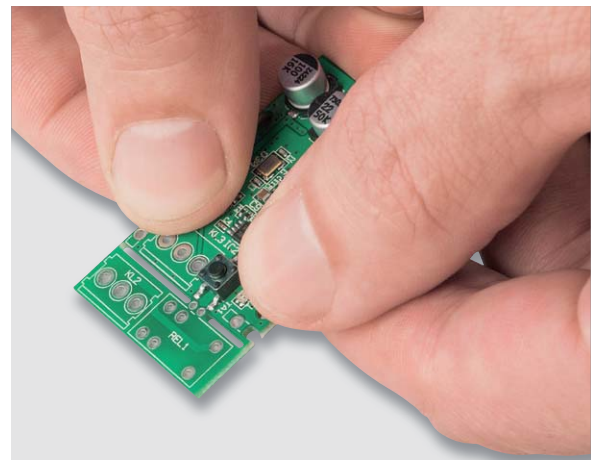


Bild 5: So erfolgt das Abbrechen des Platinenteils für das Miniaturrelais, falls man dieses nicht nutzen möchte.

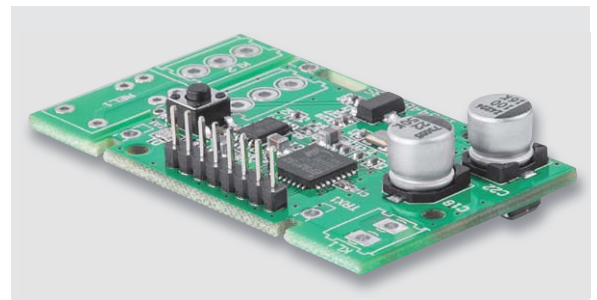


Bild 6: Die Bestückung der Stiftleiste erfolgt wie hier gezeigt, indem die langen Stifte nach oben stehen.

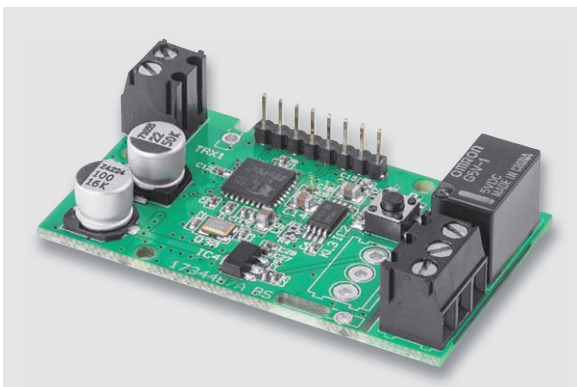


Bild 7: Die Bestückung der Schraubklemmen und des Relais

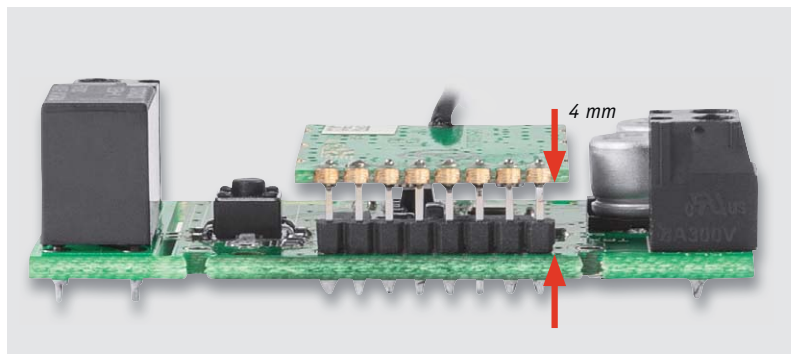


Bild 8: Das Transceivermodul ist genau parallel zur Aktorplatine und mit einem Abstand von 4 mm einzulöten.



zung des Open-Collector-Ausgangs für das Schalten von Schalteingängen oder des Relais-Schaltmoduls RSM1) nutzen will.

Im zweiten Fall sollte man die kleine Relaisplatine an der perforierten Stelle abbrechen (Bild 5), um das Aktormodul möglichst klein zu halten. In diesem Falle ist auch die dreipolige Schraubklemme auf der Aktorplatine als KL3 zu bestücken.

Wir gehen in der weiteren Beschreibung jedoch davon aus, dass das Miniaturrelais REL1 genutzt wird.

Die Bestückung beginnt mit der 8-poligen Stiftleiste für das Transceivermodul. Diese ist, wie in Bild 6 zu sehen, so zu bestücken, dass die kürzeren Stifte der Stiftleiste exakt senkrecht in die Platine eingesetzt und verlötet werden.

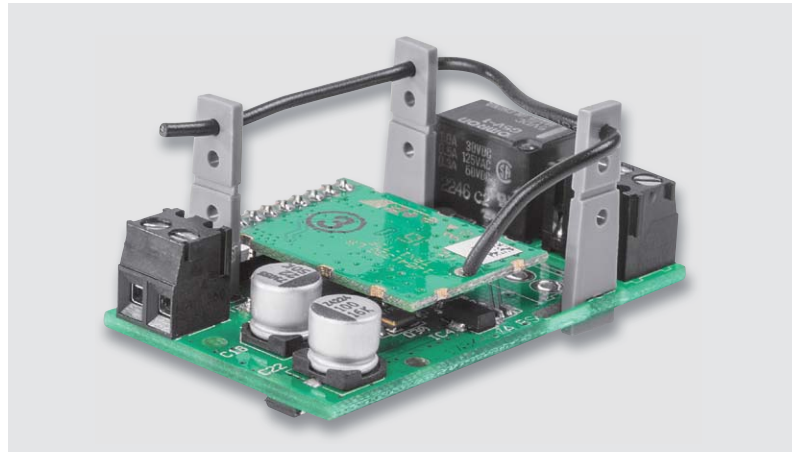
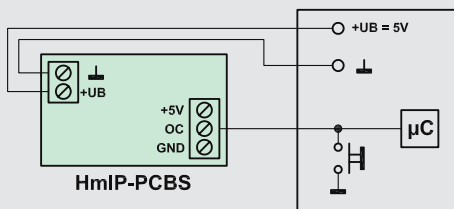
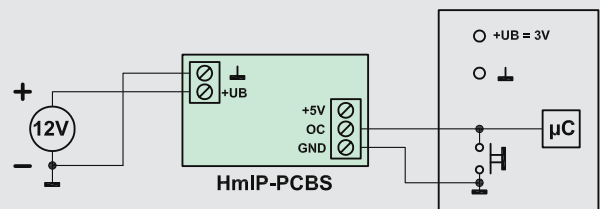


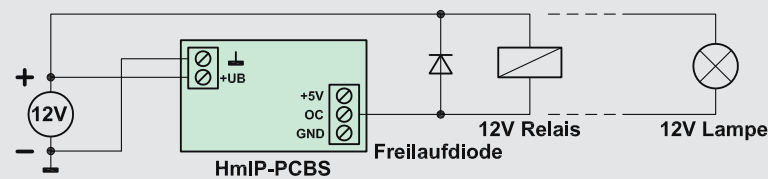
Bild 9: Hier sind die eingesetzten Antennenhalter sowie die Lage der Antenne zu sehen.



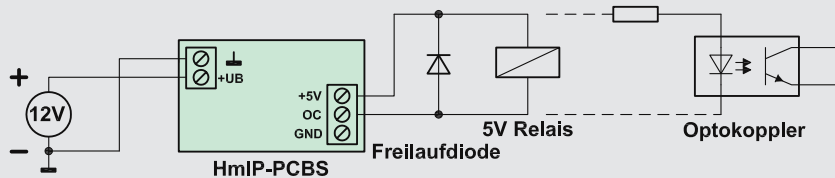
Ansteuerung eines Schalteingangs, hier einer Mikroprozessorschaltung, und Spannungsversorgung aus dieser Schaltung



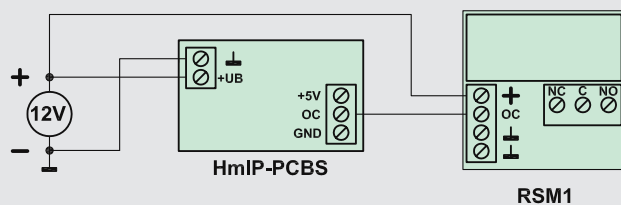
Ansteuerung eines Schalteingangs, hier einer Mikroprozessorschaltung, und Spannungsversorgung aus eigener Spannungsquelle



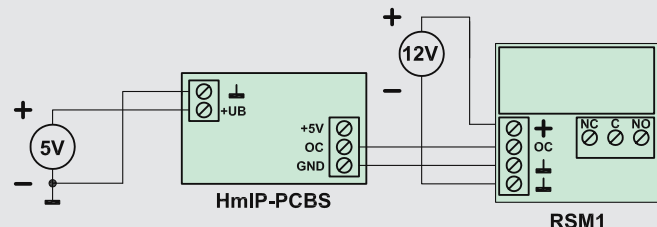
Ansteuerung eines externen Relais (mit Freilaufdiode) oder einer Last bis 0,5 A mit Laststromversorgung aus der Eingangsspannung



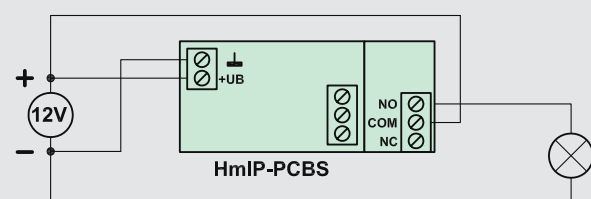
Ansteuerung eines externen 5-V-Relais (mit Freilaufdiode) oder eines Optokopplers bzw. einer LED (Vorwiderstand je nach Bauelement)



Ansteuerung des Relaismoduls RSM1 mit Versorgung des Relaismoduls aus der Aktor-Betriebsspannung. Diese Konfiguration ist auch in Bild 11 dargestellt.



Ansteuerung des Relaismoduls RSM1 mit eigenständiger Versorgung des Relaismoduls



Ansteuerung einer Last (max. 0,5 A) mit direkter Versorgung aus der Aktor-Betriebsspannung

Bild 10: Der Schaltaktor ermöglicht zahlreiche verschiedene Spannungsversorgungs- und Anschlussmöglichkeiten für Relais und Lasten, u. a. auch für das Relaismodul RSM1.



Dem folgt die Bestückung der beiden Schraubklemmen, wobei für KL2/KL3 die o. a. Ausführungen zu beachten sind. Hier ist darauf zu achten, dass die Schraubklemmen mit ihrem Körper plan auf der Platine sitzen und die Leitungsöffnungen nach außen zeigen. In Bild 7 ist die Bestückung gemeinsam mit dem Miniaturrelais REL1 auf der Relaisplatine als KL2 zu sehen.

Schließlich ist das Transceivermodul so auf der Stiftleiste zu bestücken, dass dessen Bauteilseite nach unten und die Antenne nach oben zeigt (Bild 8). Das Modul ist so mit der Stiftleiste zu verlöten, dass es genau parallel zur Aktorplatine liegt.

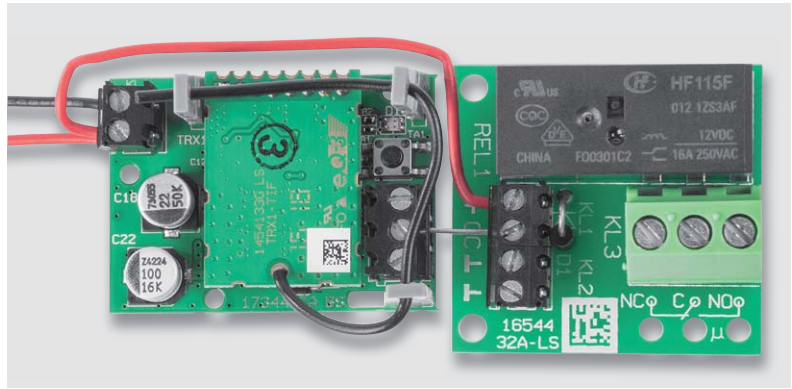


Bild 11: Der Homematic IP Aktor mit angeschlossenem Relais-Schaltmodul RSM1

Widerstände:

270 Ω /SMD/0402	R2, R7, R10
820 Ω /SMD/0402	R1
680 Ω /SMD/0402	R11
2,2 k Ω /SMD/0402	R3, R4
10 k Ω /SMD/0402	R9
PTC/0,1 A/30 V/SMD/1210	R8

Kondensatoren:

22 pF/50 V/SMD/0402	C15
33 pF/50 V/SMD/0402	C2, C3
1 nF/50 V/SMD/0402	C20
10 nF/50 V/SMD/0402	C7, C8, C19
100 nF/16 V/SMD/0402	C1, C4, C10–C13, C17, C21, C24, C26, C28
1 μ F/16 V/SMD/0402	C5, C6, C25, C27
10 μ F/16 V/SMD/0805	C9, C14, C16, C23
22 μ F/50 V	C18
100 μ F/16 V	C22

Halbleiter:

ELV161532/SMD	IC1
M24M01-DF DW 6 T G/TSSOP-8	IC2

LM22675MRX-ADJ/SMD	IC3
TS9011SCY RM/SMD	IC4
BCW66H/SMD	T1
uPA1918TE/SMD	T2
SK14/SMD	D4
1N4148W/SMD	D2
MMSZ5245B/SOD-123	D3
Duo-LED/rot/grün/SMD	D1

Sonstiges:

Speicherdrossel, SMD, 22 μ H / 450 mA	L1
Quarz, 24.000 MHz, SMD	Q30
Sender-/Empfangsmodul TRX1-TIF, 868 MHz	TRX1
Stiftleiste, 1 x 8-pol., 6 mm, gerade, print	TRX1
Relais, coil: 5 V, 1 Form C (CO) 1x toggle, 30 VDC, 120 VAC, 1 A DC, 1 A AC	REL1
Taster mit 0,9-mm-Tastknopf, 1x ein, SMD, 3,8 mm Höhe	TA1
Schraubklemme, 2-polig, Drahteinführung 90°, RM=3,5 mm, THT, black	KL1
Schraubklemme, 3-pol., Drahteinführung 90°, RM=3,5 mm, THT, black	KL2
Antennenhalter für Platinen	
QR-Code-Aufkleber für HmIP Geräte, weiß	



Wichtiger Hinweis:

Für einen ausreichenden Schutz vor elektrostatischen Entladungen ist der Einbau in ein geeignetes Gehäuse erforderlich, damit die Schaltung nicht durch eine Berührung mit den Fingern oder Gegenständen gefährdet werden kann. Das Gehäuse muss die Anforderungen an eine Brandschutzumhüllung gemäß EN 60950-1 erfüllen. Beim Einbau des HmIP-PCBS in ein Gehäuse ist darauf zu achten, dass eine ausreichende Zugentlastung nach den allgemeinen Vorschriften vorzusehen ist. Die Leitungen dürfen nicht durch scharfe Kanten an den Einführungsöffnungen beschädigt werden.

Dabei ist es im Interesse der Störsicherheit wichtig, dass ein Mindestabstand von 4 mm zur Aktorplatine eingehalten wird. Das heißt, dass das Transceivermodul nicht auf dem Kunststoffkörper der Stiftleiste aufliegen darf, siehe Bild 8.

Abschließend erfolgt nun das Einsetzen und Einrasten der drei Antennenhalter in die Aktorplatine und das Einlegen der Antenne in die obersten Löcher der Antennenhalter, wie in Bild 9 gezeigt.

Damit ist der Aufbau beendet, und der Aktor kann entsprechend Bild 10, in dem verschiedene Applikationen und Anschlussvarianten gezeigt sind, in die eigene Applikation eingebunden werden. Unter anderem ist hier auch die Verbindung mit dem Relais-Schaltmodul RSM1 zu sehen, in Bild 11 ist diese noch einmal praktisch ausgeführt zu sehen.

Beim Anschluss von Leitungen ist zu beachten, dass die maximale Leitungslänge auf 50 cm begrenzt ist. Außerdem darf nur 1 Leiter pro Klemme angeschlossen werden.



Weitere Infos:

[1] Relais-Schaltmodul RSM1: ELV Journal 1/2017, Seite 32



Manfred Gontjes, Torsten Boekhoff, Pascal Junge, Werner Müller, Andreas Bünting, Marco Fenbers, Marco Angenendt (von links)

Dialog

Experten antworten

Sie suchen Beratung oder haben Fragen zu Ihrem ELV Projekt? Wir helfen bei Ihrem Projekt! Jeden Tag beantworten wir Hunderte von Fragen per E-Mail oder Telefon. Dieses Wissen stellen wir Ihnen im Internet zur Verfügung. Die wichtigsten Fragen zum Produkt finden Sie im ELV Shop direkt beim Artikel. Mittlerweile ist so eine umfassende Datenbank entstanden.

Nützliche HomeMatic Tipps

Wir zeigen Ihnen, wie sich bestimmte Aufgabenstellungen im HomeMatic System lösen lassen. Die beschriebenen Lösungsmöglichkeiten sollen insbesondere HomeMatic Einsteigern helfen, die Einsatz- und Programmiermöglichkeiten von HomeMatic besser bzw. optimaler nutzen zu können.

Webcode #10020 im Suchfeld eingeben



Gerne können Sie auch das ELV Technik-Netzwerk nutzen, um sich mit anderen Technikbegeisterten über Ihre Anliegen auszutauschen.

www.netzwerk.elv.de

Technische Fragen?

Sie erreichen uns in der Zeit von Montag bis Freitag von 9:00 bis 19:00 Uhr. Halten Sie bitte Ihre ELV Kundennummer (wenn vorhanden) bereit.

Tel.: 0491/6008-245

E-Mail: technik@elv.de

Frage von Herrn Herbert Groß zum HomeMatic Funk-Dimmaktor für Markenschalter (Best.-Nr. CN-10 30 20):

Während der Einstellung der Helligkeit einer dimmbaren LED-Lampe habe ich festgestellt, dass sich die Helligkeit effektiv nur in einem Bereich zwischen 30 und 75 % ändert. Lässt sich die Einstellung des Funk-Dimmaktors für Markenschalter dahingehend anpassen, dass dieser nur in diesem Bereich dimmt? Ist es außerdem möglich, dass beim Wiedereinschalten die zuletzt eingestellte Helligkeit gegeben ist?



Antwort von ELV: Der Funk-Dimmaktor für Markenschalter verfügt neben dem „Steuerkanal“ (Kanal 1 des Dimmers) über zwei weitere (virtuelle) Kanäle (Kanal 2 und Kanal 3). Diese lassen sich auf vielfältige Art und Weise mit dem Steuerkanal verknüpfen. In Ihrem Fall bietet sich die Funktion „MULTI MULTI“ an, bei welcher ein beim Kanal 2 gesetzter Pegel mit dem auf dem Steuerkanal 1 eingestellten Pegel multipliziert wird. Nehmen Sie die nachfolgenden Einstellungen vor, durch welche der Kanal 2 im „MULTI MULTI“-Mode betrieben wird; der Druck auf eine der beiden Gerätetasten bewirkt das Setzen auf den Pegel von 75 %.

Verknüpfungsregel	MULTI MULTI (Pegel-Multiplikation Maximum = 100%)	Hilfe
Aktion bei Spannungszufuhr	keine	
Statusmeldungen Mindestverzögerung	Wert eingeben 2	s (0.50 - 15.50)
Statusmeldungen Zufallsanteil	1.0	s (0.00 - 7.00)
Max. Senderversuche	6	(0 - 10)
Programmierung der 1. internen Gerätetaste - #888888881:		
Dimmer - ein		
Das Licht wird durch kurzen oder langen Tastendruck auf den festgelegten Helligkeitswert eingeschaltet.		
Ch.: 2		
Rampenzeit beim Einschalten	0.5s	
Verweildauer im Zustand "ein"	unendlich	
Pegel im Zustand "ein"	Wert eingeben 75 %	
Programmierung der 2. internen Gerätetaste - #888888882:		
Dimmer - ein		
Das Licht wird durch kurzen oder langen Tastendruck auf den festgelegten Helligkeitswert eingeschaltet.		
Rampenzeit beim Einschalten	0.5s	
Verweildauer im Zustand "ein"	unendlich	
Pegel im Zustand "ein"	Wert eingeben 75 %	

Programmierung der 1. internen Gerätetaste - #888888881:		
Dimmer - aus/dunkler		
Mit einem kurzen Tastendruck wird das Licht ausgeschaltet. Ein langer Tastendruck dimmt das Licht herunter:		
Ausschaltverzögerung	keine	
Blinken in der Ausschaltverzögerung	ein	
Rampenzeit beim Ausschalten	0.5s	
Zusätzliche Einstellung für den langen Tastendruck:		
Pegelbegrenzung beim Herunterdimmen	40%	
Simuliere Tastendruck		
Programmierung der 2. internen Gerätetaste - #888888882:		
Dimmer - ein/heller		
Mit einem kurzen Tastendruck wird das Licht auf den festgelegten Helligkeitswert eingeschaltet. Ein langer Tastendruck dimmt das Licht hoch.		
Rampenzeit beim Einschalten	0.5s	
Verweildauer im Zustand "ein"	unendlich	
Pegel im Zustand "ein"	Letzter Wert	
Zusätzliche Einstellung für den langen Tastendruck:		
Pegelbegrenzung beim Hochdimmen	100%	
Simuliere Tastendruck		



Bei dem Kanal 1 des Dimmers stellen Sie die Pegelbegrenzung beim Herunterdimmen auf 40 % ein und den „Pegel im Zustand ‚ein‘“ stellen Sie auf „Letzter Wert“.

Durch diese Einstellungen ergeben sich die folgenden Funktionen: Durch Betätigung der Gerätetaste lässt sich ein Pegel (Kanal 1) zwischen 40 % und

100 % einstellen. Dieser Pegel wird mit dem fest auf 75 % gesetzten Pegel von Kanal 2 multipliziert, sodass sich folgende Min.- und Max.-Pegel ergeben:

$$\text{Min.: } 40 \% \times 75 \% = 30 \%$$

$$\text{Max.: } 100 \% \times 75 \% = 75 \%$$

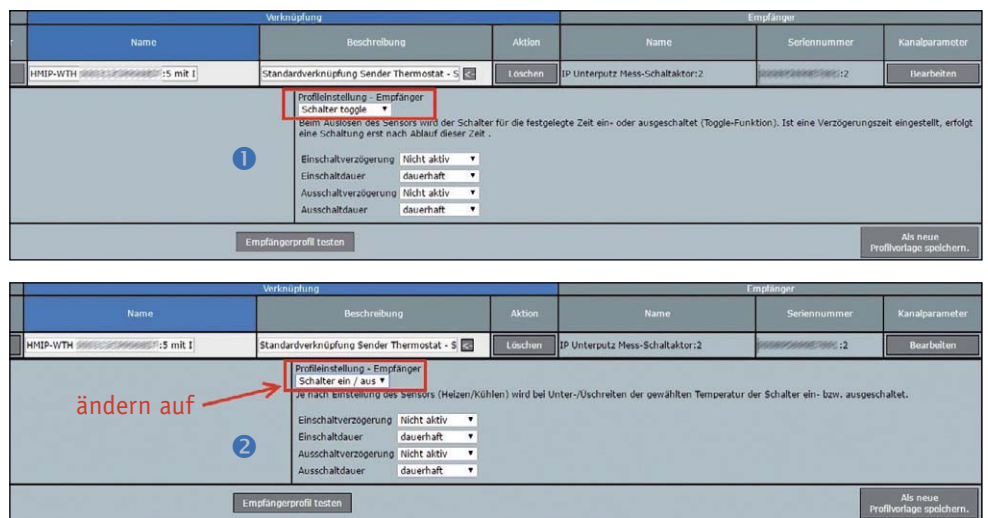
Durch die Einstellung „Pegel im Zustand ‚ein‘“ = „Letzter Wert“ wird der zuletzt gegebene Dimmwert nach dem Einschalten wieder gesetzt.

Frage von Herrn Hans-Joachim Weidemeyer zum Homematic IP Schalt-Mess-Aktor HmIP-FSM (Best.-Nr. CN-14 27 21):

Ich möchte mit dem Homematic IP FSM-Schaltaktor und dem Homematic IP Wandthermostaten einen elektrischen Radiator temperaturabhängig ein- und ausschalten. Ich habe hierzu über die Homematic Zentrale CCU2 eine direkte Verknüpfung zwischen diesen beiden Teilen erstellt. Der Funktionstest hat jedoch ergeben, dass der Schaltaktor offensichtlich willkürlich ein- und ausschaltet. Liegt hier eventuell ein Einstellungsfehler vor?

Antwort von ELV: Eine Verknüpfung zwischen dem Homematic IP Unterputz-Schaltaktor und dem Homematic IP WTH-Wandthermostaten lässt sich über die Homematic Zentrale CCU2 realisieren. Allerdings ist als Standard die Toggle-Funktion bei dem Schaltaktor gesetzt, sodass im Heizbetrieb bei jeder Unterschreitung der gewünschten Soll-

temperatur ein- und ausgeschaltet wird (Bild 1). Ändern Sie daher nach der Erstellung der Verknüpfung die werkseitige Grundeinstellung des Unterputz-Schaltaktors von „Schalter Toggle“ auf „Schalter ein/aus“. Anschließend wird dann der Unterputz-Schaltaktor in Abhängigkeit von der gegebenen und der gewünschten Temperatur ein- und ausgeschaltet (Bild 2).



Frage von Herrn Frank Ludwig zum Homematic IP Fußbodenheizungsaktor (Best.-Nr. CN-14 29 81):

Ich bekomme in meinen Neubau eine Rotex-Wärmepumpe mit Kühlfunktion und habe mir die Homematic IP Komponenten Multi IO Box und Fußbodenheizungsaktor mit Wandthermostaten gekauft. Auch habe ich eine CCU2, weil ich weitere Funktionen im Haus über Homematic Komponenten steuere. Ich habe mir dann ein Programm gemacht, dass die Kanäle hin- und herschaltet und gleichzeitig den Homematic Wandthermostaten zwischen Heizen und Kühlen hin- und herschaltet.



tor wird auch nur geschaltet, wenn die Temperatur, die ich einstelle, höher als die Raumtemperatur ist, wie im Heizmodus. Mache ich einen Gedankenfehler?



Antwort von ELV: Sie haben nur den Wandthermostaten auf den Kühlbetrieb geschaltet. Es ist erforderlich, auch den Fußbodenheizungsaktor HmIP-FAL auf den Kühlbetrieb zu schalten. Ändern Sie das Programm bitte wie folgt:

- Aktivität: Dann...**
- Geräteauswahl > HmIP-WTH2 sofort Kühlen
- Geräteauswahl > HmIP-FAL sofort Kühlen
- Geräteauswahl > HmIP-MIOB sofort Schaltzustand Ein
- Systemzustand Heizungsbetrieb sofort Sommer
- Aktivität: Sonst...**
- Geräteauswahl > HmIP-WTH2 sofort Heizen
- Geräteauswahl > HmIP-FAL sofort Heizen
- Geräteauswahl > HmIP-MIOB sofort Schaltzustand Aus
- Systemzustand Heizungsbetrieb sofort Winter

Im Thermostat wird auch immer der Stern für Kühlen oder die Heizung für Heizen angezeigt, und im Heizbetrieb wird der Aktor auch geschaltet, wenn ich die Temperatur hochdrehe. Im Kühlbetrieb würde ich jetzt erwarten, dass er einschaltet, wenn ich die gewünschte Temperatur unter die Raumtemperatur bringe. Da passiert aber nichts. Der Ak-



Raspberry Pi

Teil 4: Hören, sehen und fühlen mit Sensoren – Ultraschall-Sensor

Artikelserie

im ELV Shop

#10036

Schallali, Schallala – der Ultraschall-Sensor ist da!

Ein Raspberry Pi und ein Ultraschall-Sensor – das ist zwar nicht die Liebe auf den ersten Blick, doch mithilfe des Sensors bleiben Sie auf Wunsch immer schön auf Distanz: Wer bereits mit einem Eigenbau-Roboter Erfahrung hat, der weiß oder hat dann auch zu Recht früher oder später Befürchtungen, dass der Roboter an alle Möbelstücke anstößt und dort seine Spuren hinterlässt.

Nicht nur um solche Dinge zu vermeiden macht der Einsatz eines Abstandssensors Sinn – auch jenseits der Robotik lassen sich verschiedene Anwendungsbereiche für Abstandssensoren finden. So lässt sich das Modul als Schaltsensor nutzen, beispielsweise wird eine Schaltung nur geöffnet oder geschlossen, wenn das Modul eine Bewegung in einem bestimmten Abstand misst.

Den Raspberry-Pi-tauglichen Ultraschall-Distanz-Sensor mit der Typenbezeichnung HC-SR04 (Bild 1) für das nachfolgend beschriebene Projekt können Sie im ELV Shop unter der Best.-Nr. CN-12 21 21 bestellen.

Laut Datenblatt des HC-SR04-Abstandssensors beträgt der Öffnungswinkel des Ultraschallsensors 15°, damit bietet er einen Messbereich von 2–400 cm bei einer maximalen Genauigkeit von 3 mm. In Sachen Spannungsversorgung benötigt das Modul die Raspberry-Pi-freundlichen 5 V und hat im Betrieb eine Stromaufnahme von 15 mA bzw. im Ruhezustand von weniger als 2 mA.

Abstandssensor – Schaltungsaufbau (Bild 2)

Damit Sie später mit einem Programm die Entfernung zwischen dem Sensor und einem Gegenstand messen können, verbinden Sie den Raspberry Pi mit dem HC-SR04-Modul, welches dafür vier Anschlüsse bereitstellt. Für die 5-V-Versorgungsspannung nutzen Sie den mit VCC beschrifteten Pin, für die gemeinsame Masse ist der gewohnte GND-Pin zuständig.

HC-SR04-Abstandssensor-Pin	Raspberry-Pi-Bezeichnung	Raspberry-Pi-Pin	Wiring Pi
VCC	5 V	2	–
TRIG (GER)	GPIO23 (Ausgang)	16	4
ECHO	GPIO24 (Eingang)	18	5
GND	Masse	6	–

Was für den Raspberry Pi ein Ausgang ist, ist bei dem Sensor der dazugehörige Eingangspin. Am Anschluss TRIG sendet die Logik einen 10-µs-Impuls, damit die Messung gestartet wird. In diesem Fall sendet das HC-SR04-Modul die Impulse aus und wartet auf das Echo.

Das Ergebnis der Entfernungsmessung wird über den Ausgang ECHO dem Raspberry Pi zur Verfügung gestellt. Beachten Sie, dass der zugeordnete Raspberry-Pi-GPIO-Eingang nur eine maximale Eingangsspannung von 3,3 V verträgt. Da der Abstandssensor jedoch mit 5-V-Spannung versorgt wird, sollten Sie im Praxiseinsatz hierfür sicherheitshalber einen Spannungsteiler verwenden. Hier sorgen dann zwei Widerstände dafür, dass die vom Sensor bereitgestellte Spannung von 5 V auf 3–3,3 V am GPIO-Eingang begrenzt wird.

Wie groß Sie die Widerstände dimensionieren, hängt in erster Linie von der Bastelkiste ab: Wichtig ist zu wissen, dass die Stromstärke in einer Reihenschaltung bekanntlich gleich bleibt. Um nun die Spannung von 5 V auf die gewünschten 3–3,3 V zu reduzieren, benötigen Sie hier also dasselbe Widerstandsverhältnis. Setzen Sie für beide Widerstände dieselben Werte ein, wird die Spannung halbiert (2,5 V). Ziel ist es also, hier ein Widerstandspaar zu finden, welches dafür sorgt, dass am Ende der Leitung genau 3–3,3 V herauskommen. Das Verhältnis der Spannung von 3,3 V/5 V mit dem Wert 0,66 entspricht nahezu dem Verhältnis der beiden Widerstände 330 Ω/470 Ω mit dem Wert 0,7. In diesem Schaltungsbeispiel sind diese Widerstände mit 330 Ω (R1) und 470 Ω (R2) groß.

Bitte beachten Sie, dass es sich hierbei um eine grobe Überschlagsrechnung handelt. Diese reicht aus, da der Raspberry-Pi tolerant gegenüber Abweichungen von den idealen Signalpegeln ist. Falls Sie genauere Ergebnisse benötigen, sollten Sie stattdessen mit den Formeln für belastete bzw. unbelastete Spannungsteiler arbeiten.

Sofern Sie die genannten Widerstandswerte nicht zur Hand haben, können Sie Widerstände zwischen 300 Ω und 10 kΩ verwenden, das Verhältnis der Widerstände zueinander sollte jedoch bestmöglich beibehalten werden.

Abstandssensor – vom Datenblatt zum Python-Skript

Um den Abstandssensor in Betrieb zu nehmen, reicht ein einfaches Python-Skript aus. Erstellen Sie im ersten Schritt aus Übersichtlichkeitsgründen ein eigenes Projektverzeichnis – in diesem Beispiel wird dieses schlicht *distance* genannt.

```

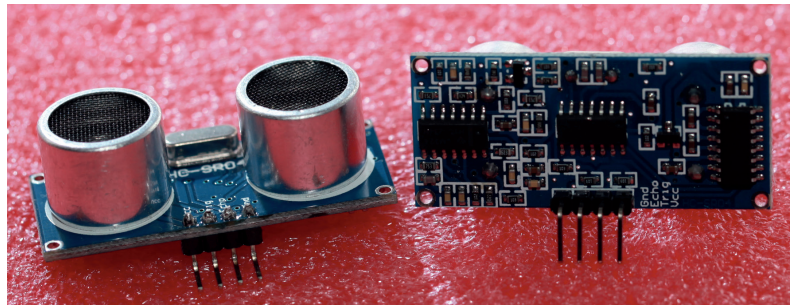
sudo -i
mkdir distance
cd distance
nano distance.py

```

In diesem Verzeichnis wird die Datei *distance.py* mit dem *nano*-Editor erzeugt, die Sie später mit Leben füllen. Doch zunächst machen Sie sich mit den Parametern und Hintergründen des Sensors vertraut, hier liefert das Datenblatt zu dem Sensor, das Sie im Internet unter der Bezeichnung *HC-SR04Users_Manual.pdf* über eine beliebige Suchmaschine finden, wertvolle Informationen. Für die Abstandsmessung wird über den TRIGGER-Eingang des HC-SR04 Moduls ein Ultraschallimpuls erzeugt.



Bild 1: Die Baugröße des HC-SR04-Moduls ist nahezu ideal – mit 45 x 20 x 15 mm lässt es sich bequem sowohl in der Sensorik als auch in der Robotik für viele Projekte verwenden.



Anschließend wird das reflektierte Signal über den Ausgang ECHO als Signalpuls signalisiert, dessen Impulsdauer proportional zur zurückgelegten Entfernung des Ultraschallimpulses ist.

Im Datenblatt wird eine Umrechnungsformel für die Impulsdauer des Abstandssensors angegeben, mit der sich der Abstand in Zentimetern bzw. in angelsächsischen Inches berechnen lässt. Hier wird die Impulsdauer (Time) in μs (Mikrosekunden) angegeben. Um nun die Distanz in Zentimetern zu berechnen, ist diese Impulsdauer durch den Wert 58 zu dividieren, im Falle der Maßeinheit *Inch* durch den Wert 148. Beachten Sie, dass es sich hier um die doppelte Entfernung handelt (Hin- und Rückweg des Impulssignals) – um den tatsächlichen Abstand zu errechnen, ist der Umrechnungsfaktor von $29,15 \mu\text{s}/\text{cm}$ der richtige.

- Distanz in Zentimetern = Impulsdauer / 58
- Distanz in Inches = Impulsdauer / 148

Hier wird also die Zeitspanne, zwischen Aussendung und Empfang des Schalls, bereits durch das Sensor-modul in eine Entfernung umgerechnet. Alternativ lässt sich der Abstand auch über die Schallgeschwindigkeit (ca. 340 m/s) berechnen, über den Sie ebenfalls auf den Umrechnungsfaktor von $29,15 \mu\text{s}/\text{cm}$ für die Impulsdauer kommen. Auch hier gibt diese die doppelte Entfernung zwischen Sensor und Gegenstand in cm an, die Sie noch halbieren müssen, um die tatsächliche Entfernung zwischen Sensor und Gegenstand/Wand zu errechnen.

Da die Schallgeschwindigkeit auch von der Raumtemperatur abhängt, ist diese in die Betrachtung mit einzubeziehen:

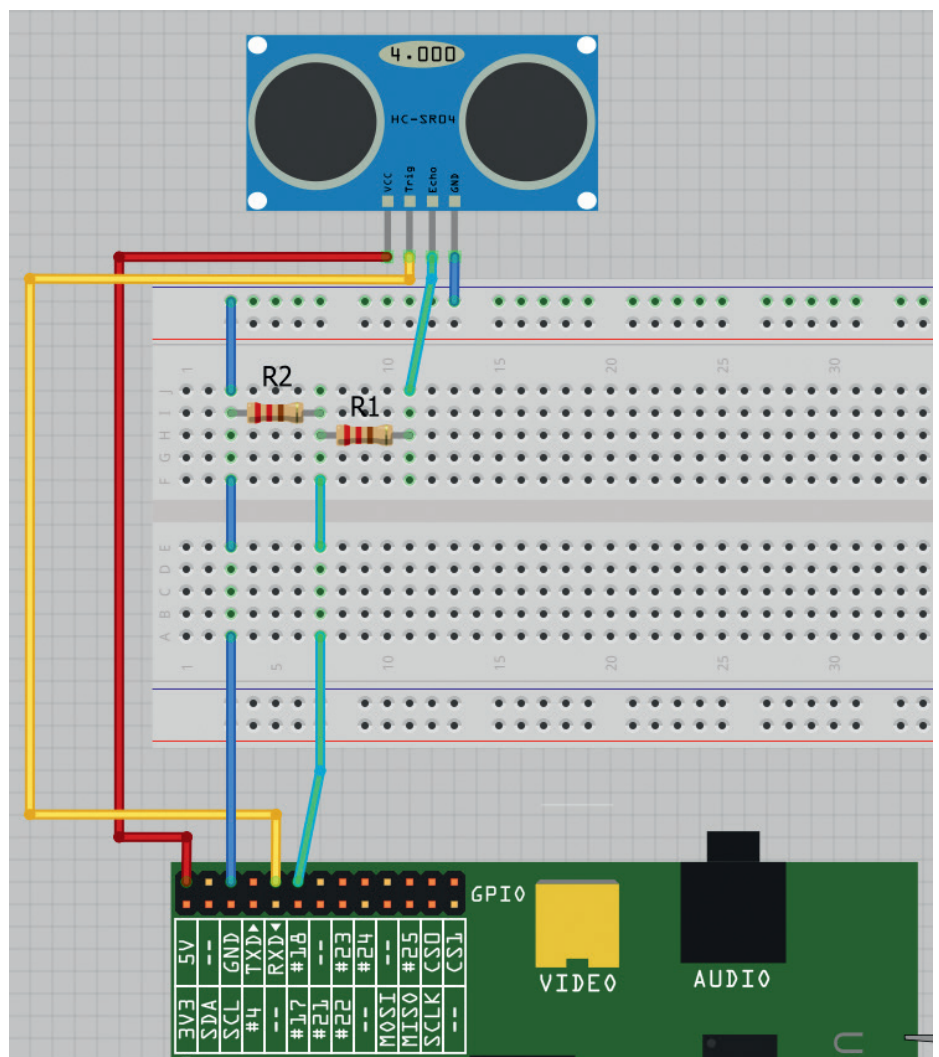


Bild 2: Schematische Anschlussdarstellung via Fritzing: Hier benötigen Sie bei der GPIO-Leiste des Raspberry Pi neben der 5-V- und Masse-Leitung zwei GPIO-Pins.



- Schallgeschwindigkeit = $331,5\text{m/s} * \sqrt{1 + \text{Temperatur}/273,15^\circ \text{C}}$

Bei einer angenommenen Raumtemperatur von 21 Grad entspricht dies:

- Schallgeschwindigkeit = $331,5\text{m/s} * 1,0377286903165997470678940342939 \text{ m/s}$
- Schallgeschwindigkeit = $344,00706083995281615300687236813 \text{ m/s}$

Dadurch ergibt sich die Formel für die Entfernung zwischen Sensor und Gegenstand/Wand:

- Abstand = $\frac{1}{2} * \text{Schallgeschwindigkeit} * \text{Zeit in } \mu\text{s}$
- Abstand = $\frac{1}{2} * 344,007 * \text{Zeit in } \mu\text{s}$

Damit können Sie die Impulsdauer in Sekunden mit 344 m/s beziehungsweise $34400,7 \text{ cm/s}$ multiplizieren und anschließend halbieren, damit man die Entfernung zwischen Sensor und Gegenstand erhält. Bei einem Sensor zu einem Preis von wenigen Euro bleibt die Messgenauigkeit auf der Strecke: Die Versuche zeigten, dass in der Praxis die Messwerte des Ultraschallsensors, zum Teil deutlich um mehr als einen Zentimeter, von der tatsächlichen gemessenen Distanz abweichen.

Die Unterschiede wurden zudem sowohl bei den direkt ausgelesenen Werten als auch bei den aus der Schalllaufzeit berechneten Werten sichtbar. Leider gibt es auch keine Empfehlung, ob nun der direkt vom Sensor gelieferte Wert oder der aus der obigen Schalllaufzeiten-Formel nachträglich berechnete Wert genauer ist. Um hier die Abweichungen besser in den Griff zu bekommen, wurde das Programm um eine Mittelwert-Funktion erweitert. In diesem Fall nimmt der Sensor in kurzen Intervallen drei Messungen vor, bildet daraus einen Mittelwert und übermittelt diesen an die aufzurufende Funktion zurück.

```
# -*- coding: utf-8 -*-
#!/usr/bin/python
# -----
# distance.py in der Schleife
import time
import RPi.GPIO as GPIO
# -----
GPIO.setmode(GPIO.BCM)
# -----
# Ultrasonic-Sensor
#
GPIO_TRIG = 23 # GPIO23 (Ausgang) / Pin 16 / wiring Pi: 4
GPIO_ECHO = 24 # GPIO24 (Eingang) / Pin 18 / wiring Pi: 5
# Ausgang und Eingang einstellen
GPIO.setup(GPIO_TRIG,GPIO.OUT) # Trigger
GPIO.setup(GPIO_ECHO,GPIO.IN) # Echo
# -----
def distanzmessung():
    GPIO.output(GPIO_TRIG, True)
    time.sleep(0.00001) # nach 10 Microsekunden auf Low
    GPIO.output(GPIO_TRIG, False)
    start = time.time()
    while GPIO.input(GPIO_ECHO)==0:
        start = time.time()
    while GPIO.input(GPIO_ECHO)==1:
        stop = time.time()
    elapsed = stop-start
    distance = (elapsed * 34400)/2
    # Schallgeschwindigkeit = 344,0070608m/s = 34400 in cm
    return distance
# -----
def distanzmessung_mittel():
    # Drei hintereinanderfolgende Messungen werden gemittelt (anz)
    anz=3
    dis=0
    sum=0
    for i in range(anz):
        dis=distanzmessung()
        time.sleep(0.15)
        sum=sum+dis
    dis = sum / anz
    return dis
# -----
#
```




```
# -----
# Start Python-Skript
# -----
print („[ultrasonic] Abstandsmessung“)
# GPIO Trigger setzen auf Low, Ausis
GPIO.output(GPIO_TRIG, False)
try:
    while True:
        distance = distanzmessung_mittel()
        print („[ultrasonic] Abstand : %.1f cm (Mittelwert)“ % distance)
        time.sleep(2) # Sekunden warten
except KeyboardInterrupt:
    # CTRL-C gedrueckt
    # Reset GPIO
    print („[ultrasonic] Messung abgebrochen.“)
    GPIO.cleanup() # Aufräumen !
# ----- EOF -----
```

Das Skript läuft in einer *while*-Schleife so lange, bis der Anwender die Verarbeitung mit der Tastenkombination `<STRG/CTRL><C>` unterbricht.

Infrarot-Abstandssensor im Einsatz

Der im folgenden Projekt eingesetzte Infrarot-Abstandssensor basiert auf den LM393-Chip. Dieser IR-Sensor bringt einen integrierten Infrarot-Sender und Empfänger mit, der den Abstand des Signals als Analogwert empfängt und anschließend per Digitalsignal darüber informiert, ob der Sensor angeschlagen hat oder nicht. Der Anschluss des Sensors an den Raspberry Pi gestaltet sich demnach recht einfach.

LM393 IR Abstandssensor-Pin	Raspberry-Pi-Bezeichnung	Raspberry-Pi-Pin	Wiring Pi
VCC	3,3 V	1	-
ECHO	GPIO18 (Eingang)	12	1
GND	Masse	6	-

In diesem Beispiel wurde auf eine Pull-up-Schaltung für den Digital-Eingang verzichtet. Neben der Spannungsversorgung von 3,3 V und Masse ist lediglich ein GPIO-Pin notwendig, der später als Eingang definiert werden muss, damit das Messergebnis dargestellt oder weiterverarbeitet werden kann.

Infrarot-Abstandssensor mit Python

Das Skript für den Infrarot-Abstandssensor ist selbst für Einsteiger eine leichte Übung. Hier legen Sie zunächst ein Projekt-Verzeichnis (hier: *lm393-ir*) an und erzeugen dort eine Datei mit der *py*-Dateierweiterung.

```
mkdir lm393-ir
cd lm393-ir
nano lm393-step1.py
```

In diesem wird in der Datei *lm393-step1.py* zunächst die *RPi.GPIO*-Bibliothek eingebunden und mit dem *GPIO_IR* eine Variable für den Pin 12/GPIO18 festgelegt und dieser als Eingang definiert.

```
# -*- coding: utf-8 -*-
#!/usr/bin/python
# -----
#
# lm393-step1.py
# -----
import time
import os
import RPi.GPIO as GPIO
# -----
GPIO.setmode(GPIO.BCM) # GPIO Mode
# -----
GPIO_IR = 18 # GPIO18 (Eingang) / Pin 12 / wiring Pi: 1
GPIO.setup(GPIO_IR, GPIO.IN)
# -----
# Start Python-Skript
# -----
delay = 2
# Dauer-Schleife
os.system(„clear“)
```



```

try:
    print(„[IR-Sensor] Sensor aktiv...“)
    while True:
        if(GPIO.input(GPIO_IR)):
            print(„OK!“)
        else:
            print(„SENSOR AUSGELÖST!“)
            time.sleep(delay)
except KeyboardInterrupt:
    # CTRL-C gedrueckt Reset GPIO
    print(„[IR-Sensor] Sensorprüfung.“)
GPIO.cleanup()
# ----- EOF -----

```

Durch die Dauerschleife mit *while* wird permanent der Status abgefragt – erkennt der Sensor ein Signal bzw. eine Unterbrechung des ausgesendeten IR-Signals, schlägt das Python-Skript Alarm (Bild 3).

Im nächsten Projekt gehen Sie einen Schritt weiter – hier koppeln Sie den vorliegenden IR-Abstandssensor mit einem Ultraschallsensor auf der Steckboard-Schaltung. Anschließend erweitern Sie den Python-Code und bauen die notwendige Logik für den Ultraschallsensor ein.

Ultraschall- und IR-Abstandssensoren kombinieren

Um nun herauszufinden, ab wann genau der IR-Abstandssensor Alarm schlägt, können Sie den Ultrasonic-Sensor HC-SR04 verwenden, der Ihnen den Abstand der Fläche laut Datenblatt in einem Bereich von 2 bis 400 cm anzeigt. Da es in diesem Projekt nicht darauf ankommt, den Abstand permanent zu melden, sondern nur dann, wenn auch der IR-Abstandssensor aktiv wird, wird die Logik der vorherigen Projektdatei *lm393-step1.py* dahingehend verändert, dass die beiden notwendigen Funktionen für die Abstandsmessung *distanzmessung()* und *distanzmessung_mittel()* nur aktiviert werden, wenn die *if not(GPIO.input(GPIO_IR))-Bedingung* erfüllt wird. Die für den Betrieb des Ultraschallsensors HC-SR04 ergänzte Datei kann nach den Anpassungen im Projektverzeichnis unter der Bezeichnung *lm393-step2.py* abgespeichert werden.

```

# -*- coding: utf-8 -*-
#!/usr/bin/python
# -----
# ultra-sonic und IR-Sensor im Einsatz
# lm393-step2.py
# -----
import time
import os
import RPi.GPIO as GPIO
# -----
GPIO.setmode(GPIO.BCM) # GPIO Mode
# -----
GPIO_IR = 18 # GPIO18 (Eingang) / Pin 12 / wiring Pi: 1
GPIO.setup(GPIO_IR, GPIO.IN)
# -----
# Ultrasonic-Sensor
#
GPIO_TRIG = 23 # GPIO23 (Ausgang) / Pin 16 / wiring Pi: 4
GPIO_ECHO = 24 # GPIO24 (Eingang) / Pin 18 / wiring Pi: 5
# Ausgang und Eingang einstellen
GPIO.setup(GPIO_TRIG,GPIO.OUT) # Trigger
GPIO.setup(GPIO_ECHO,GPIO.IN) # Echo
# -----
def distanzmessung():
    GPIO.output(GPIO_TRIG, True)
    time.sleep(0.00001) # nach 10 Nanosekunden auf Low
    GPIO.output(GPIO_TRIG, False)
    start = time.time()
    while GPIO.input(GPIO_ECHO)==0:
        start = time.time()
    while GPIO.input(GPIO_ECHO)==1:
        stop = time.time()
    elapsed = stop-start
    distance = (elapsed * 34400)/2

```

```

pi@raspiBreakout: ~/lm393-ir
[IR-Sensor] Sensor aktiv...
SENSOR AUSGELÖST!
SENSOR AUSGELÖST!
OK!
OK!
OK!
SENSOR AUSGELÖST!
SENSOR AUSGELÖST!
SENSOR AUSGELÖST!
SENSOR AUSGELÖST!
SENSOR AUSGELÖST!
OK!
OK!

```

Bild 3: Steht der Sensor bereit, dann wird dies über OK!-Meldung dargestellt. Bei einem gewissen Abstand zu einem Objekt wird jedoch der IR-Sensor aktiv und weist mit dem Hinweis SENSOR AUSGELÖST! darauf hin.



```

# Schallgeschwindigkeit = 344,0070608m/s = 34400 in cm
return distance
# -----
def distanzmessung_mittel():
    # Drei hintereinanderfolgende Messungen werden gemittelt (anz)
    anz=3
    dis=0
    sum=0
    for i in range(anz):
        dis=distanzmessung()
        print("Wert ,, i+1 ,,"Abstand = ,, dis)
        time.sleep(0.15)
        sum=(sum+dis)
    print "Summe =", sum
    dis = sum / anz
    return dis
# -----
# Start Python-Skript
# -----
delay = 2
# Dauer-Schleife
os.system("clear")
try:
    print("[IR-Sensor] Sensor aktiv...")
    while True:
        if not(GPIO.input(GPIO_IR)):
            print("SENSOR AUSGELÖST!")
            distance = distanzmessung_mittel()
            print("[ultrasonic] Abstand : %.1f cm (Mittelwert)" % distance)
            time.sleep(delay)
except KeyboardInterrupt:
    # CTRL-C gedrueckt Reset GPIO
    print("[IR-Sensor] Sensorprüfung.")
    GPIO.cleanup()
# ----- EOF -----

```

Da der Infrarotsensor *LOW*-aktiv ist, also bei einer fallenden Flanke anschlägt, erscheint in diesem Fall die genaue (gemittelte) Abstandsmessung, die anschließend als Wert auf dem Bildschirm ausgegeben wird. Das obige Skript wird wie gewohnt per `sudo python lm393-step2.py` gestartet – die Abstandsmessung schlägt zur Laufzeit nur dann an, wenn das Hindernis bzw. das Objekt sich in Reichweite des IR-Sensors befindet ([Bild 4](#)).

Neben dem vorgestellten IR-Sensor gibt es zig weitere IR-Sensoren samt IR-Technik am Markt, die sich auch für den Raspberry Pi verwenden lassen. Je nach Anwendungszweck und gewünschter Genauigkeit ist der Sensor auszuwählen.

Freie Auswahl – Sharp-Abstandssensor

Die Firma Sharp hat in Sachen Abstandssensoren ein breit gefächertes Sortiment, was sich vorwiegend hinsichtlich des Messbereichs sowie der Datenschnittstelle unterscheidet. Für den Anschluss an den Raspberry Pi ist die bereits vorgestellte MCP3008-Schnittstelle für die vorgestellten Sharp-Sensoren das Mittel der Wahl – nachfolgend wird der Sharp-GP2D120-Sensor exemplarisch in einem Projekt verwendet. Zunächst schließen Sie diesen gemäß der in [Bild 5](#) dargestellten Anschlussbelegung am MCP3008-IC an.

Die Anschlussbelegung des MCP3008 sowie die dazu verbundenen Pins des Raspberry Pi sowie des Sharp-Abstandssensor sind in der nachstehenden Tabelle zusammengefasst.

MCP3008-Pin	MCP3008	Sharp-GP2D120-Sensor	Raspberry-Pi-Bezeichnung	Raspberry-Pi-Pin	Wiring Pi
1	CHO	1	–	–	–
–	–	3	5 V	2	–
15	VREF	–	3,3 V	1	–
16	VDD	–	3,3 V	1	–
11	DIN	–	GPIO10 / SPI MOSI	19	12
12	DOUT	–	GPIO / SPI MISO	21	13
13	CLK	–	GPIO 11 / SPIO_SCLK	23	14
10	CS/SHDN	–	GPIO8 / SPIO_CEO_N	24	10
9	DGND	–	Masse	6	–
14	AGND	2	Masse	6	–



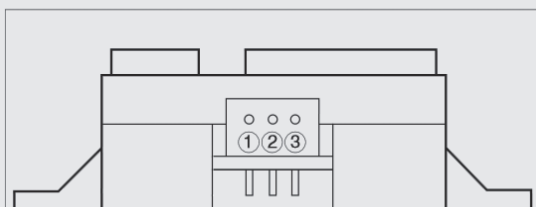
```

pi@raspiBreakout: ~/lm393-ir
[IR-Sensor] Sensor aktiv...
SENSOR AUSGELÖST!
Wert 1 Abstand = 11.4207267761
Summe = 11.4207267761
Wert 2 Abstand = 11.5601539612
Summe = 22.9808807373
Wert 3 Abstand = 12.0727539062
Summe = 35.0536346436
[ultrasonic] Abstand : 11.7 cm (Mittelwert)
SENSOR AUSGELÖST!
Wert 1 Abstand = 20.8976745605
Summe = 20.8976745605
Wert 2 Abstand = 21.4143753052
Summe = 42.3120498657
Wert 3 Abstand = 20.5737113953
Summe = 62.885761261
[ultrasonic] Abstand : 21.0 cm (Mittelwert)
SENSOR AUSGELÖST!
Wert 1 Abstand = 23.7518310547
Summe = 23.7518310547
Wert 2 Abstand = 24.1455078125
Summe = 47.8973388672
Wert 3 Abstand = 24.2193222046
Summe = 72.1166610718
[ultrasonic] Abstand : 24.0 cm (Mittelwert)
SENSOR AUSGELÖST!
Wert 1 Abstand = 23.1490135193
Summe = 23.1490135193
Wert 2 Abstand = 23.0834007263
Summe = 46.2324142456
Wert 3 Abstand = 23.3540534973
Summe = 69.5864677429
[ultrasonic] Abstand : 23.2 cm (Mittelwert)
SENSOR AUSGELÖST!
Wert 1 Abstand = 20.8648681641
Summe = 20.8648681641
Wert 2 Abstand = 20.7746505737
Summe = 41.6395187378
Wert 3 Abstand = 20.3645706177
Summe = 62.0040893555
[ultrasonic] Abstand : 20.7 cm (Mittelwert)

```

Bild 4: Ultraschall- und IR-Sensor im Zusammenspiel: Nur im Falle des aktivierten IR-Sensors liefert der Ultraschallsensor einen Messwert.

GP2D120 Optoelectronic Device



PIN	SIGNAL NAME
①	V _O
②	GND
③	V _{CC}

GP2D120-B

Figure 1. Pinout

Bild 5: In der Mitte die Masse, anhand der Seitenansicht lassen sich die Pins für die Versorgungsspannung sowie die Ausgangsspannung sicher zuordnen.

Nach dem Anschluss des Sensors an der MCP3008-Schaltung und die Inbetriebnahme am Raspberry Pi steuern Sie den Sensor nun mit einem Python-Skript an. Voraussetzung ist hier, wie im dritten Teil der Artikelreihe im Abschnitt *SPI-Nutzung ohne Umwege* beschrieben, die Installation der SPIDEV-Moduls, um den einfachen Zugriff auf den SPI-Bus sicherzustellen.

Messwertbestimmung der Sharp-Abstandssensoren

Jeder Abstandssensor hat seinen bestimmten Anwendungszweck: Je nach Genauigkeit und Geldbeutel sowie je nach Robotik- und Navigationsanwendung wählen Sie den passenden Sensor aus. Für die SOHO-Einsatz zum Basteln reichen die kostengünstigen Sharp-Sensoren völlig aus, falls es nicht auf eine millimetergenaue Messung ankommt. Je nach Modell haben die Sharp-Sensoren unterschiedliche Messbereiche – hier können Sie nach dem gewünschten Anwendungszweck gezielt einkaufen.

Sharp-Sensor	Messbereich
GP2D120XJ00F	4–30 cm
GP2D120	6–20 cm
GP2D12	10–80 cm
GP2Y0A02YK	20–150 cm



Je nach verwendetem Sensor können die Anschlüsse für Masse, Spannungseingang sowie Datenausgabe anders belegt sein – auch Werte können hier je nach Modul etwas differieren, da der Messbereich der Sensoren unterschiedlich ist. So ist der Sharp GP2Y0A02YK für einen Messbereich von 20–150 cm spezifiziert, während das Modell Sharp GP2D120 für Entfernungen von 4–20 cm und der Sharp GP2D12 von 10–80 cm geeignet ist. Für den Einsatz der Sensoren in der Praxis bedeutet dies, dass der Einsatzzweck darüber entscheidet, ob ein oder mehrere Sensoren benötigt werden.

Im Falle des GP2D120-Sensors liefert der Hersteller in den Abbildungen 4 und 5 des Datenblatts den technischen Zusammenhang zwischen Distanz und gemessener Spannung in zwei Diagrammen. Da die direkte Zuordnung von Spannung zur Distanz bauteilbedingt nicht linear verläuft, wurde eine Anpassung der Wertepaarzuordnung gewählt, die eine Linearisierung im gültigen Messbereich ermöglicht. In der Fußnote von Abbildung 5 ist die folgende Formel zu finden.

$$X(\text{Spannung}) = 1 / (\text{Distanz} + 0.42) \text{ (1/cm)}$$

Die Konstante K mit dem Wert 0,42 ist ein vom Hersteller vorgeschlagener Wert, der in erster Näherung für eine gute Linearisierung im gültigen Messbereich sorgt.

Da anzunehmen ist, dass der Wert X also von der Distanz abhängt, wird im nächsten Schritt diese Kurve aus $X = 1 / (\text{Distanz} + 0.42)$ mittels Regressionsanalyse in eine Formel in der Form $y = mx + c$ gebracht. Somit werden nun die Steigung m und der konstante Ausdruck c benötigt.

Durch die Anpassung der Wertepaarzuordnung ergibt sich, dass sich der konstante Ausdruck c zu Null auflöst. Der Wert für die Steigung m lässt sich einfach über zwei Wertepaare mit der folgenden Formel bestimmen.

$$m = (V2 - V1) / (X2 - X1)$$

Hierbei sind die beiden Variablen $V1$ und $V2$ die berechneten Spannungen aus den gemessenen Werten des MCP3008-ICs zu zwei definierten Abständen D . Nun nehmen Sie einfach einen Zollstock mit einem Stück Karton und messen die Spannung beim minimalen Abstand (beispielsweise bei 6 cm) und beim maximalen Abstand (beispielsweise bei 20 cm). $V1$ und $V2$ werden in dem vorliegenden Code-Beispiel von der Funktion `def convertsVolts(data,places)` geliefert.

In unserem Beispiel ergeben sich daraus die folgenden Werte.

$$X1 = 1 / (6 + 0,42) = 0,156$$

$$X2 = 1 / (20 + 0,42) = 0,049$$

$$m = (2,025 - 0,650) / (0,156 - 0,049) = 12,85$$

Aus den so erhaltenden Daten kann nun die oben angegebene Gleichung komplettiert werden.

$$\text{Spannung} = m * (1 / (\text{Distanz} + 0.42)) + c$$

$$\text{Spannung} = 12,85 * (1 / (\text{Distanz} + 0.42)) + 0$$

Formen Sie die Gleichung um, dann erhalten Sie die Distanz in Abhängigkeit von der anliegenden Spannung.

$$\text{Distanz} = (12,85 / \text{Spannung}) - 0,42$$

Um eine genauere Anpassung der Werte auf Ihren eigenen Sensor zu ermöglichen, können Sie später selbst eigene Messreihen aufnehmen und so z. B. einen anderen Wert für die vom Hersteller vorgeschlagene Konstante K von 0,42 verwenden.

Diese Formel überführen Sie in das Python-Skript für die Berechnung der Distanz in cm.

$$\text{cm} = (12,85 / v) - 0.42$$

Nachstehend sind die wichtigsten Funktionen für den Sensoreinsatz dargestellt.

```
def getadcChannel(channel): # Anschluss des Sensors
    # Nur Kanal 0-7 erlaubt
    if ((channel > 7) or (channel < 0)):
        return -1
    rdata = spi.xfer2([1,(8+channel)<<4,0])
    data = ((rdata[1]&3) << 8) + rdata[2]
    return data
```

Die Funktion `getadcChannel` erwartet die Angabe des Kanals des MCP3008-ICs und liefert den dort zugehörigen Registerwert zurück.

```
def convertVolts(data,places):
    volts = (data * 3.3) / 1023 # 3.3V Referenzspannung!
    volts = round(volts,places) # und runden
    return volts
```

Die Funktion `convertVolts` verwendet den Registerwert der `getadcChannel`-Funktion und errechnet daraus die Spannung am anliegenden Anschluss. Anhand der Spannung kann im letzten Schritt der Abstand zum Objekt mit der Funktion `convert_cm_GP2D12` in `cm` bestimmt werden.

```
def convert_cm_GP2D12(v):
    if v == 0:
        return -2
    cm = (12,85 / v) - 0.42
    if(cm > 80):
        return -1
    return cm
```



Für andere Sensoren wie z. B. den Sharp GP2Y0A21 ist die Herangehensweise dieselbe. Wer sich die Rechenarbeit abkürzen möchte, der kann hier natürlich auch auf Fertigrezepte zurückgreifen. So gibt es beispielsweise die Produktseite www.acroname.com/articles/linearizing-sharp-ranger.html, auf der die Herleitung der Formeln ebenfalls erläutert sind. Dort ist auch die Formel für den Sensor Sharp GP2Y0A21 zu erhalten.

Zusammenfassend hängt es natürlich vom Anwendungszweck ab, welcher Sensor zum Einsatz kommt. Auch wenn mehrere zusammen verwendet werden, sollte dies nun kein Problem mehr darstellen, da die Sharp-Sensoren nur Stromversorgung und Ausgangs-Signal besitzen.

```
# -*- coding: utf-8 -*-
#!/usr/bin/python
# -----
# Das Skript nutzt die Analog-Eingaenge des MCP3008 IC
# und liest diese über SPI Bus aus
# Sensor : Sharp GP2D120
# Datei messung-sharp-step1.py
# -----
import spidev
# wget https://raw.githubusercontent.com/doceme/py-spidev/master/setup.py
# wget https://raw.githubusercontent.com/doceme/py-spidev/master/spidev_module.c
import time
import os
# SPI Objekt erstellen
spi = spidev.SpiDev()
spi.open(0,0)
# Funktion um Daten aus dem MCP3008 IC zu lesen
# Kanalaehlung von 0-7
def getadcChannel(channel):
    # Nur Kanal 0-7 erlaubt
    if ((channel > 7) or (channel < 0)):
        return -1
    # Start Bit senden
    # Send start bit, sgl/diff, odd/sign, MSBF
    # channel = 0 sends 0000 0001 1000 0000 0000 0000
    # channel = 1 sends 0000 0001 1100 0000 0000 0000
    # sgl/diff = 1; odd/sign = channel; MSBF = 0
    rdata = spi.xfer2([1,(8+channel)<<4,0])
    data = ((rdata[1]&3) << 8) + rdata[2]
    return data
# Funktion um Werte in Spannungswerte zu konvertieren
# (auf 2^10 Einheiten)
def convertVolts(data,places):
    volts = (data * 3.3) / 1023 # 3.3V Referenzspannung!
    volts = round(volts,places) # und runden
    return volts

def convert_cm_GP2D12(v):
    if(v == 0):
        return -2
    cm = (1.0 / (v / 13.17)) - 0.35
    if(cm > 80):
        return -1
    return cm
```



```

# -----
# Start Python-Skript
# -----
os.system("clear")
print("[Sharp-Infrarotsensor] Abstandsmessung...")
# Kanalnummer auswaehlen
sharp_channel = 0
# -----> max 7 !
#
# Pause zwischen Messungen
delay = 5 # Sekunden
# Anzahl der Dezimalstellen
places = 3
try:
    # Dauer-Schleife
    while True:
        # Sharp Sensor
        tmp_level = []
        for i in range (0,10):
            tmp_level.append(getadcChannel(sharp_channel))
        sharp_level = sum(tmp_level)/10.0
        #sharp_level = getadcChannel(sharp_channel)
        sharp_volts = convertVolts(sharp_level, places)
        sharp_GP2D12_cm = convert_cm_GP2D12(sharp_volts)
        # Datenausgabe
        print("-----")
        print("Sharp-Sensor GP2D12-Abstand betraegt : {} cm | {} ({}V)".format(sharp_GP2D12_cm, sharp_level, sharp_volts))
        # und warten bis zur naechsten Messung
        time.sleep(delay)
except KeyboardInterrupt:
    # CTRL-C gedrueckt
    print("[Sharp-Infrarotsensor] Messung abgebrochen.")
# ----- EOF -----

```

Egal ob Sie mit dem Sensor nun einen kleinen Roboter, einen Türöffner, einen Seifenspender, eine Toilettenspülung, Alarmanlage oder gar eine Einparkhilfe für das Kfz nachrüsten – der Python-Code ist nahezu immer derselbe wie oben abgedruckt. Starten Sie das Programm *messung-sharp-step1.py*, dann liefert dieses den gemessenen Abstand. Wird kein Hindernis oder Objekt erkannt, dann ist der Messwert meist größer als 80 cm, was laut Spezifikation des Sensors ungültig wäre. Aus diesem Grund werden Messwerte jenseits der 80 cm abgefangen und mit dem Rückgabewert von -1 belegt (Bild 6).

Im nächsten Schritt können Sie den gemessenen Wert grafisch darstellen lassen – je nach angeschlossenen Ausgabegerät am Raspberry Pi kann dies auch als einfacher cm-Wert im Terminal erfolgen. Alternativ nutzen Sie ein am Raspberry Pi angeschlossenes Display und nutzen dafür einen eindimensionalen Balken für die Abstandsdarstellung. **ELV**

```

pi@raspiBreadboard: ~/sharpir
[Sharp-Infrarotsensor] Abstandsmessung...
-----
Sharp-GP2D12-Sensor-Abstand betraegt : -1 cm | 18.9 (0.061V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : 13.1095701126 cm | 302.8 (0.977V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : 12.4046071775 cm | 319.6 (1.031V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : 12.6312438302 cm | 314.1 (1.013V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : 11.5289521229 cm | 343.2 (1.107V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : 10.9569647463 cm | 360.4 (1.163V)
-----
Sharp-GP2D12-Sensor-Abstand betraegt : -1 cm | 9.0 (0.029V)

```

Bild 6: Messbereich überschritten: Liefert das Testprogramm den Rückgabewert -1, kann dieses in diesem Beispiel kein Hindernis feststellen.



openHAB 2 mit neuen Möglichkeiten Alexa steuert Homematic und MAX!

Seit dem erfolgreichen Erscheinen der Sprachsteuerung von Amazon Echo stellt sich die Frage: „Wie kann ich mit diesem System meine Hausautomation steuern, meine Heizungssteuerung übernehmen oder meine smarten Haushaltsgeräte aus der Unterhaltungstechnik per Sprache bedienen?“ openHAB 2 heißt die Lösung, die alle smarten Geräte unter einer Oberfläche zusammenfasst und mit Amazon Echo verknüpft.

Kein Skill für Homematic & Co.?

Eine direkte Bedienung über die Spracheingabe von Amazon Echo setzt leider immer voraus, dass hierzu ein passender Befehlssatz, ein sogenannter Skill, für das jeweilige System zur Verfügung steht, welchen der Hersteller in Lizenz mit Amazon bereitstellt. Nicht jeder Hersteller entwickelt einen solchen Skill, sodass viele smarte Systeme „unverknüpfbar“ bleiben.

Durch eine Open-Source-Lösung namens „openHAB 2“ wurde genau diese Lücke geschlossen: Smarte Geräte werden unter einer Oberfläche

zusammengefasst und mit bestehenden Skills des Echo-Systems verknüpft.

Neben der Kommunikation mit Homematic stehen noch viele weitere Systeme in der Kompatibilitätsliste von openHAB 2, beispielhaft seien hier MAX!-Heizungsregelung, MiLight-Lichtsteuerung sowie DENON-AVR-Musiksteuerung genannt.

Was wird benötigt, um eine Sprachsteuerung mit Alexa zu realisieren?

In den Anfängen von openHAB stellte sich die Installation sehr aufwendig dar. Da dieses System aber mit einem offenen Quellcode betrieben wird, was anderen Entwicklern ermöglicht, aktiv an der Entwicklung von openHAB mitzuwirken, stehen heute bereits vorgefertigte Image-Dateien für den Raspberry Pi 2 und Raspberry Pi 3 kostenlos zur Verfügung. Daher konzentrieren wir uns in diesem Beitrag auf die Installation und Einrichtung auf einem Raspberry Pi 3 mithilfe eines Windows-10-Betriebssystems.

Neben dem Raspberry Pi 3, der microSD-Speicherkarte, einem Netzteil und ggf. einem Gehäuse wird lediglich ein Netzkabel als Verbindung zum Rou-

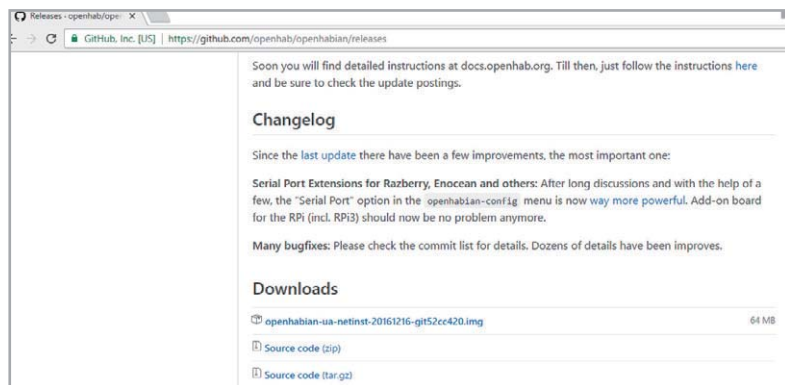


Bild 1

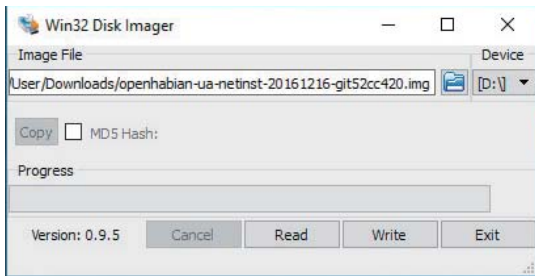


Bild 2

ter benötigt. Des Weiteren wird eine Software benötigt, um die heruntergeladene Image-Datei auf die Speicherkarte zu übertragen, und eine Software, mit welcher es möglich ist, Dateien komfortabel im Linux-System via SSH zu bearbeiten. Daher fällt die Softwareauswahl auf das Tool „Win32 Disk Imager“ [1] für das Kopieren des Images und das Tool „WinSCP“ [2], um ein SSH-Programm in Verbindung mit einem Linux-Dateimanager zu kombinieren. Eine einfache Alternative bietet hier auch das Tool „Putty“ [3].

Ein Download des bereits installationsfertigen openHAB-Images [4] für den Raspberry, das sogenannte openHABian, vervollständigt die Vorbereitungen.

Die Installation

Nach dem Herunterladen des openHABian Images (Bild 1) wird dieses über den Win32-Disk-Imager (Bild 2) auf die Speicherkarte kopiert und diese in den Raspberry Pi eingesteckt. Grundsätzlich wird keine Tastatur, Maus oder Monitor benötigt. Sollte die Möglichkeit bestehen, einen Monitor anzuschließen, kann man diesen zur Veranschaulichung des Installationsablaufs an den Raspberry Pi anschließen.

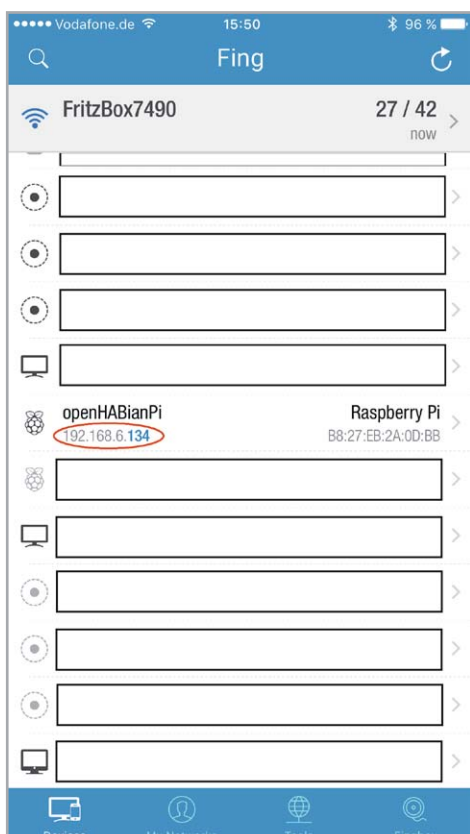


Bild 4

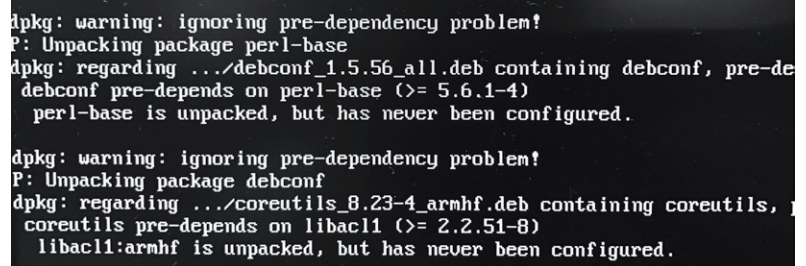


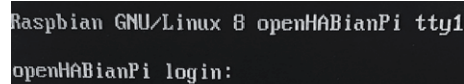
Bild 3

Was natürlich bei der Installation nicht fehlen darf, ist das Anschließen des Netzwerkabels, da der Raspberry während der Installation viele Installationsdaten aus dem Internet beziehen muss.

Sind die bisher genannten Arbeitsschritte abgeschlossen, erfolgt nun das Zuschalten der Stromversorgung. Der automatische Installationsablauf dauert ca. 30–60 min.

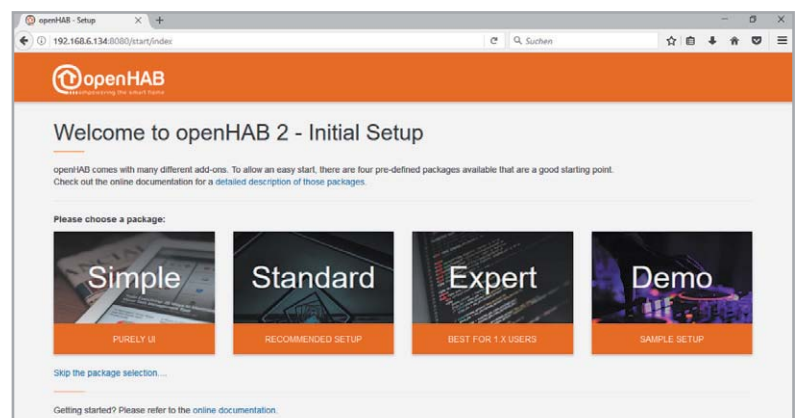
Während des Installationsablaufs darf der Raspberry Pi weder von der Stromversorgung getrennt noch anderweitig unterbrochen werden. Sollten während der Installation Warnungen wie „dpkg: warning: ignoring pre-dependency problem!“ (Bild 3) erscheinen, kann man diese ignorieren.

Nach Abschluss der Installation erscheint der folgende Bildschirm:

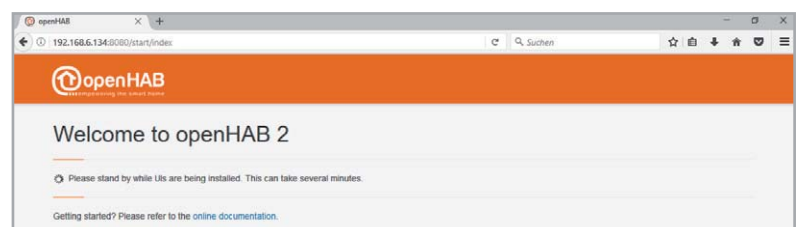


Nun erfolgt über das kostenlose Smartphone-Tool „FING“ (Bild 4) die Suche nach dem Raspberry Pi im Netzwerk. Dieser taucht hier mit dem Gerätenamen „openHABianPi“ mitsamt seiner IP-Adresse auf. Alternativ kann die IP-Adresse des Raspberry Pi auch über den verwendeten Router ermittelt werden.

Durch die Eingabe der ermittelten IP-Adresse des Raspberry Pi in den Browser kann man auf das openHAB-System zugreifen. Wichtig: Der Port 8080 muss hinter die IP-Adresse geschrieben werden. Beispiel: 192.168.6.134:8080.



Bei den auswählbaren „Packages“ wählen wir das Standard-Package, da dieses ausreichende Möglichkeiten für dieses Anwendungsvorhaben bietet.





Hinzufügen von Homematic Komponenten

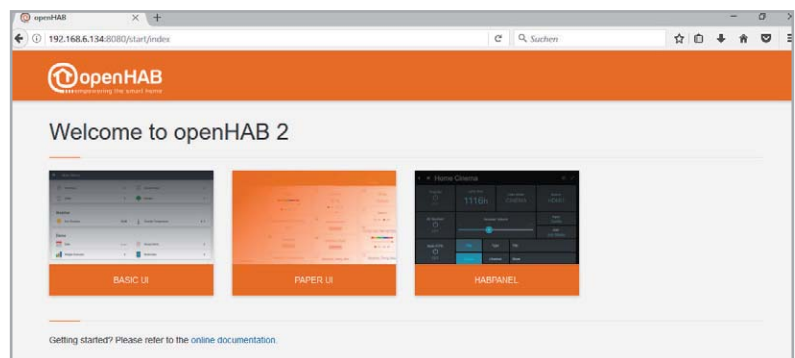
Ein wichtiger Zwischenschritt, bevor man anfangen kann, mit openHAB 2 zu arbeiten, ist die zuvor ermittelte IP-Adresse in die Firewall der CCU2 einzutragen.

Hierzu ist die WebUI der CCU2 zu öffnen, die Menüpunkte „Einstellungen → Systemsteuerung → Firewall konfigurieren“ anzuwählen und die IP-Adresse einzutragen. Hierbei muss man beachten, dass die IP-Adressen mit ; (Semikolon) getrennt werden. Hinter der letzten Adresse darf jedoch kein Semikolon stehen.

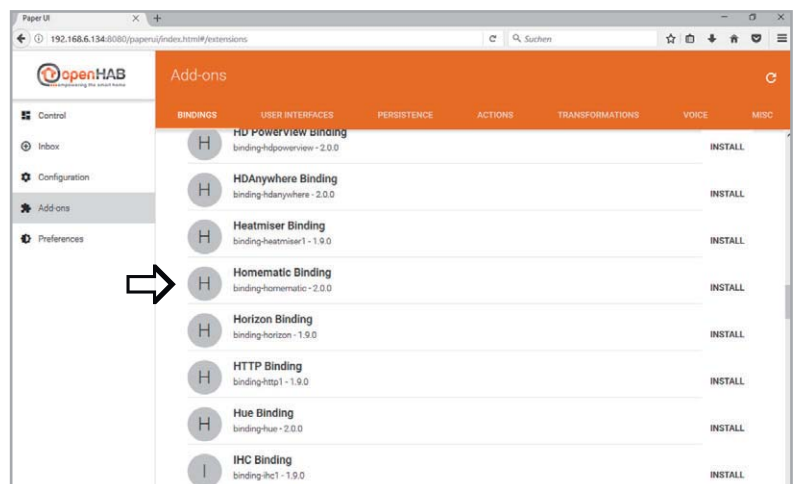


Nach erfolgter Eintragung der IP-Adresse in die Firewall der CCU2 geht es weiter mit der openHAB-2-Konfiguration.

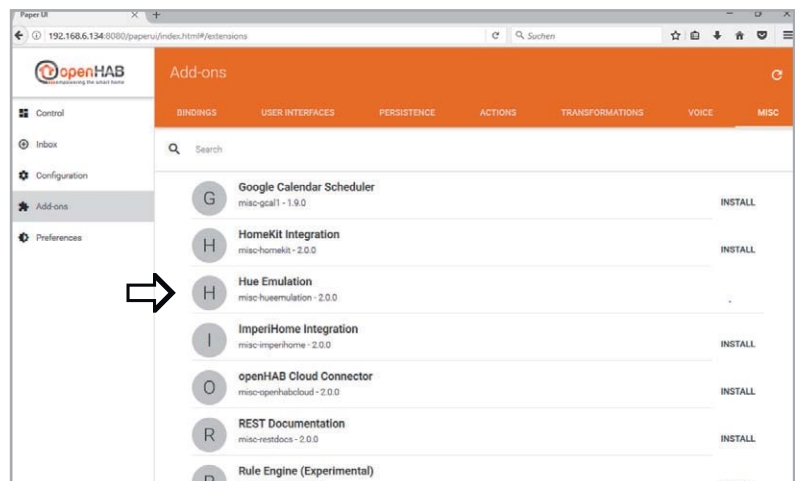
Jetzt kann man das gewünschte Oberflächendesign, z. B. die PAPER UI, auswählen.

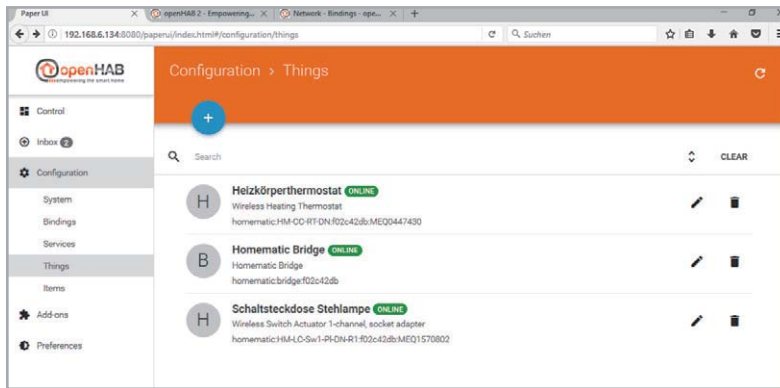


Auf der linken Seite erscheint nun das Navigationsmenü und auf der rechten Seite der jeweilige Inhalt. Um nun das sogenannte Binding, also die Verknüpfung des jeweiligen Smart Home Systems, ausführen zu können, erfolgt zunächst die Anwahl des Reiters „Add-ons“. Hier findet sich unter dem Reiter „Bindings“ auch das Homematic Binding, das man nun auswählt und installiert.

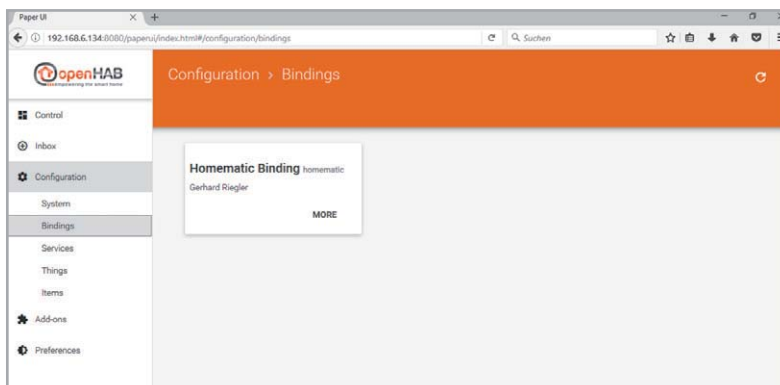


Ein weiteres wichtiges Add-on stellt die unter dem Reiter „MISC“ abgelegte „Hue Emulation“ dar. Diese wird final die Kommunikation mit Alexa ermöglichen.

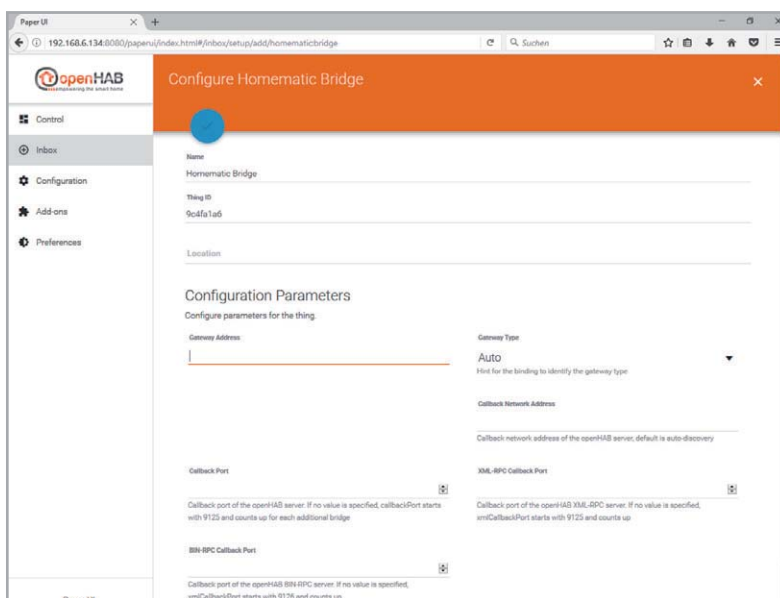




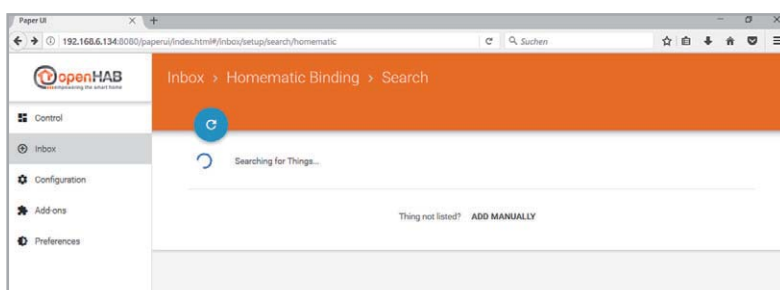
Nach der Installation der Add-ons müssen diese konfiguriert werden. Hierzu erfolgt die Auswahl des Punkts „Configuration – Things“ aus der Menüleiste.



Durch das Klicken auf den großen blauen Plus-Button kann man nun das passende Binding auswählen, über welches neue Geräte gesucht werden sollen, in diesem Fall das Homematic Binding.



In dieser Konfiguration findet die Software die CCU2 nicht automatisch, daher ist zunächst der Button „ADD MANUALLY“, dann das Homematic Binding anzuwählen, unter dem Menüpunkt „Gateway Address“ die IP-Adresse der CCU2 einzugeben und schließlich als „Gateway Type“ „CCU“ auszuwählen.



Nach der Eingabe aller Daten erfolgt wieder das Anwählen der Inbox und durch Klicken auf den „Scan for Things“-Button das Laden der Homematic Geräte in die openHAB-Konfiguration.

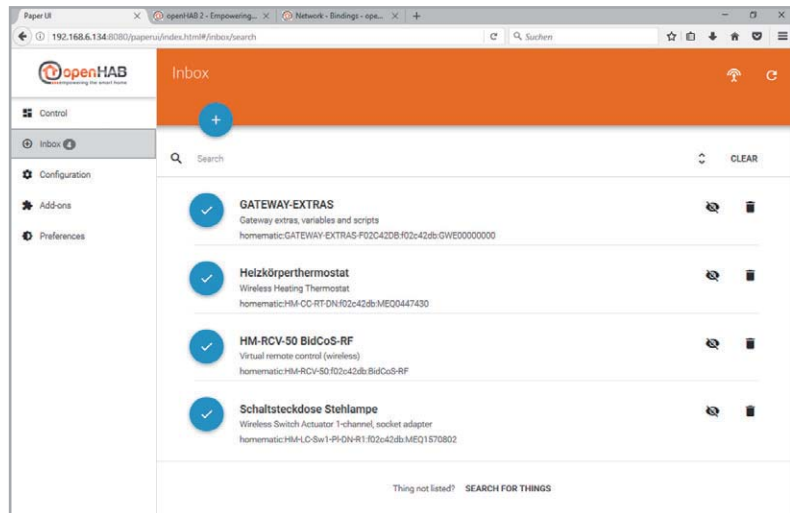


Um mögliche Geräte finden zu können, ist das openHAB System nun neu zu starten. Dazu öffnet man Putty und gibt die IP-Adresse ein. Der Benutzername sowie das Passwort lauten: **openhbian**

Nun erfolgt die Eingabe von:

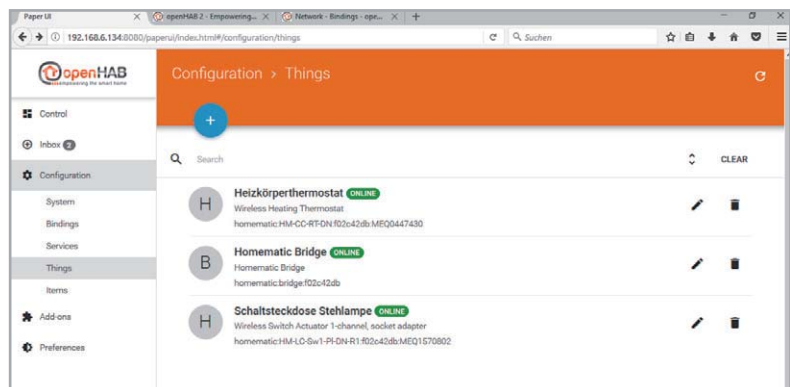
```
sudo reboot
```

Nach dem Neustart befinden sich die CCU2-Geräte jetzt in der Inbox.



Die Geräte, die verwendet werden sollen, können nun durch Klicken auf den blauen Haken in die openHAB-Konfiguration übernommen werden, im Beispiel sind dies die Geräte „Heizkörperthermostat“ und „Schaltsteckdose Stehlampe“.

Nach dem Hinzufügen der Geräte erscheinen diese in „Configuration – Things“.



Nun ist es erforderlich, Putty zu öffnen und eine „item-Datei“ zu erstellen. Hierzu loggt man sich wieder mit Putty auf dem Raspberry Pi ein:

```
cd /etc/openhab2/items
sudo nano alexa.items
```

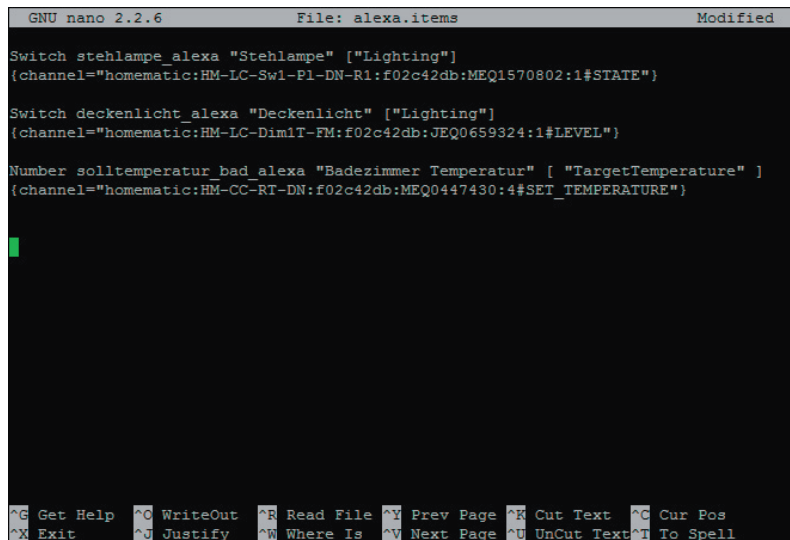
Durch den Einsatz des „sudo-Befehls“ erhält man die notwendigen Rechte, um diese Datei zu erstellen oder zu ändern. Möglicherweise wird hier nochmals das Passwort („openhbian“) abgefragt.

In diese Datei muss in unserem Beispiel folgender Inhalt geschrieben werden, die Tabelle unten erläutert den Inhalt:

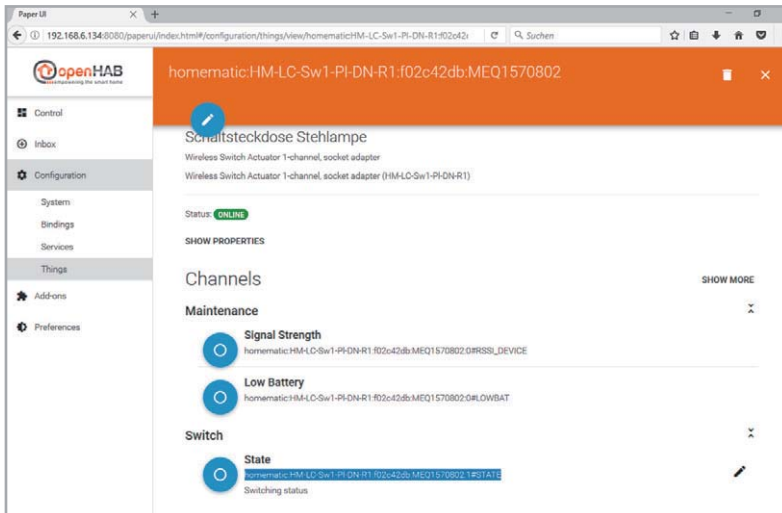
```
Switch stehlampe_alex "Stehlampe" ["Lighting"]
{channel="homematic:HM-LC-Sw1-Pl-DN-R1:
f02c42db:MEQ1570802:1#STATE"}

Switchdeckenlicht_alex "Deckenlicht" ["Lighting"]
{channel="homematic:HM-LC-Dim1T-FM:
f02c42db:JEQ0659324:1#LEVEL"}

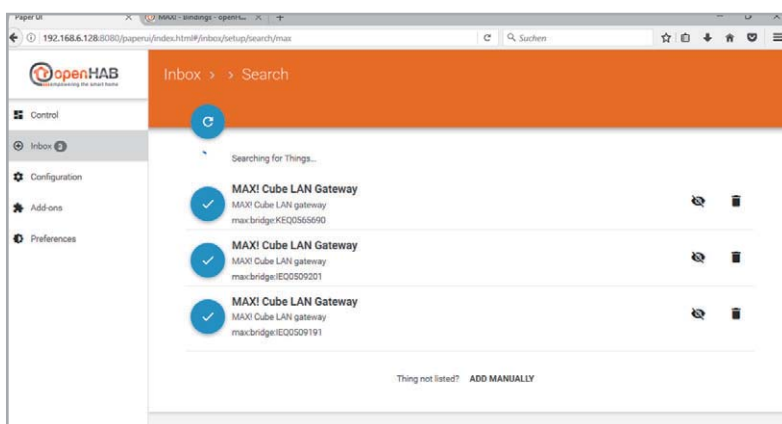
Number solltemperatur_bad_alex "Badezimmer Temperatur" [ "TargetTemperature" ]
{channel="homematic:HM-CC-RT-DN:
f02c42db:MEQ0447430:4#SET_TEMPERATURE"}
```



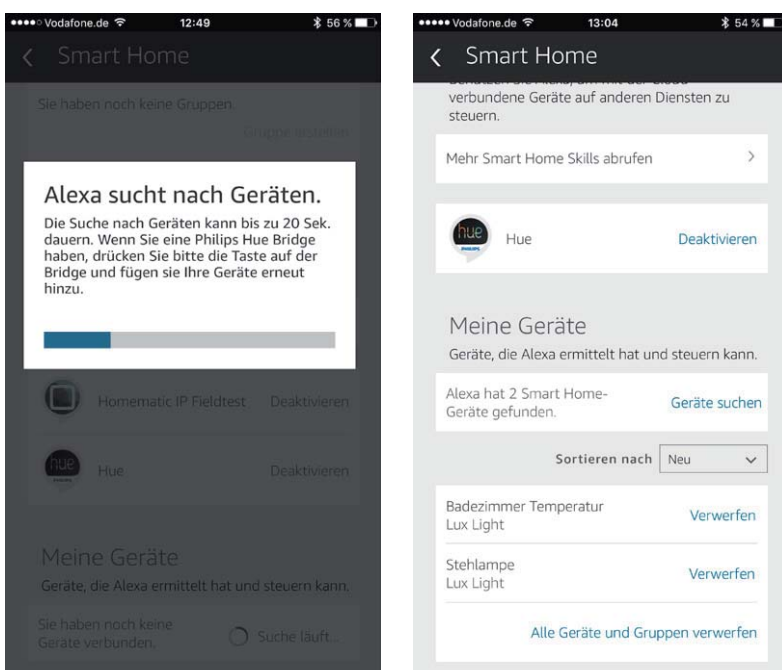
Funktion	Name der Funktion (intern)	Name (Sprachbefehl)	Zuordnung	Kanal, welcher etwas bewirken soll
Switch	stehlampe_alex	„Stehlampe“	[„Switchable“]	{channel="homematic:HM-LC-Sw1-Pl-DN-R1:f02c42db:MEQ1570802:1#STATE"}
Number	solltemperatur_bad_alex	„Badezimmer Temperatur“	[„TargetTemperaure“]	{channel="homematic:HM-CC-RT-DN:f02c42db:MEQ0447430:4#SET_TEMPERATURE"}
Dimmer	deckenlicht_alex	„Deckenlicht“	[„Lighting“]	{channel="homematic:HM-LC-Dim1T-FM:f02c42db:JEQ0659324:1#LEVEL"}



Einrichtung von MAX!-Komponenten



Einrichtung der Alexa-Smartphone-App



Die Gerätebezeichnung „homematic:HM-CC-RT-DN:f02c42db:MEQ0447430:4#SET_TEMPERATURE“ entnimmt man dem Menüpunkt „Things“.

Nach der Übernahme der Tabelle können die Änderungen mit der Tastenkombination „STRG + X“ übernommen werden.

Auf die Nachfrage, ob die Datei gespeichert werden soll, erfolgt die Antwort mit „Y“ für yes.

Alternativ gibt es die Möglichkeit, mit „MAX! Binding“ auf gleiche Weise MAX!-Geräte zu laden, welche innerhalb des Netzwerks mit einem MAX! Cube verbunden sind.

Um die bisher erstellte Sprachkonfiguration um das MAX!-System zu erweitern, fügt man der Konfigurationsdatei „alexa.items“ weitere Befehle hinzu:

```
Number maxplus_alex "Heizung Esszimmer"
["TargetTemperature"] {channel="max:thermostat:
JEQ0437598:MEQ0685513:set_temp"}
Number maxwandthermostat_alex "Heizung Wohn-
zimmer" [ "TargetTemperature" ] {channel="max:
wallthermostat:JEQ0437598:JEQ0167820:set_temp"}
```

Nach dem Hinzufügen dieser (natürlich auch wieder persönlich angepassten) Textzeilen wird mit der Tastenkombination „STRG + X“ der Editor wieder verlassen.

Damit ist die Installation und Einrichtung der Sprachsteuerung mit openHAB 2 abgeschlossen.

Nach dem Start der Alexa-App ist zuerst der Phillips-HUE-Skill zu installieren. Wenn der Skill installiert und aktiviert ist, wird in der App unter „Smart Home“ der Button „Geräte suchen“ betätigt ([Bild links](#)).

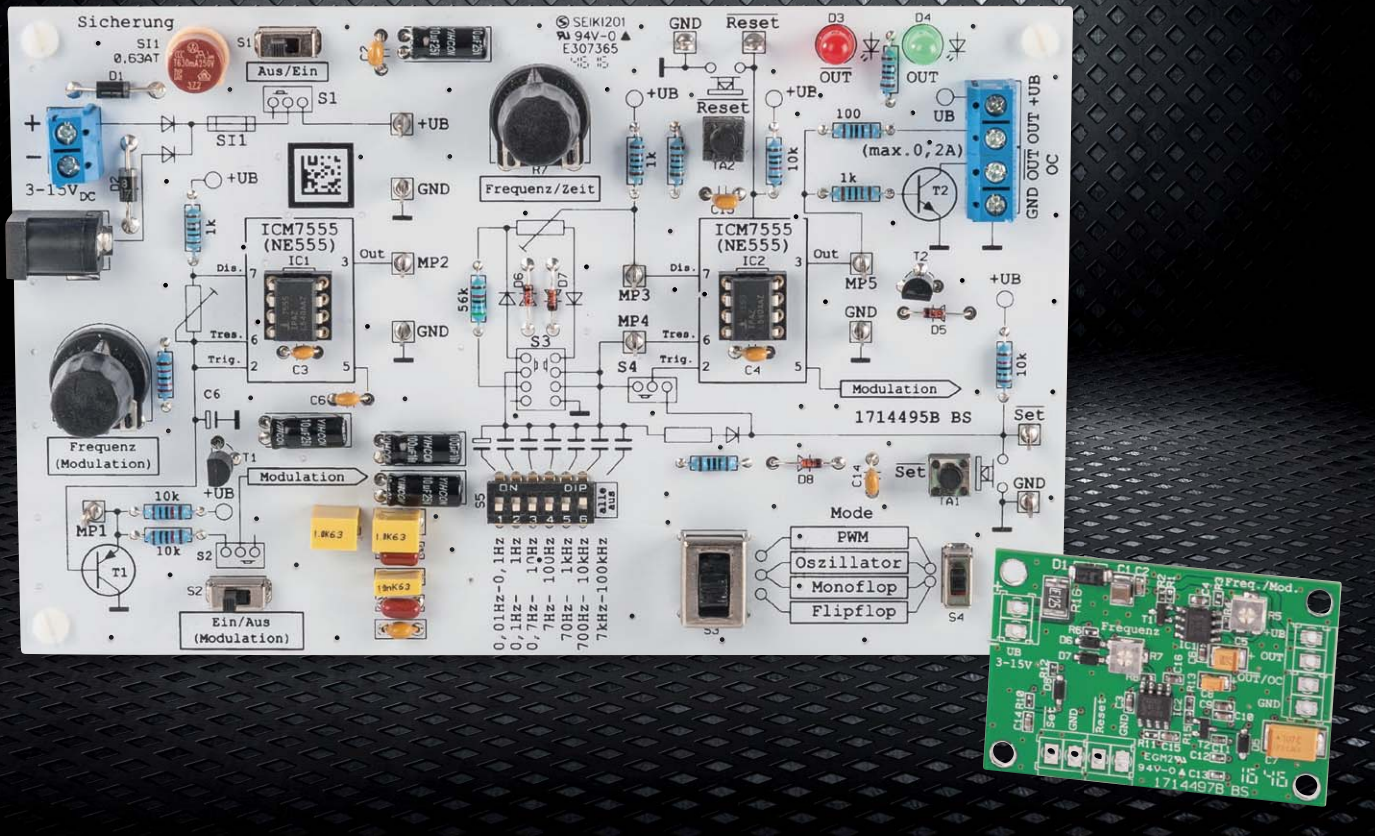
Nach Abschluss der Suche sollten nun die Geräte innerhalb der App angezeigt werden ([Bild rechts](#)).

Nun können die Geräte via Sprachbefehl von Amazon Echo angesteuert werden. **ELV**



Weitere Infos:

- [1] Win32 Disk Imager: <https://sourceforge.net/projects/win32diskimager/>
- [2] WinSCP: <https://winscp.net/eng/docs/lang:de>
- [3] Putty: www.putty.org/
- [4] openHABian Image: <https://github.com/openhab/openhabian/releases>



Elektronik von Grund auf

NE555-Experimentier- und -Anwenderboard

Infos zum Bausatz
im ELV Shop
#10055

Kaum ein aktives Bauteil der Elektronik hat solch eine Berühmtheit und eine so enorme Anwendungsbreite erzielt, und nur wenige Bauteile haben ein solch langes „Leben“ hinter sich. Der NE555 ist DER Klassiker der Elektronik – Grund genug, sich auch heute noch mit seinen Möglichkeiten zu beschäftigen und diesen äußerst vielseitigen Schaltkreis in eigenen Schaltungen einzusetzen. Genau dem dient die Kombination aus NE555-Experimentier- und -Anwenderboard.

NE555 – der Klassiker

Wer kennt ihn nicht, den Timerbaustein NE555? Dieser von Signetics im Jahr 1972 entwickelte Schaltkreis ist wohl jedem bekannt, der sich mit Elektronik beschäftigt. Bis heute wird dieser Timerbaustein eingesetzt, der zu den meistverkauften Halbleiterbausteinen weltweit gehört und geradezu Kultstatus genießt. Versuchen Sie doch einmal, in Ihrem Landkreis oder Ihrer Stadt ein Autokennzeichen mit diesem Kürzel anzumelden – garantiert schon vergeben!

Vielleicht sind die einfache Handhabung und die für die damalige Zeit (1972) geniale Einfachheit der Schlüssel zum Erfolg. Es lassen sich zahlreiche Schaltungen mit dem NE555 realisieren. Zu den Klassikern gehören natürlich Oszillator- und Zeitverzögerungsschaltungen. Jahrzehntlang haben sich Autoren und Tüftler damit beschäftigt, was sich sonst noch alles mit diesem Baustein anstellen lässt. Zahlreiche Bücher sind zu diesem Thema geschrieben worden. Der NE555 ist einfach aufgebaut und die verwendete

Daten NE555-EXB	Geräte-Kurzbezeichnung:	NE555-EXB
	Versorgungsspannung:	3–15 V _{DC}
	Stromaufnahme:	0,26 A max.
	Ausgangsfrequenz:	100 kHz max.
	Umgebungstemperatur:	5 bis 35 °C
	Lagertemperatur:	-40 bis +85 °C
	Abmessungen (B x H):	160 x 100 mm
	Gewicht:	80 g

Daten NE555-AWS	Geräte-Kurzbezeichnung:	NE555-AWS
	Versorgungsspannung:	3–15 V _{DC}
	Stromaufnahme:	0,26 A max.
	Ausgangsfrequenz:	100 kHz max.
	Umgebungstemperatur:	5 bis 35 °C
	Lagertemperatur:	-40 bis +85 °C
	Abmessungen (B x H):	51 x 34 mm
	Gewicht:	8 g



Schaltungstechnik ist eigentlich sehr verständlich. Doch die wenigsten beschäftigen sich wirklich mit der Funktionsweise des NE555. Wer einen Oszillator mit einem NE555 bauen will, sucht sich die entsprechende Schaltung in der Literatur oder im Internet und wendet den Schaltkreis einfach an. Dabei birgt der NE555 unendlich viele Möglichkeiten, die der Elektroniker kaum einmal tiefer ergründet. Dazu aber soll dieser Artikel sowie die vorgestellte Experimentierplattform anregen. Denn wer sich einmal ausgiebig mit der Funktionsweise beschäftigt hat, stellt schnell fest, wie einfach und zugleich genial die Technik des NE555 ist. Mit den gewonnenen Erkenntnissen lassen sich auch eigene Ideen umsetzen – dabei hilft das ebenfalls vorgestellte Anwenderboard. In der vorliegenden Kombination sind die beiden Boards übrigens auch eine hervorragende Plattform für die Ausbildung!

Experimentieren, messen, testen und praktisch anwenden

Um die Funktionsweise des NE555 besser verstehen zu können, haben wir das Experimentierboard NE555-EXB (Bild 1) entwickelt. Die Platine ist großzügig dimensioniert und dient zum Testen, Messen und Experimentieren. Die wichtigen Funktionsblöcke sind oben auf der Platine aufgedruckt. An allen wichtigen Schaltungspunkten befinden sich Messpunkte, an die z. B. ein Oszilloskop angeschlossen werden kann.

Die Bezeichnung lautet zwar NE555-xxx, es werden aber die CMOS-Varianten ICM7555 eingesetzt, da diese ICs weniger Strom benötigen. Sie sind von der Funktion her identisch (siehe auch Abschnitt „NE555 oder ICM7555?“).

Das Besondere an diesem Board ist, dass die vier Grundschaltungen Oszillator, PWM-Modulator, Monoflop und Flipflop durch einfaches Umschalten von zwei Schaltern ausgewählt werden können. Ein Lötkolben ist also zum Verändern der Parameter nicht notwendig. Die frequenzbestimmenden Kondensatoren werden mittels eines DIP-Schalters zugeschaltet.

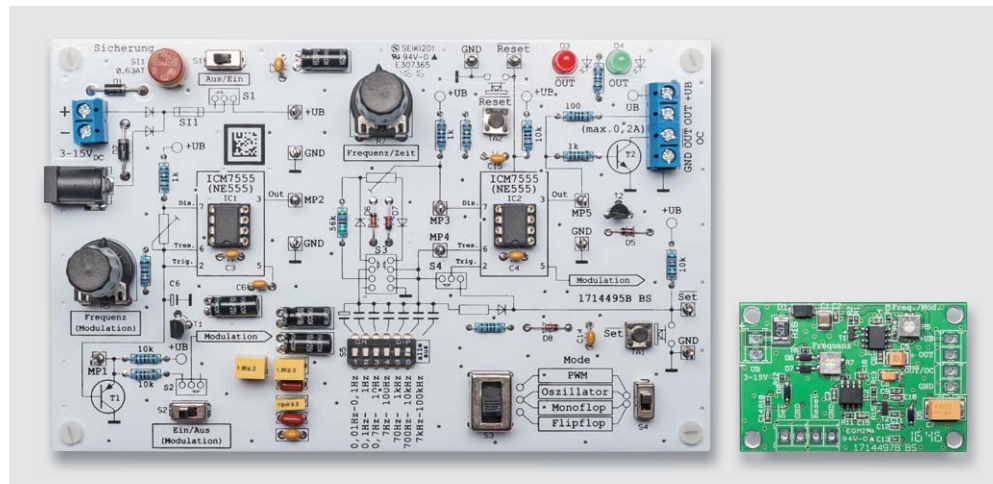


Bild 1: Beide Platinenvarianten im direkten Vergleich, links das Experimentierboard NE555-EXB, rechts das Anwenderboard NE555-AWS

Mithilfe von Potentiometern kann auf einfache Weise die Oszillatorfrequenz bzw. die Timerzeit verändert werden.

Es stehen zwei Ausgänge zur Verfügung. Der Ausgang „OUT“ führt über einen 100-Ω-Widerstand direkt auf den Ausgang des ICM7555. Der Widerstand schützt den Ausgang bei einem Kurzschluss. Der zweite Ausgang ist als „Open Collector“ ausgeführt und wird von einem Transistor (T2) getrieben, der einen Ausgangsstrom von 0,2 A schalten kann.

Es gibt zwei ICM7555 auf der Platine: IC2 ist der Hauptoszillator, während der zweite Oszillator IC1 nur der Modulation des Hauptoszillators dient. Hierdurch ist ein auf- und abschwellender Ton erzeugbar (amerikanische Polizeisirene).

Die Spannungsversorgung kann wahlweise über Batterien oder ein Steckernetzteil erfolgen. Das Experimentierboard ist für praktische Anwendungen wie z. B. den Einbau in ein Gerät natürlich nicht so gut geeignet. Aus diesem Grund gibt es die Schaltung in SMD-Ausführung als Anwenderschaltung NE555-AWS. Die Platine ist durch den Einsatz von SMD-Bauteilen sehr kompakt. Die Auswahlsschalter wurden durch Lötbrücken ersetzt. Die durch Experimentieren und Testen gefundene Schaltungsvariante mit entsprechender Dimensionierung der Kondensatoren kann direkt auf die Anwenderschaltung übertragen werden, da beide Schaltungen von der Funktion her identisch sind und auch alle Bauteilreferenzbezeichnungen gleich sind.

NE555 (ICM7555) – Grundfunktion

Der NE555 (ICM7555) ist ein integrierter Timerbaustein, dessen Innenleben recht einfach aufgebaut ist. Wie man im Blockschaltbild (Bild 2) erkennt, besteht dieser Baustein aus einem Flipflop [3] und einem Fensterkomparator, der mit zwei Operationsverstärkern reali-

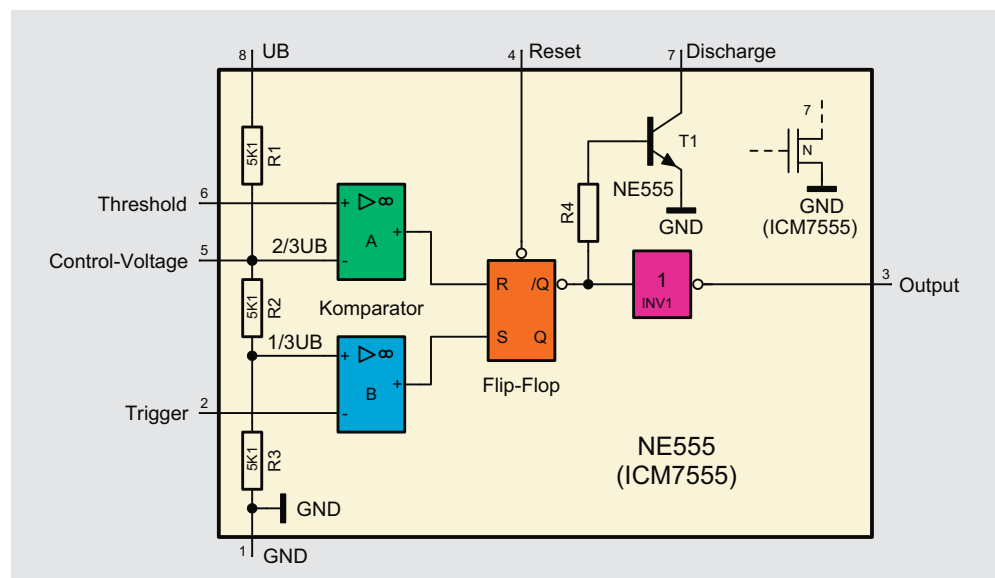


Bild 2: Das Blockschaltbild des NE555/ICM7555

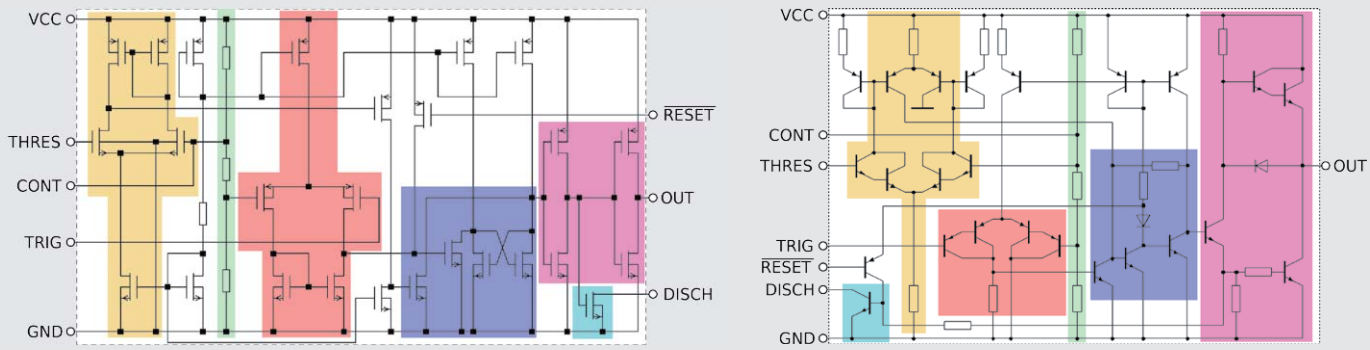


Bild 3: CMOS- und Bipolar-Version im Vergleich. Quelle: By Wdwd (Own work, GFDL CC BY 3.0). Siehe auch [4]

siert ist. Das Flipflop ist eine bistabile Kippstufe mit zwei möglichen Ausgangszuständen. Entweder ist der Q-Ausgang auf High-Pegel und der /Q-Ausgang auf Low-Pegel oder umgekehrt. Mit dem Eingang „S“ (Set) wird das Flipflop gesetzt und mit „R“ (Reset) zurückgesetzt. Gesetzt bedeutet, dass der Ausgang Q High-Pegel führt. Zusätzlich kann über den externen Anschluss „Reset“ mit einem Low-Pegel ein Reset durchgeführt werden. Ein Reset kann also über den internen „R“ (Reset) oder über den externen Reset-Eingang durchgeführt werden, wobei der externe Reset Vorrang hat.

Über einen Spannungsteiler, bestehend aus den Widerständen R1 bis R3, werden die Spannungsreferenzen für die beiden Komparatoren festgelegt. Die Schaltschwelle des oberen Komparators „A“ liegt bei $2/3$ der Betriebsspannung U_B . Bei einer Betriebsspannung von 9 V wären das 6 V ($9 \text{ V} \times 2/3$). Steigt die Spannung am zugehörigen Eingang „Threshold“ auf über $2/3$ von U_B , wechselt der Komparatorausgang auf High-Pegel und setzt das Flipflop zurück. Der untere Komparator „B“ überwacht die Spannung am Anschluss „Trigger“, dessen Schaltschwelle bei $1/3$ von U_B liegt. Da der Triggereingang mit dem negativen Eingang des Komparators „B“ verbunden ist, wechselt der Ausgang dieses Komparators beim Unterschreiten der Schaltschwelle ($1/3$) am Eingang „Trigger“ auf High-Pegel und setzt somit das Flipflop. Wie man sieht, erfolgt die Steuerung des Flipflops mit den beiden Eingängen „Trigger“ (Setzen) und „Threshold“ (Zurücksetzen/Reset). Das Flipflop besitzt, wie schon erwähnt, einen Q- und einen /Q-Ausgang. Ist das Flipflop gesetzt, liegt „Q“ auf High- und der andere Ausgang „/Q“ auf Low-Pegel. Wie man im Blockschaltbild (Bild 2) erkennt, führt der /Q-Ausgang über einen Inverter zum Anschluss (Output).

An den /Q-Ausgang ist über den Widerstand R4 ein Transistor T1 angeschlossen, der primär zum Entladen eines Kondensators in der Anwenderschaltung gedacht ist. Diese Funktion wird später in der Schaltungsbeschreibung für die jeweilige Grundschaltung erklärt.

Beim internen Spannungsteiler ist der Knotenpunkt „ $2/3U_B$ “ als Anschluss herausgeführt (Control Voltage). In erster Linie wird hier ein externer Kondensator angeschlossen, um die Spannung am

Spannungsteiler bei Schwankungen der Versorgungsspannung zu stabilisieren. Durch eine extern zugeführte Spannung an „Control Voltage“ lassen sich aber auch die Referenzspannungen des Spannungsteilers in gewissen Grenzen manipulieren. Dies kann z. B. bei einer Oszillatorschaltung zur Frequenzbeeinflussung genutzt werden. Aus diesem Grund wird dieser Eingang als „Control Voltage“, was übersetzt Steuerspannung heißt, bezeichnet.

Mit einfachen Worten kann man sagen, das Flipflop wird gesetzt, wenn die Spannung am Eingang „Trigger“ unterhalb $1/3$ von U_B fällt. Steigt die Spannung am Eingang „Threshold“ auf über $2/3$ von U_B , wird das Flipflop zurückgesetzt. Die detaillierte Beschreibung erfolgt an praktischen Beispielen in den nächsten Absätzen.

NE555 oder ICM7555?

Die Frage müsste eigentlich lauten: bipolar oder CMOS? Im Laufe der Zeit wurde der NE555 weiterentwickelt, und es entstand eine moderne CMOS-Variante. Eine Technologie in CMOS bietet den wesentlichen Vorteil, dass weniger Energie verbraucht, also weniger Strom benötigt wird. Die herkömmlichen Transistoren, mit denen ja der NE555 aufgebaut ist, wurden durch CMOS-Transistoren ersetzt (siehe Bild 3). Die Funktion beider Varianten ist identisch und somit sind diese kompatibel. Es gibt allerdings ein paar geringe Unterschiede, was die Strombelastbarkeit der Ausgänge angeht. In 90 % der Fälle kann ein NE555 gegen die CMOS-Variante getauscht werden.

Die verschiedenen Varianten sind an den Bauteil-Bezeichnungen erkennbar:

Bipolar-Variante: NE555, LM555, MC14555, KA555

CMOS-Variante: ICM7555, TLC555, LMC555

Schaltung

In den Bildern 4 und 5 sind die Schaltbilder für das NE555-EXB und das NE555-AWS dargestellt. Technisch gesehen sind beide Schaltbilder identisch, lediglich bei der Spannungsversorgung gibt es geringe Unterschiede.

Wie man erkennt, kommen zwei ICM7555, also der CMOS-Ersatz für den NE555, zum Einsatz. IC1 ist als Oszillator mit einem festen Frequenzbereich aufgebaut. Wie genau die Oszillatorschaltung funktioniert, erfährt man im Abschnitt „Grundschaltung Oszillator“. Dieser mit IC1 realisierte Oszillator dient nur zur Modulation des zweiten, mit IC2 aufgebauten Oszillators. Genutzt wird die annähernd dreieckförmige Spannung am Kondensator C5. Eine echte Dreiecksspannung würde man allerdings nur mit einem konstanten Lade- und Entladestrom erreichen. Über einen Vorwiderstand als Ladestrombegrenzung ergibt sich eine exponentielle Kurve. Die Kondensatorspannung gelangt auf einen als Emitterfolger geschalteten Transistor T1, der zur Entkopplung (Impedanzwandler) dient. Die Modulation kann mit dem Schalter S2 bzw. dem Jumper J1-A aktiviert werden, wodurch das Modulationssignal auf den



Steuereingang (Control Voltage/Pin 5) des zweiten Oszillators gelangt. Die Modulationsfrequenz kann mit dem Trimmer R5 in einem Bereich von 0,1 bis 1 Hz verändert werden.

Der zweite Schaltungsteil mit IC2 ist etwas komplizierter aufgebaut. Durch die Außenbeschriftung sollen unterschiedliche Betriebsmodi möglich sein. Der zeitbestimmende Kondensator kann mithilfe des

DIP-Schalters bzw. der Jumper J5-x ausgewählt werden. Es sind Kondensatorwerte von 47 pF (C13) bis 100 µF (C7) möglich. So können unterschiedliche Frequenzen bzw. Timerzeiten eingestellt werden. Durch die Kombination der Schalter „Mode 1“ und „Mode 2“ können unterschiedliche Grundschaltungen wie z. B. Oszillator, PWM-Generator usw. bequem ohne Lötarbeiten ausgewählt werden. Bei der Anwenderschal-

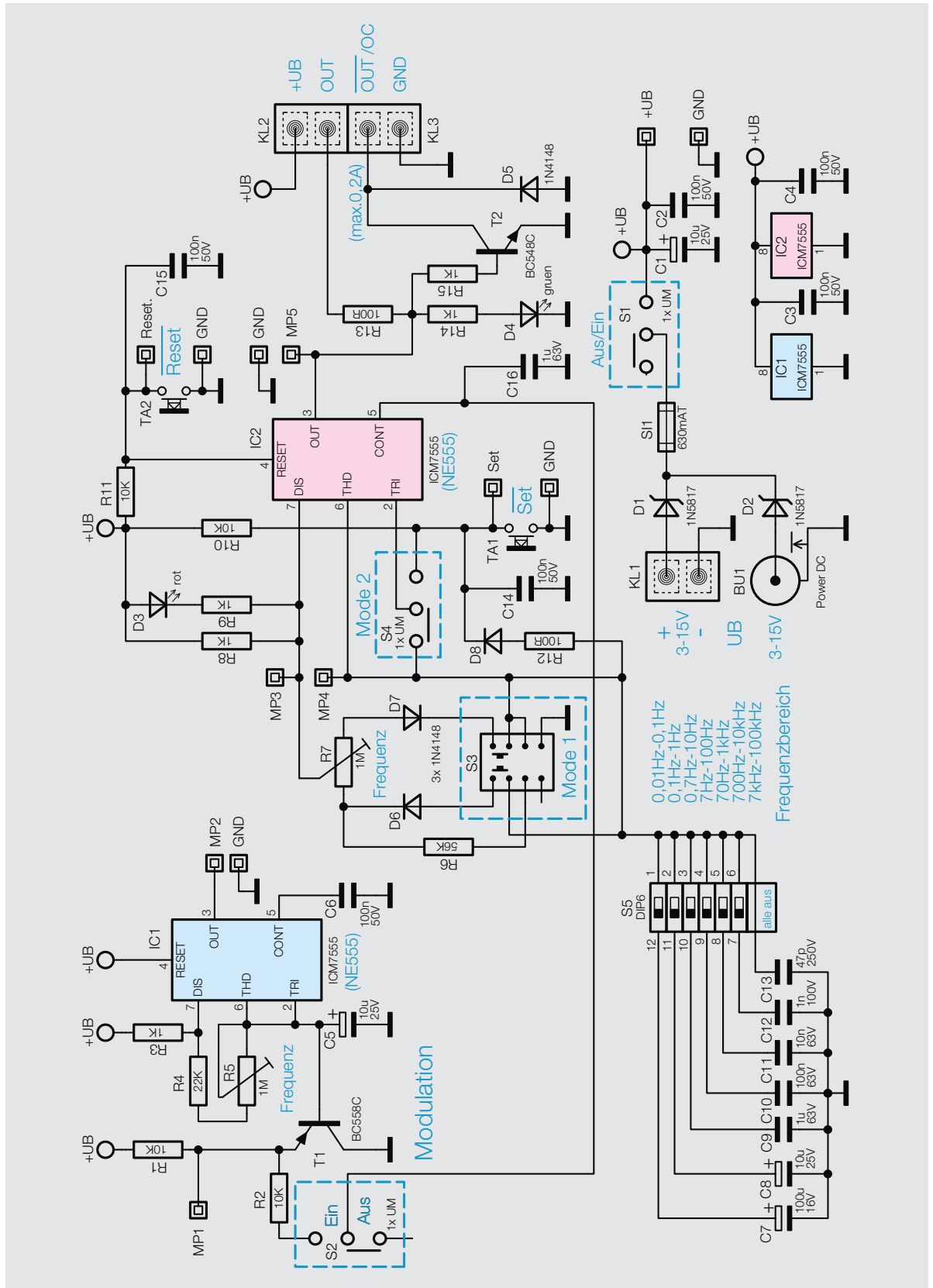


Bild 4: Das Schaltbild des Experimentierboards NE555-EXB

tung müssen allerdings die entsprechenden Jumper gebrückt (verlötet) werden, die man vorher am Experimentierboard ermittelt hat. Welche Bauteile für welchen Betriebsmode aktiviert werden, ist in den folgenden Abschnitten erklärt.

Eine Frequenz- bzw. Zeiteinstellung erfolgt mit dem Trimmer R7. Pin 3 von IC2 bildet den Ausgang, der über den Widerstand R13 auf den Anschluss „OUT“

führt. Für höhere Ströme gibt es einen Schaltertransistor T2, der als „Open Collector“ geschaltet ist, um z. B. ein Relais anzusteuern. Dieser Ausgang ist als „/OUT“ (OC) bezeichnet und kann max. 200 mA schalten. Die Diode D5 ist eine Freilaufdiode, sie schützt den Transistor T2 bei induktiven Lasten (Relais). Für die Betriebsmodi Flipflop und Monoflop sind zwei Taster (Set und Reset) bzw. die entsprechenden Eingänge vorhanden.

Beim Experimentierboard NE555-EXB wird der Ausgangszustand über zwei Leuchtdioden angezeigt. Die grüne LED (D4) ist über den Vorwiderstand R14 direkt mit dem Ausgang des Timers IC2 verbunden und zeigt einen High-Pegel am Ausgang an. Liegt am Ausgang ein Low-Pegel an, wird die rote LED D3 aktiv, die über den Vorwiderstand R9 mit dem internen Entladetransistor am Anschluss „Discharge“ (Pin 7) verbunden ist. Im Prinzip hätte man die LEDs auch direkt am Ausgang Pin 3 nach UB schalten können, aber dies würde zu einer unnötigen Strombelastung führen. Bei der Anwenderschaltung sind diese beiden LEDs nicht vorhanden. Beim NE555-EXB kann die Versorgungsspannung zum einen über den Hohlsteckerkontakt BU1 oder die Klemme KL1 zugeführt werden. Beim NE555-AWS ist dies der Anschluss „UB 3-15V“. Wie die Bezeichnung verrät, kann die Versorgungsspannung im Bereich von 3 bis 15 V liegen. Beim Experimentierboard dient eine Sicherung (SI1) zum Schutz der Schaltung, während beim NE555-AWS ein reversibler PTC-Widerstand diese Aufgabe erfüllt. Die Dioden D1 (D2) dienen bei beiden Schaltungsvarianten als Verpolungsschutz. Beim Experimentierboard ist zusätzlich noch ein Ein-/Aus-Schalter für die Betriebsspannung vorhanden.

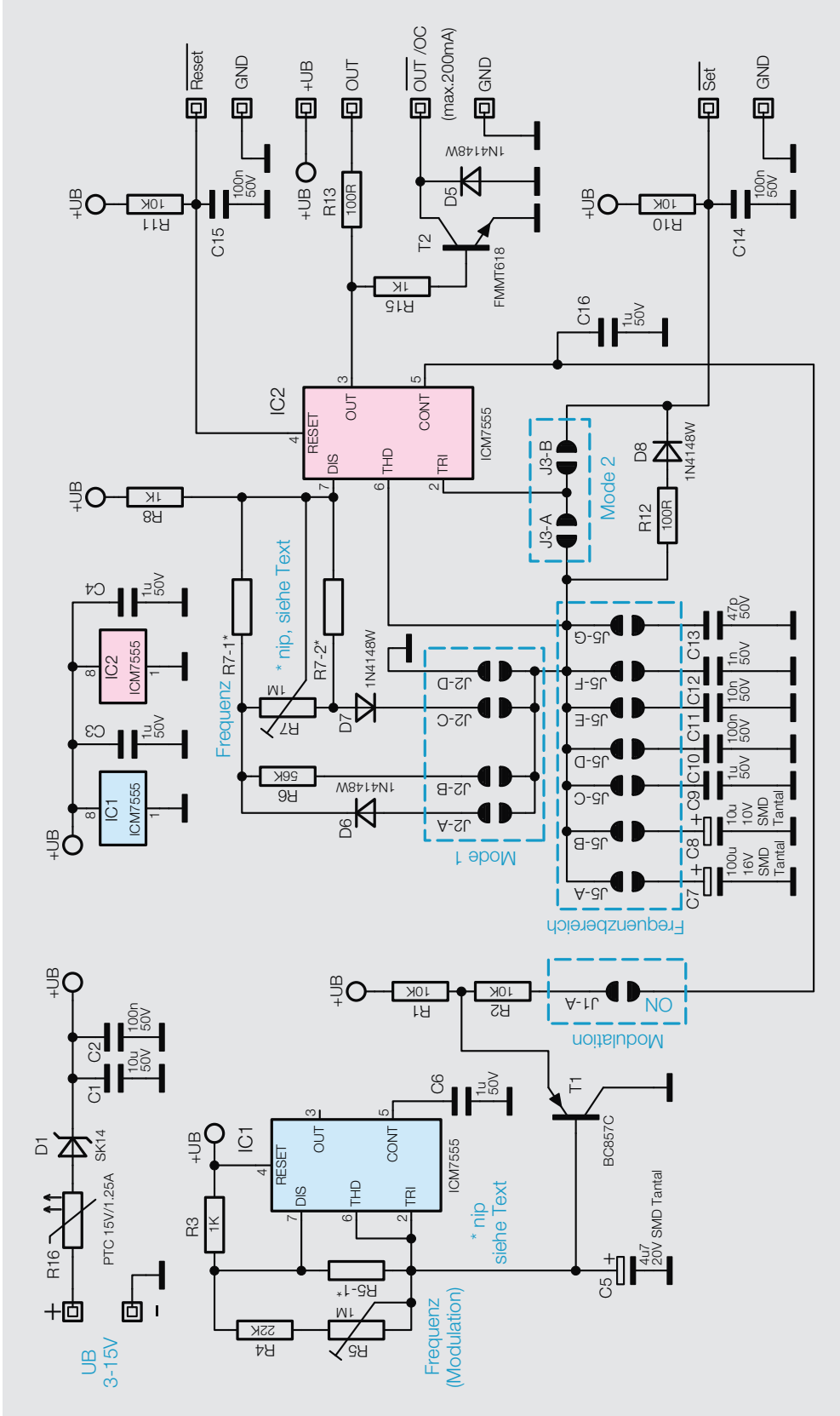


Bild 5: Das Schaltbild der Anwendungsschaltung NE555-AWS

Die Grundschaltungen und Experimente

Grundschaltung – Flipflop

Einen NE555 als Flipflop zu beschalten ist wohl die Schaltung mit dem minimalsten Aufwand. Schließlich ist ein wesentlicher Bestandteil des NE555 ein Flipflop, genauer gesagt ein RS-Flipflop. Diesen gilt es, mittels der vorgeschalteten internen Komparatoren zu steuern.

In Bild 6 ist die Grundschaltung mit zugehörigem Zeitdiagramm dargestellt. Die rot hervorgehobenen Schaltungsteile sind die in diesem Mode aktiven Bauelemente.

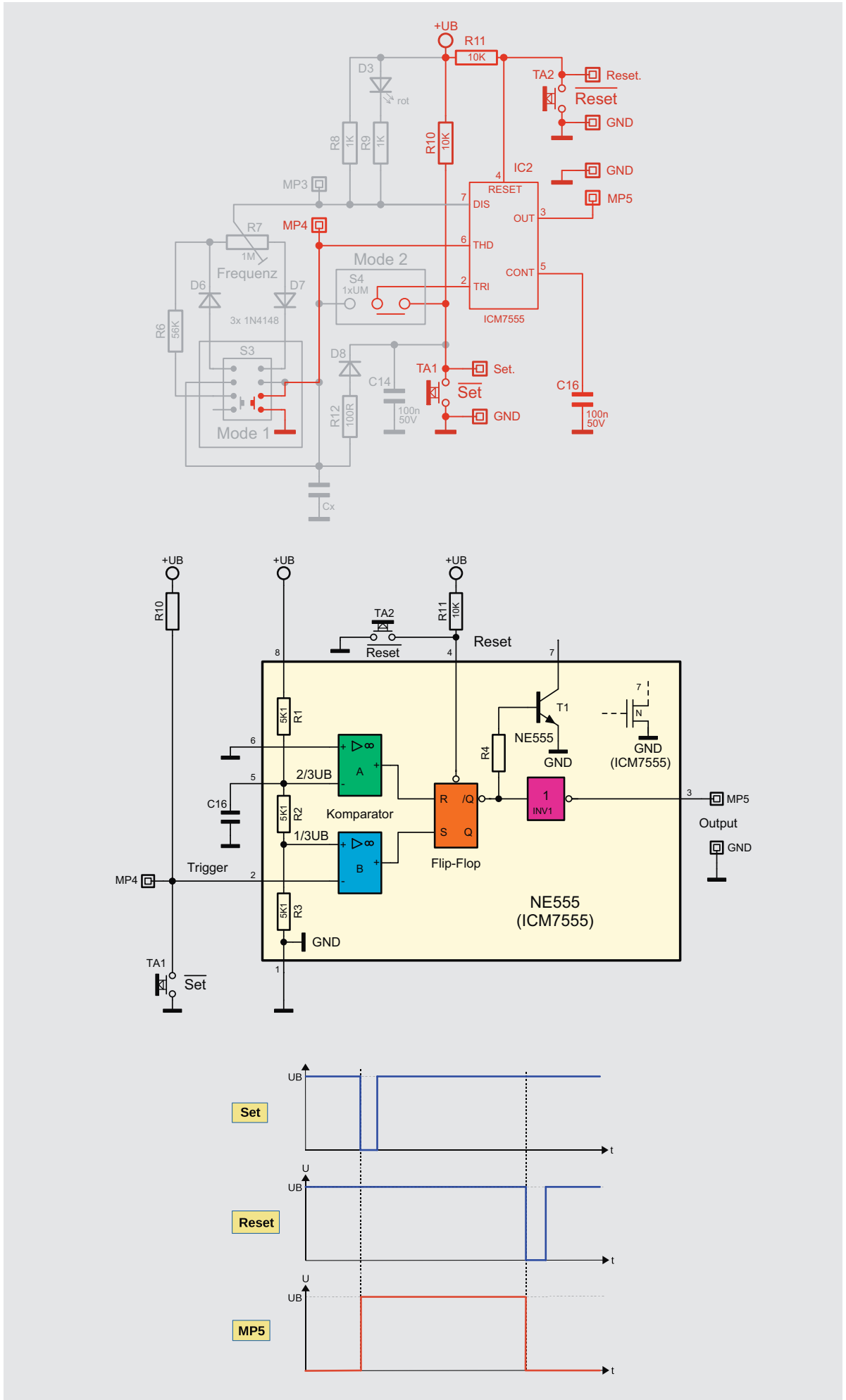


Bild 6: Die Grundschtung „Flipflop“

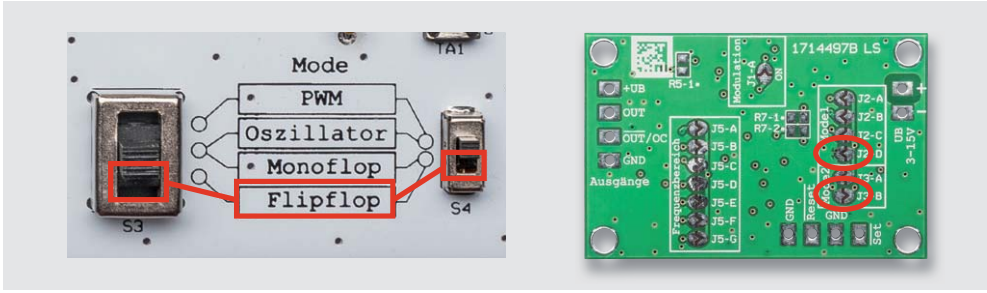


Bild 7: So werden die Jumper für den Mode „Flipflop“ gesetzt.

In Bild 7 sind die Schalter- bzw. Jumperstellungen für diesen Mode dargestellt. Beim Anwenderboard müssen J2-D und J3-B gebrückt werden.

Wie wir im Abschnitt „NE555 (ICM7555) – Grundfunktion“ erfahren haben, hat man über die beiden Eingänge „Threshold“ und „Trigger“ Zugriff auf die Komparatoren A und B und somit auch indirekt Zugriff auf das Flipflop. Das Flipflop kann gesetzt werden, indem die Spannung an „Trigger“ unterhalb von $1/3$ UB sinkt. Im Normalfall liegt „Trigger“ über R10 auf Pegel der Betriebsspannung UB. Durch den Taster TA1 bzw. den Eingang „/Set“ kann der Eingang „Trigger“ auf Masse gezogen werden. Jetzt ist die Schaltschwelle des Komparators A unterschritten, und das Flipflop des NE555 wird gesetzt. Der Ausgang „Output“ des NE555 führt jetzt High-Pegel.

Im Prinzip kann das Flipflop durch eine Spannung größer als $2/3$ UB am Eingang „Threshold“ zurückgesetzt werden. Damit beide Steuereingänge low-aktiv sind, wird das Flipflop durch einen Low-Pegel am Anschluss „/Reset“ zurückgesetzt, der normalerweise über R11 an +UB liegt. Der Eingang „Threshold“ liegt in unserem Fall auf Masse und hat keine Funktion.

Grundschaltung – Monoflop

Eine sehr häufige Anwendung für einen NE555 ist die Erzeugung von zeitlich begrenzten Impulsen, wie sie z. B. bei Ausschaltverzögerungen benötigt werden. Diese Schaltungsvariante wird als monostabile Kippstufe, kurz Monoflop, bezeichnet. Durch einen Triggerimpuls oder Pegelsprung wird ein in der zeitlichen Länge bestimmtes Signal erzeugt, und zwar nur ein Impuls, daher der Begriff Mono.

In Bild 8 ist die Grundschaltung mit zugehörigem Zeitdiagramm dargestellt. Die rot hervorgehobenen Schaltungsteile sind die in diesem Mode aktiven Bauelemente. In Bild 9 sind die Schalter- bzw. Jumperstellungen für diesen Mode dargestellt.

Aus dem Grundmodus Flipflop wissen wir, dass das interne Flipflop des NE555 durch die beiden Steuer- bzw. Komparatoreingänge „Trigger“ gesetzt und mit „Threshold“ zurückgesetzt wird.

Durch den Taster TA1 bzw. dem Eingang „/Set“ wird der Eingang „Trigger“ kurz auf Masse gezogen und das Flipflop gesetzt. Normalerweise liegt „Trigger“ über R10 auf UB. Der Ausgang MP5 wechselt nun auf High-Pegel. Über die Widerstände R6 bis R8 wird der Kondensator Cx anschließend aufgeladen, was man an der steigenden Ladekurve im Zeitdiagramm (Bild 8 unten) sieht.

Die Kondensatorspannung von Cx ist mit dem Eingang „Threshold“ des Komparators A verbunden. Erreicht die Spannung am Kondensator $2/3$ der Betriebsspannung UB, schaltet der Ausgang des Komparators A auf High-Pegel und setzt das Flipflop zurück. Möchte man einen erneuten Impuls erzeugen, ist dies durch Betätigen des Taster TA2 (Set) möglich. Mit der Diode D8 und dem Widerstand R8 wird eine mögliche vorhandene Restspannung von Cx, gleichzeitig beim Betätigen von „/Set“ entladen.

Grundschaltung – Oszillator

Die Oszillatorschaltung ist wohl die häufigste Anwendung für den NE555. Wie die Schalter bzw. die Lötbrücken gesetzt werden müssen, um diese Betriebsart zu aktivieren, ist in Bild 11 dargestellt. Beim Anwenderboard müssen J2-B und J3-A gebrückt werden.

Der Frequenzbereich kann beim NE555-EXB direkt über die DIP-Schalter bzw. beim NE555-AWS durch das Auflöten einer Brücke ausgewählt werden. Bei der kleinen Anwenderschaltung reicht der Platz für eine Beschriftung auf der Platine nicht aus. Aus diesem Grund entnimmt man die Auflistung der Frequenzen und die Jumper-Zuordnung der Tabelle 1.

Die Modulation sollte deaktiviert sein, also Schalter S2 auf „Aus“ bzw. Jumper J1-A nicht bestückt. Zu der Funktion „Modulation“ kommen wir später noch.

Schauen wir uns nun im Detail an, wie dieser Oszillator arbeitet. Hierzu betrachten wir auch die Zeichnungen in Bild 2 und den Abschnitt „NE555 (ICM7555) – Grundfunktion“. Im oberen Teil von Bild 10 ist ein Ausschnitt des Schaltbilds dargestellt. Die nicht aktiven Bauteile bzw. Schaltungsbereiche sind grau hinterlegt. An den Messpunkten MP3 bis MP5 können die Oszillogramme aufgenommen werden. In der Bildmitte ist ein vereinfachtes Schaltbild und unten die zugehörigen Oszillogramme dargestellt. Der Kondensator Cx steht stellvertretend für einen der wählbaren Kondensatoren C7 bis C13.

Gehen wir von dem Fall aus, dass der Kondensator Cx entladen ist. In diesem Fall ist die Spannung an Cx (MP4) gleich 0 V. Da der Kondensator Cx mit dem Eingang (Trigger) des unteren Komparators verbunden ist und die Spannung an Cx (MP4) unterhalb $1/3$ der Betriebsspannung liegt, wird das Flip-Flop über den Setzeingang „S“ gesetzt und der Q-Ausgang führt High-Pegel bzw. der/Q-Ausgang Low-Pegel. Da zwischen dem Ausgang /Q vom Flipflop und dem Ausgang Pin 2 (Output) noch ein Inverter geschaltet ist, liegt am Ausgang (MP5) ein High-Pegel an.

Nun lädt sich Cx über die Widerstandskette R6, R7 und R8 auf. Im Zeitdiagramm sieht man, wie die Spannung an Cx auf einen Wert von $2/3$ UB ansteigt. Denn ab diesem Punkt wechselt der Ausgang des oberen Komparators „A“ auf High-Pegel, und das Flip-Flop wird zurückgesetzt.

Frequenzbereiche für die Jumperstellung J5 beim NE555-AWS	
Jumper J5	Frequenzbereich
J5-A	0,01 Hz – 0,1 Hz
J5-B	0,1 Hz – 1 Hz
J5-C	0,7 Hz – 10 Hz
J5-D	7 Hz – 100 Hz
J5-E	70 Hz – 1 kHz
J5-F	700 Hz – 10 kHz
J5-G	7 kHz – 100 kHz

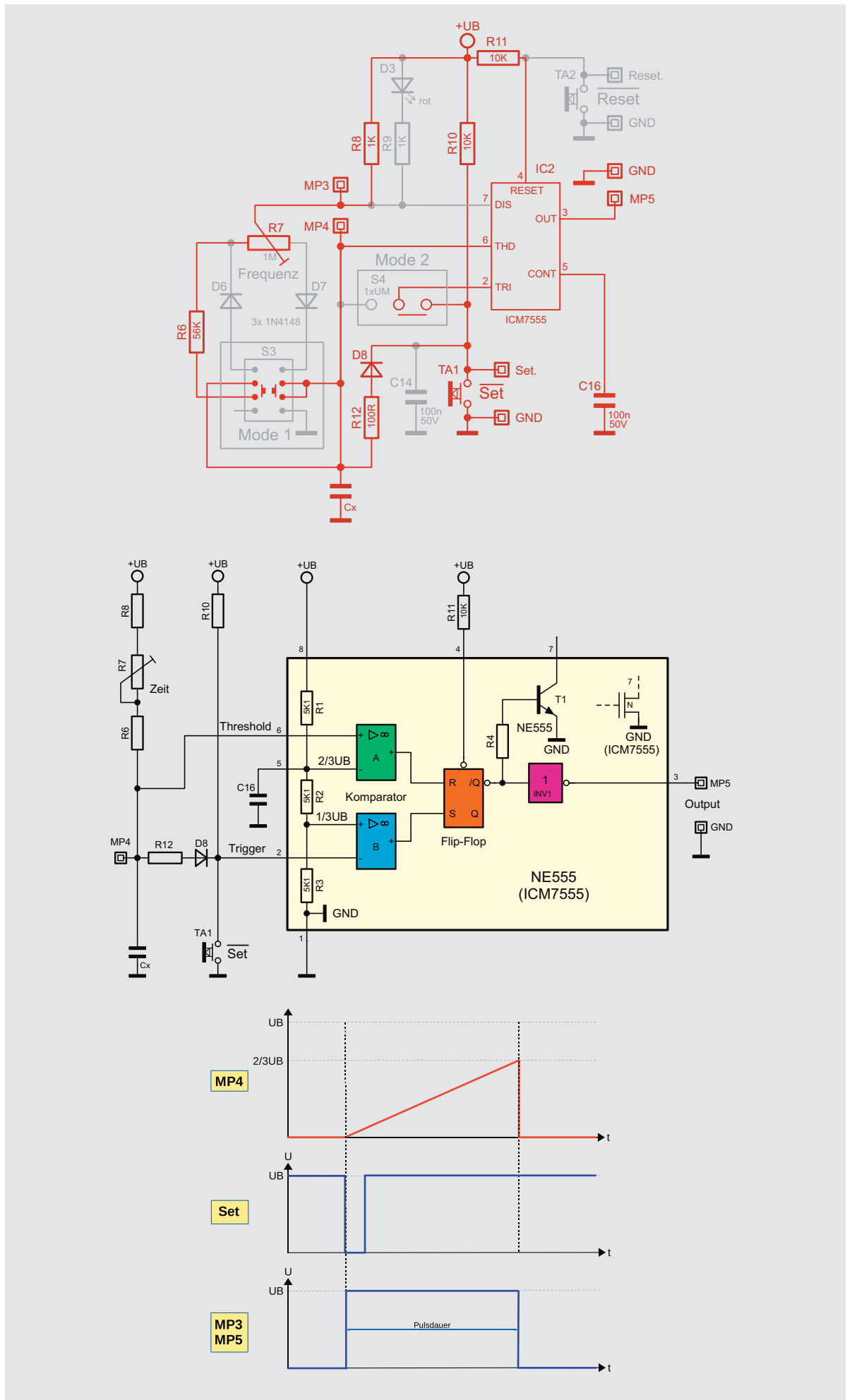


Bild 8: Die Grundschtung „Monoflop“

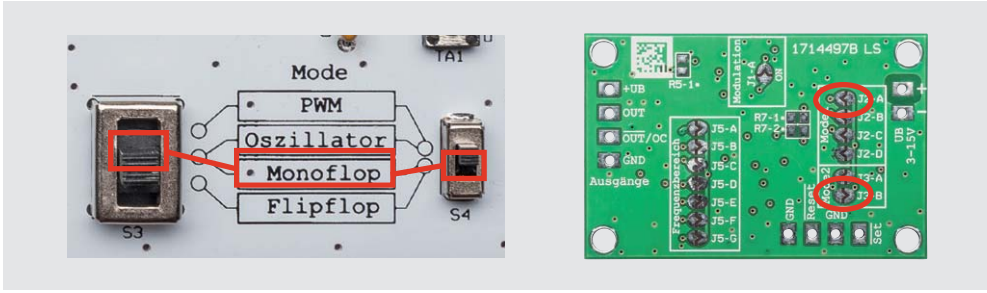
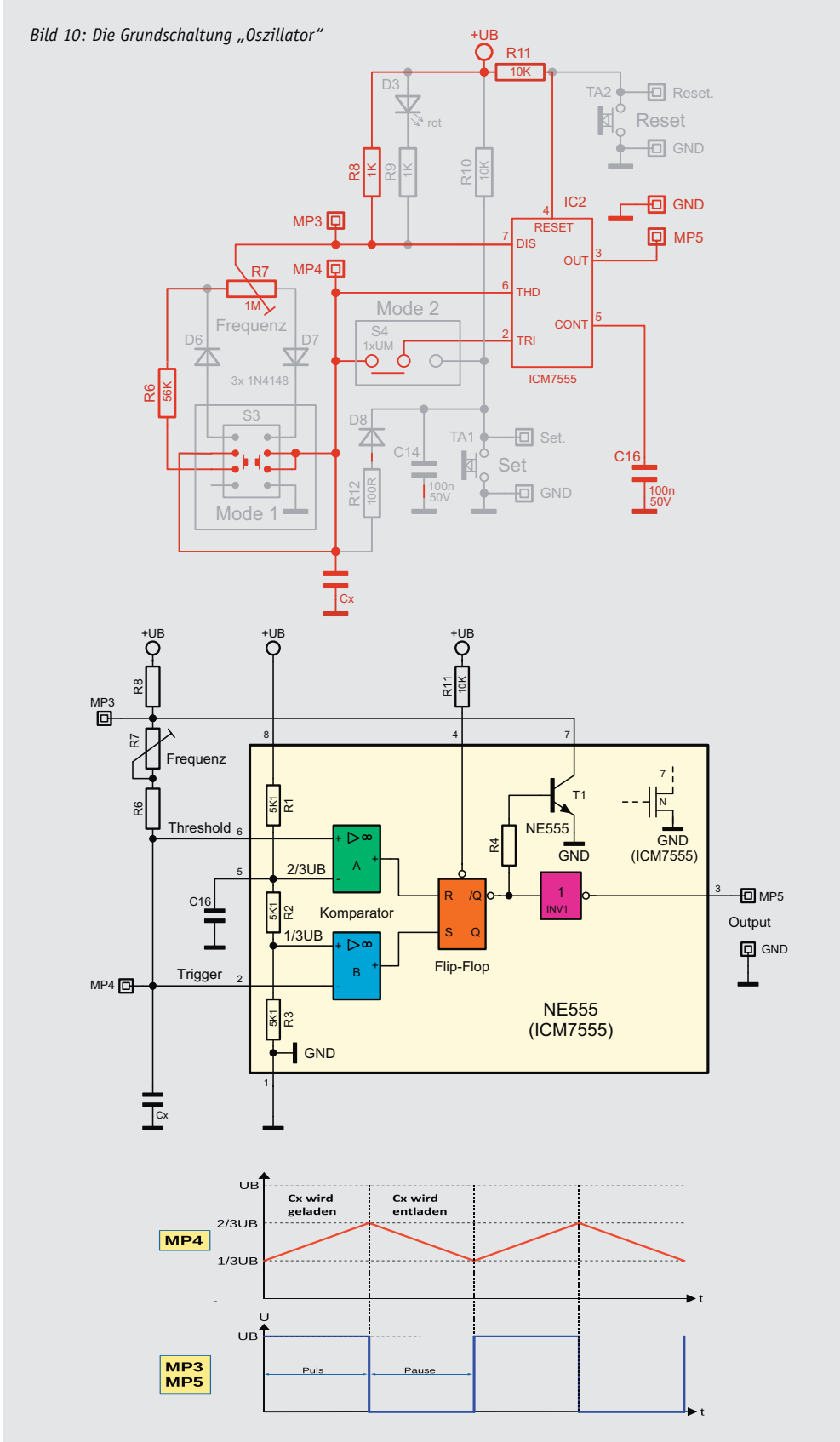


Bild 9: So werden die Jumper für den Mode „Monoflop“ gesetzt.



Nun wird der interne Entladetransistor des NE555 (Ausgang „Discharge“) leitend, und über die Widerstände R6 und R7 wird Cx entladen – die Spannung sinkt wieder, bis $1/3$ von UB erreicht ist. Ab diesem Zeitpunkt setzt der untere Komparator „B“ das Flipflop wieder. Dieser Vorgang wiederholt sich periodisch – wir haben einen Oszillator.

Mit dem Widerstand R7 kann die Frequenz in einem weiten Bereich verändert werden. Beim Laden von Cx sind die Widerstände R6 bis R8 vorgeschaltet, beim Entladen hingegen wirken nur R6 und R7. Der Widerstand R8 ist im Gegensatz zur Summe von R6 und R7 relativ klein. Aus diesem Grund kann man sagen, dass Lade- und Entladezeit fast gleich sind. Somit ist das Puls-Pausenverhältnis vom Ausgangssignal ca. 50 zu 50.

Grundsaltung – PWM-Generator

Der PWM-Generator (Pulsweitenmodulator) ist eine leicht abgewandelte Oszillatorschaltung. Es soll nicht die Frequenz, sondern das Verhältnis von Puls- und Pausenzeit eingestellt bzw. verändert werden. Diese Schaltungstechnik wird vorwiegend zur Helligkeitseinstellung oder Motorregelung verwendet. Wie die Schalter bzw. die Lötbrücken gesetzt werden müssen, um diese Betriebsart zu aktivieren, ist in Bild 12 dargestellt. Beim Anwenderboard müssen J2-A, J2-C und J3-A gebrückt werden. Durch den DIP-Schalter bzw. die Lötbrücken J5-x kann eine feste PWM-Frequenz ausgewählt werden. Eine variable Frequenzeinstellung ist hier nicht vorgesehen.



In [Tabelle 2](#) sind die PWM-Frequenzen für die einzelnen Schalter- bzw. Jumperstellungen dargestellt.

Wie schon im Absatz „Oszillator“ erwähnt, sollte man sich zum besseren Verständnis auch den Absatz „NE555 (ICM7555) – Grundfunktion“ anschauen. Im oberen Teil von [Bild 13](#) ist ein Ausschnitt der realen Schaltung dargestellt. Zur besseren Verständlichkeit ist die Schaltung in der Bildmitte noch einmal vereinfacht mit dem Innenleben des NE555 gezeichnet.

Wir gehen von der Ausgangssituation aus, dass der Cx entladen ist. Die Spannung am Eingang „Trigger“ beträgt 0 V und das Flipflop ist gesetzt. Am Ausgang „Output“ (MP5) liegt High-Pegel. Der interne Transistor T1 vom NE555 sperrt.

Über R8, R7 und die Diode D7 fließt ein Ladestrom in den Kondensator Cx – die Spannung über Cx steigt. Der Knotenpunkt MP4 liegt am Eingang „Threshold“ des NE555 an. Dies ist auch die Spannung, die an Cx anliegt. Erreicht diese Spannung 2/3 von UB, schaltet der obere Komparator „A“ am Ausgang auf High-Signal und das Flipflop wird zurückgesetzt. Hierdurch wechselt das Flipflop seine Ausgangszustände. Der Ausgang „Output“ (MP5) führt nun Low-Pegel und der Transistor T1 wird leitend. Und jetzt erkennt man den wesentlichen Unterschied zur „normalen“ Oszillatorschaltung. Da beim Entladen von Cx ein Strom aus dem Kondensator fließt, sperrt die Diode D7 und der Strom muss über die entgegengesetzte Diode D6 fließen. Der Entladestrom von Cx fließt also über D6 und den Teilwiderstand von R7 in den internen Transistor T1 (Discharge).

Auch hier wird die Spannung an MP4 überwacht. Dieses Mal reagiert wieder der untere Komparator „B“, und zwar genau dann, wenn die Spannung an Cx unterhalb von 1/3 UB sinkt. Jetzt wechselt der Ausgangszustand des Flipflops wieder – der Ausgang „Output“ ist wieder auf High-Pegel und der Transistor T1 sperrt. Der Entladevorgang wird abgebrochen und stattdessen wird Cx wieder geladen. Wir sind wieder am Anfang unserer Beschreibung. Dieser Vorgang setzt sich nun periodisch fort.

Der Unterschied zum Oszillator besteht darin, dass es für den Lade- und Entladevorgang zwei unterschiedliche Signalwege gibt. Dies wird mit den beiden Dioden D6 und D7 erreicht, die ja bekanntlich den Strom nur in einer Richtung durchlassen. Je nach Stellung des Trimmers R7 ergibt sich nun ein unterschiedlicher Lade- bzw. Entladewiderstand. Steht der Trimmer R7 mittig, sind beide Teilwiderstände gleich groß und es ergibt sich ein Puls-Pausen-Verhältnis von 50 zu 50. Wird R7 nun zu einer Seite gedreht, verändern sich die Zeiten, d. h., einmal wird die Ladezeit größer und entsprechend die Entladezeit kleiner oder genau umgekehrt. Das Entscheidende ist, dass die Summe aus Lade- und Entladezeit immer gleich bleibt, da sich der Gesamtwiderstand von R7 nicht ändert. Die Frequenz bleibt durch Veränderung von R7 immer gleich, nur das Puls-Pausen-Verhältnis variiert. Beim Experimentierboard befinden sich zwei LEDs auf der Platine, die mit dem Ausgang des NE555 verbunden sind. Die grüne LED zeigt den Pegelzustand am Ausgang (OUT) an, während die rote LED mit dem invertierten Ausgang „/OUT“ verbunden ist. Durch die Veränderung vom Verhältnis Puls/Pause verändert sich auch die Helligkeit der LEDs. Mal wird die rote LED heller und die grüne dunkler oder umgekehrt. In Mittelstellung von R7 sind beide LEDs gleich hell. Hier erkennt man die Funktionsweise der Helligkeitseinstellung mit PWM.

Grundschialtung – Sirene

Ein besonderes Gadget nicht nur für unsere jüngeren Leser und Anwender ist die beliebte Sirenschaltung, die es seit Jahrzehnten in dieser Form gibt. Bisher haben wir uns mit dem Oszillator IC2 befasst. Wie man im Blockschaltbild ([Bild 14](#)) erkennt, ist noch ein zweiter NE555 (IC1) vorhanden. Wie ein Oszillator funktioniert, ist im Absatz „Grundschialtung – Oszillator“ beschrieben. Wir nehmen nun noch einen zweiten Oszillator hinzu, mit dem wir den Hauptoszillator modulieren.

In [Bild 15](#) sind die Schalter- bzw. Jumperstellungen für diesen Mode dargestellt.

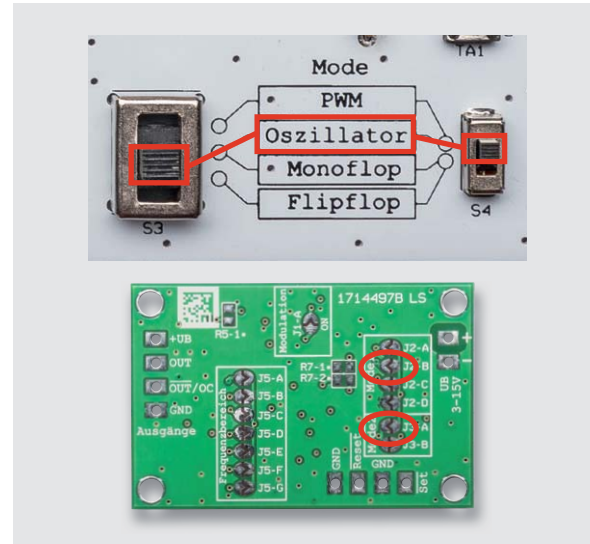


Bild 11: So wird der Betriebsmode „Oszillator“ bei den unterschiedlichen Platinen aktiviert.

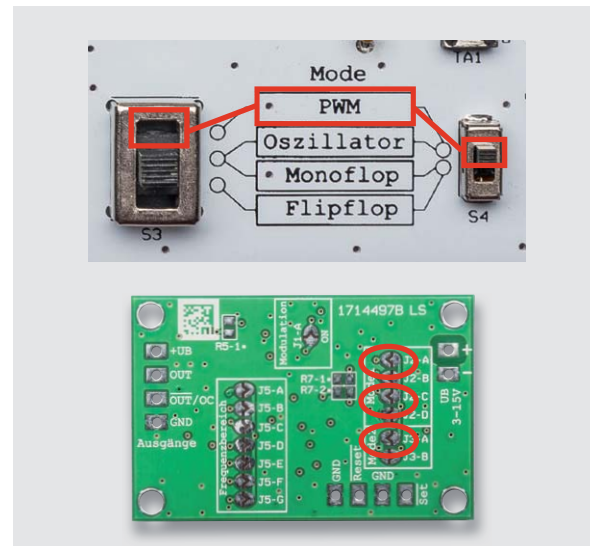


Bild 12: So wird der Betriebsmode „PWM“ bei den unterschiedlichen Platinen aktiviert.

Der erste Oszillator IC1 erzeugt eine sehr niederfrequente, sägezahnähnliche Spannung. Diese Spannung gelangt über den Schalter S2 auf den Steuerungseingang (Control Voltage) des zweiten Oszillators IC2. Über einen Widerstand kann direkt ein kleiner Lautsprecher an den Ausgang angeschlossen werden. Durch die Modulation wird ein an- und abschwelliger Ton ähnlich einer Polizeisirene erzeugt.

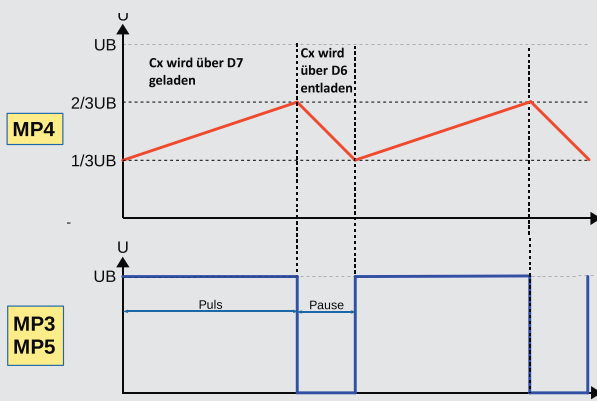
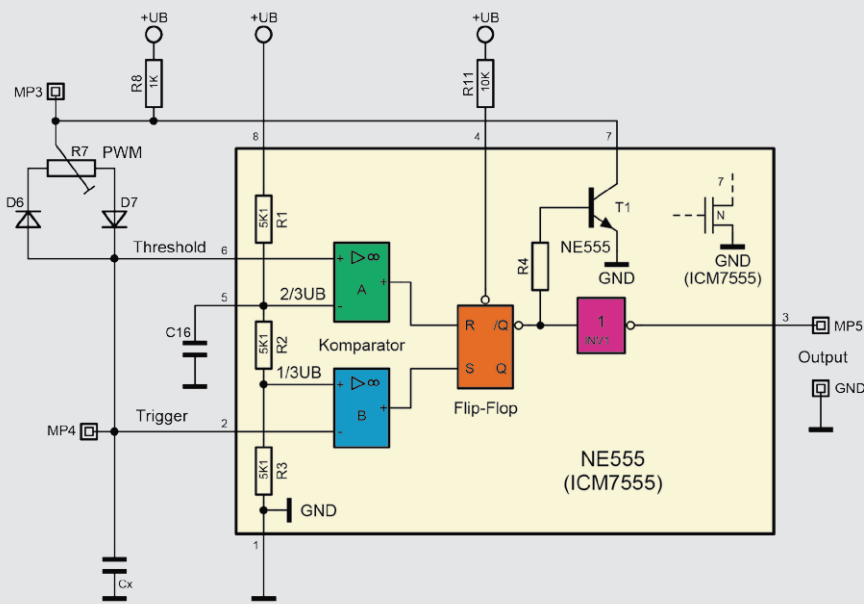
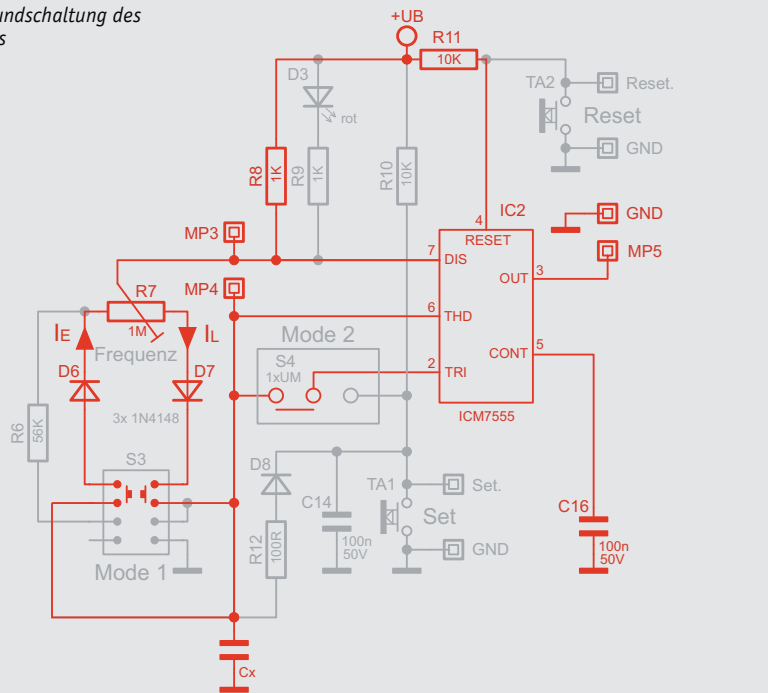
Frequenzen für den PWM-Mode		
Schalter S5	Jumper J5	PWM-Frequenz
S5-1	J5-A	0,09 Hz
S5-2	J5-B	0,09 Hz
S5-3	J5-C	0,9 Hz
S5-4	J5-D	9 Hz
S5-5	J5-E	90 Hz
S5-6	J5-F	900 Hz
S5-offen	J5-G	9 kHz

Tabelle 2



Hier noch ein Hinweis zu den nicht bestückten SMD-Widerständen R5-1, R7-1 und R7-2: Anstelle des Potentiometers können hier die entsprechenden Festwiderstände eingelötet werden. Dieses Feature richtet sich an erfahrene Elektroniker, denn hierzu müssen die SMD-Trimmer ausgelötet und mit den beschriebenen Widerständen ersetzt werden.

Bild 13: Die Grundschtung des PWM-Generators



Inbetriebnahme und Bedienung

Das Experimentierboard verfügt zur Spannungsversorgung über eine Klemmleiste, an die eine Spannungsquelle angeschlossen werden kann. Die Hohlsteckerbuchse bietet auch die Möglichkeit, direkt ein passendes Steckernetzteil anzuschließen. Für die ersten Versuche kann eine 9-V-Batterie verwendet werden. Hierzu wird der beiliegende Batterieclip an die Klemme KL1 angeschlossen. Die rote Zuleitung wird mit „+“ und die schwarze Zuleitung mit „-“ verbunden. Nach Anklempfen der 9-V-Batterie kann die Schaltung in Betrieb genommen werden.

Nun kann die Schaltung eingeschaltet werden. Je nach Grundschaltung müssen die entsprechenden Schalter S1 und S2 in die richtige Stellung gebracht werden. Dies ist im jeweiligen Abschnitt „Grundschaltung“ erklärt. Die Auswahl des Kondensators mit dem Dip-Schalter S5 gibt den Frequenzbereich bzw. die Timerzeit vor. Der Frequenzbereich ist dem jeweiligen Schalter zugeordnet und auf der Platine aufgedruckt. Die Schalterstellung „alle aus“ bedeutet, dass keiner der einzelnen DIP-Schalter auf „ON“ steht, also alle ausgeschaltet sind. In diesem Fall ist nur der Kondensator C13 (47 pf) aktiv. Beim Flipflop- und Monoflop-Mode werden die Taster „/Set“ und „/Reset“ verwendet. Bei Bedarf können an die Anschlusspunkte (Lötstifte) auch externe Taster bzw. ein Logikausgang einer anderen Schaltung angeschlossen werden. Hierbei ist zu beachten, dass die zugeführte Spannung mit der Betriebsspannung UB des Boards identisch sein muss.

An der Klemmleiste KL2 und KL3 stehen zwei unterschiedliche Ausgänge zur Verfügung. Der Ausgang „OUT“ ist über einen Widerstand direkt mit dem Ausgang des Timers IC2 verbunden. Hier können z. B. kleine Verbraucher wie LEDs mit angeschlossen werden. Ein Vorwiderstand von 100 Ω ist ja schon in Reihe geschaltet, sodass bei einer Betriebsspannung bis 5 V kein zusätzlicher Vorwiderstand zwischen Ausgang und LED eingesetzt werden muss. Im Zwei-

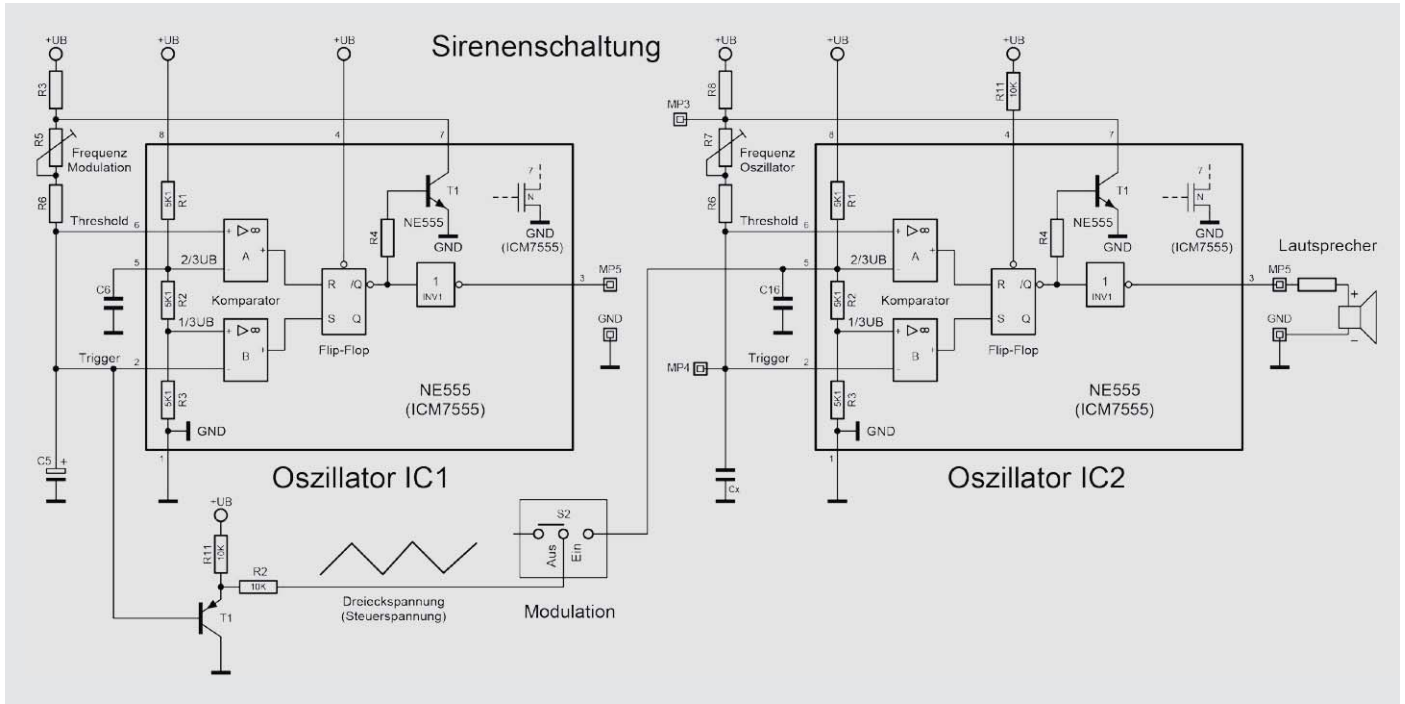


Bild 14: Die Sirenschaltung nutzt beide NE555 auf den Platinen.

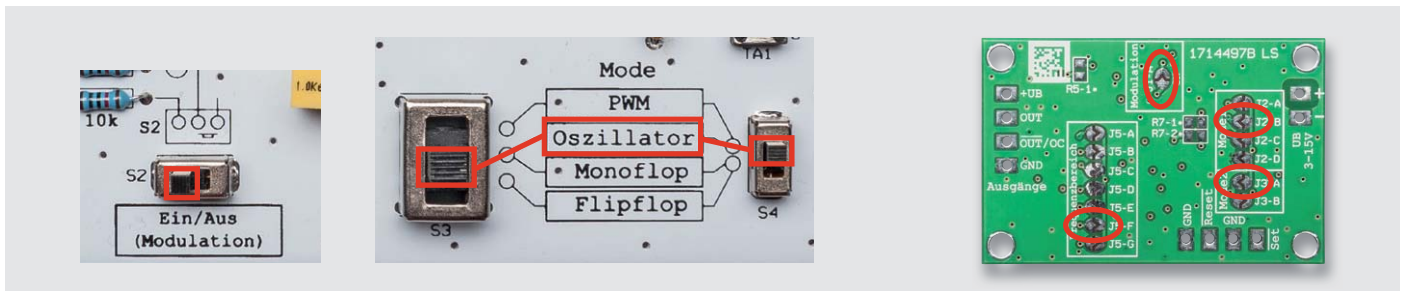


Bild 15: So wird die Sirenschaltung aktiviert.

felsfall muss noch ein Vorwiderstand von ca. 220 bis 470 Ω eingefügt werden. Für etwas „größere“ Verbraucher, wie z. B. ein Relais ist der Open-Collector-Ausgang „/OUT“ vorgesehen. Dieser kann eine Last von bis zu 200 mA treiben. Open Collector bedeutet, dass dieser Ausgang mit einem Transistor gegen Masse geschaltet wird. Der Verbraucher, z. B. ein Relais, muss deshalb zwischen den Anschluss „/UB“ und „/OUT OC“ angeschlossen werden (siehe Bild 16).

Bei der Anwenderschaltung gelten alle bisher gemachten Angaben in gleicher Weise. Hier sind die Anschlusspunkte als Lötflächen ausgelegt. Bei Be-

darf können auch Klemmleisten eingelötet werden, wie wir noch beim Nachbau sehen werden. Die mit dem Experimentierboard ermittelten Einstellungen können direkt auf die Anwenderschaltung übertragen werden. Hier sind die entsprechenden Lötbrücken (Jumper) zu setzen. Das Brücken (Kurzschließen) geschieht durch Auflöten einer Lötzinnbrücke, wie in Bild 17 dargestellt. Das Entfernen der Brücke kann mit Entlötlitze erfolgen.

Hier noch ein Hinweis bei Verwendung der Sirenschaltung: Ein kleiner Lautsprecher kann direkt zwischen den Klemmen „+UB“ und „OUT“ angeschlossen werden. Der vorhandene Widerstand auf der Platine sorgt für die notwendige Strombegrenzung bei sehr niederohmigen Lautsprechern. Die Lautstärke ist nicht sehr hoch, aber ausreichend. Wer es ger-

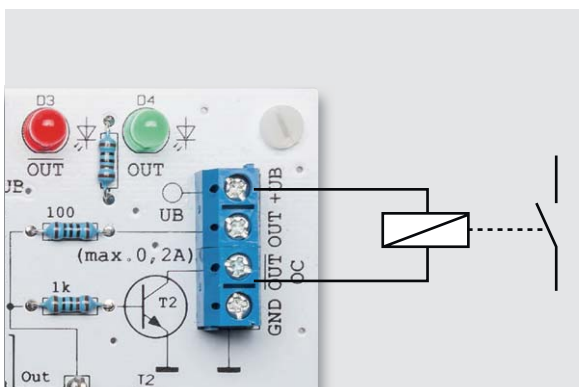


Bild 16: So wird ein Relais an den Ausgang „/OUT OC“ angeschlossen.

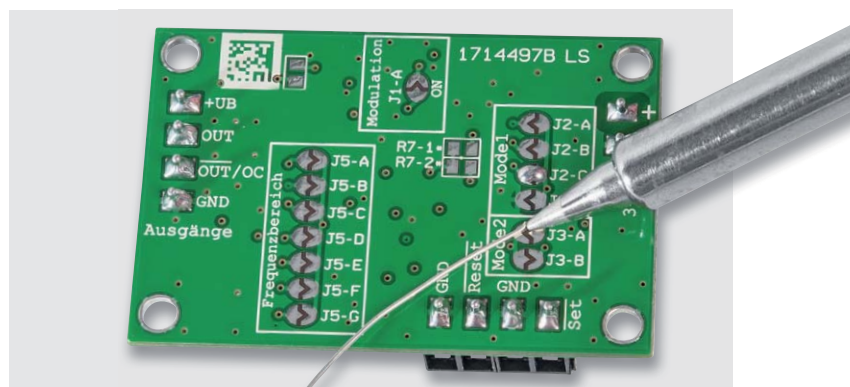


Bild 17: Das Aufbringen einer Kurzschlussbrücke mittels Auflöten

ne etwas lauter haben möchte, verwendet den Ausgang „/OUT OC“. Hier wird in den meisten Fällen noch ein Widerstand in Reihe benötigt, der zum einen die Ohren schützt und zum anderen den Strom auf max. 200 mA begrenzt. Bild 18 zeigt, wie dieser Lautsprecher und der Widerstand angeschlossen werden.

Nachbau NE555-EXB

Der Nachbau erfolgt auf vielfachen Kundenwunsch und wegen der besseren Übersichtlichkeit dieses Mal komplett mit bedrahteten Bauteilen. Dies steigert den Nachbauspaß, allerdings auch den Zeitaufwand im Vergleich zu einer vorbestückten SMD-Platine.

Für den Aufbau steht eine doppelseitige Platine mit den Abmessungen 100 x 160 mm zur Verfügung. Die Bestückung erfolgt anhand der Stückliste und des Bestückungsplans sowie des Platinenfotos (Bild 19). Wir beginnen mit der Bestückung der niedrigen Bauteile, gefolgt von den nächsthöheren Bauteilen. Die Reihenfolge der Bauteile nach der Bauhöhe zu gestalten hat den Vorteil, dass die bestückten Bauteile beim Umdrehen der Platine nicht so leicht herausfallen können. Zur Vorgehensweise beim Bestücken kann auch das Montagevideo [2] hilfreich sein.

Die Bauteile werden entsprechend dem Rastermaß abgewinkelt und durch die im Bestückungsdruck vorgegebenen Bohrungen geführt.

Nach dem Verlöten der Anschlüsse auf der Platinenunterseite (Lötseite) werden überstehende Drahtenden mit einem Seitenschneider sauber abgeschnitten, ohne die Lötstelle selbst dabei zu beschädigen. Beim Bestücken kann auch das Platinenfoto (Bild 19) hilfreich sein.

Beim Einsetzen der Halbleiter (ICs und Dioden) sowie der gepolten Kondensatoren (Elkos) ist auf die richtige Einbaulage bzw. die richtige Polung zu achten.

Die beiden Timerbausteine IC1 und IC2 werden nicht direkt eingelötet, sondern auf IC-Sockel gesetzt. Sowohl die Sockel als auch die beiden ICs haben zur Kennzeichnung der Einbaulage eine Einkerbung an einer Seite, die auch auf dem Platinenaufdruck erkennbar ist.

Die Dioden haben zur Kennzeichnung der Polarität eine Strichmarkierung, die die Katode kennzeichnet. Bei den Leuchtdioden (LEDs) D3 und D4 ist der Pluspol durch den etwas längeren Anschlussdraht ersichtlich. Bei den Elkos ist der Minuspol am Gehäuse bzw. der Pluspol durch den Anschlussdraht gekennzeichnet.

Stückliste NE555 EXB

Widerstände:

100 Ω	R12, R13
1 kΩ	R3, R8, R9, R14, R15
10 kΩ	R1, R2, R10, R11
22 kΩ	R4
56 kΩ	R6
PT15/liegend/1 MΩ	R5, R7

Kondensatoren:

47 pF/ker	C13
1 nF/100V/MKT	C12
10 nF/63 V/MKT	C11
100 nF/ker	C2-C4, C6, C14, C15
100 nF/63 V/MKT	C10
1 µF/63 V/MKT	C9, C16
10 µF/25 V	C1, C5, C8
100 µF/16 V	C7

Halbleiter:

ICM7555	IC1, IC2
BC558C	T1
BC548C	T2
SB120 (1N5817)	D1, D2
1N4148	D5-D8
LED/5 mm/rot	D3
LED/5 mm/grün	D4

Sonstiges:

Schraubklemmleisten, 2-polig, print	KL1-KL3
Schiebeschalter, 1x um	S1, S2, S4
Print-Schiebeschalter	S3
Mini-DIP-Schalter, 6-polig, liegend	S5
Mini-Drucktaster, 1x ein, print	TA1, TA2
Hohlsteckerbuchse, 2,1 mm, print	BU1
Rundsicherungshalter, print	SI1
Rundsicherung, 0,63 A, träge, print	SI1
Lötstifte mit Lötöse	MP1-MP5
Präzisions-IC-Fassungen, 8-polig	IC1, IC2
9 V-Batterieclip	
Distanzrollen mit Innengewinde, M3 x 10 mm, weiß	
Kunststoffschrauben, M3 x 6 mm	
Kunststoff-Steckachsen, ø 6 x 16,8 mm	
Aufsteckdrehknöpfe, ø 12 mm, schwarz	

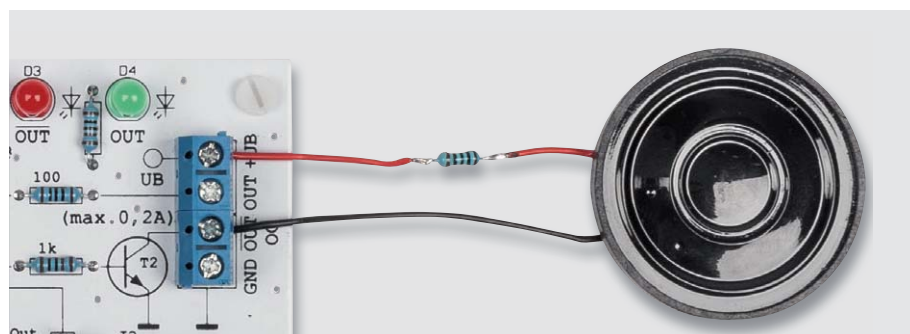


Bild 18: So wird ein Lautsprecher angeschlossen (der Zusatz-Widerstandswert sollte zwischen 22 und 100 Ω liegen).



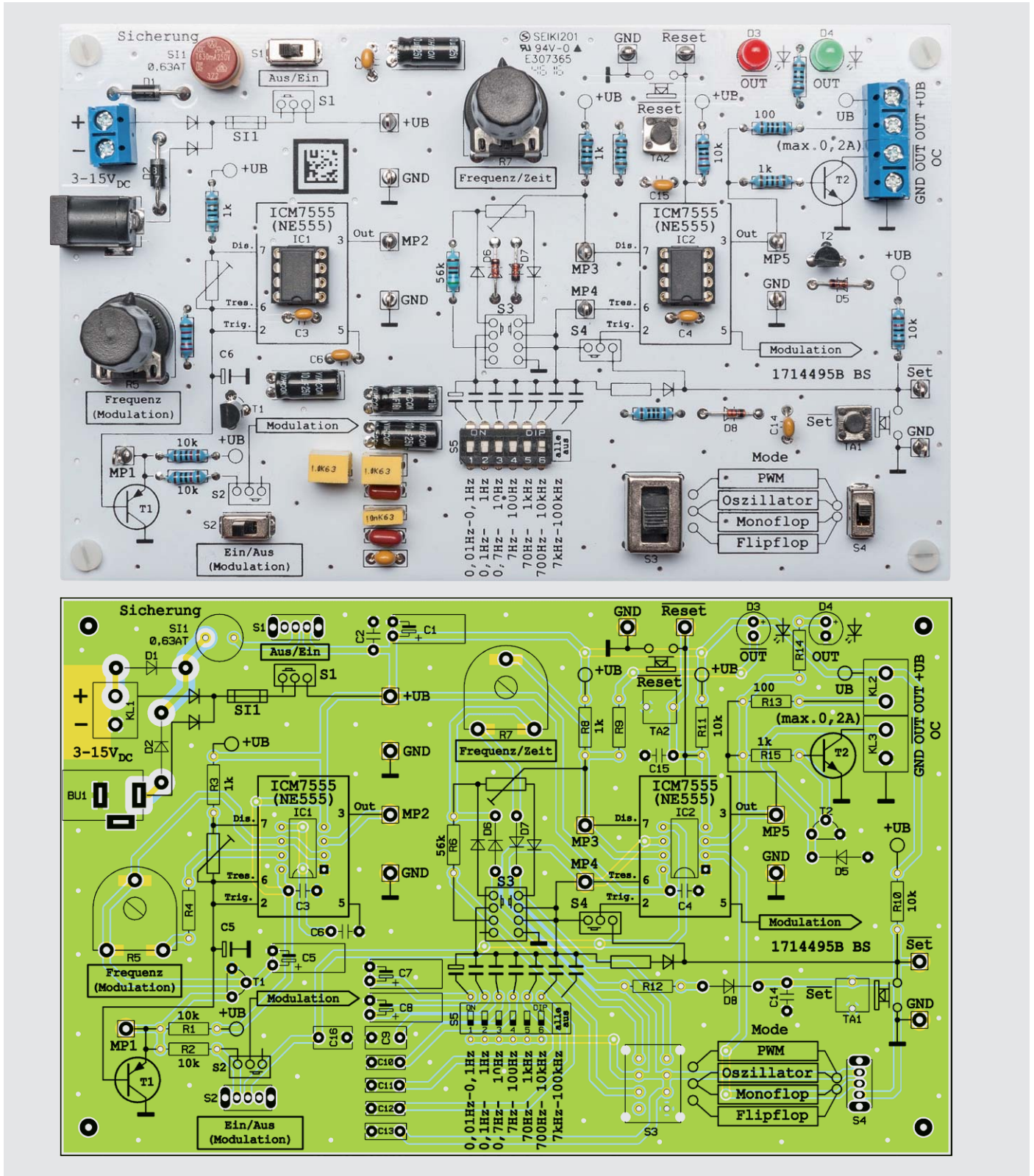


Bild 19: Platinenfoto und Bestückungsplan der Experimentierplatine NE555-EXB

Sind alle Bauteile so weit bestückt, sind noch die beiden Steckachsen mit Drehknopf (Bild 20) auf die Potis R5 und R7 aufzustecken.

Bevor der Drehknopf aufgesetzt wird, sollte die Achse auf Linksanschlag gedreht werden. Der kleine Zeiger vom Drehknopf wird dann so aufgesteckt, dass dieser auf den Linksanschlag zeigt. Ein erneutes Abnehmen und Wiederaufsetzen des Drehknopfs ist problemlos möglich.

Zum Schluss werden noch die vier Abstandshalter angeschraubt. Diese Kunststoffabstandshalter besitzen ein durchgängiges Innengewinde. Von oben ist jeweils eine Kunststoffschraube zur Befestigung einzuschrauben (siehe Bild 21).

Nachbau NE555-AWS

Die Anwenderschaltung NE555-AWS ist im Gegensatz zum Experimentierboard auf kompakte Abmessungen optimiert. Alle Bauteile sind in SMD-Technik ausgeführt und werkseitig bestückt. Eine manuelle Bestückung ist nicht notwendig, es sollte lediglich anhand Bild 22 eine Kontrolle auf exakte Bestückung und Lötfehler vorgenommen werden.

Die Platine ist als Multilayer-Platine mit vier Lagen ausgeführt und bietet durch eine großflächige



Bild 20: Drehknopf und Steckachse



Bild 21: Seitliche Ansicht des Abstandshalters

Massefläche guten Schutz vor Störeinstrahlung. Die Platine ist für die Anwendung in eigene Schaltungen bzw. Geräte gedacht. Bei Bedarf können die Anschlusspunkte mit Klemmleisten bestückt werden (siehe Bild 23), die nicht im Bausatz enthalten sind. Die SMD-Trimmer sollten vorsichtig behandelt werden, zum Verstellen sollten nur genau passende Schraubendreher verwendet werden. **ELV**

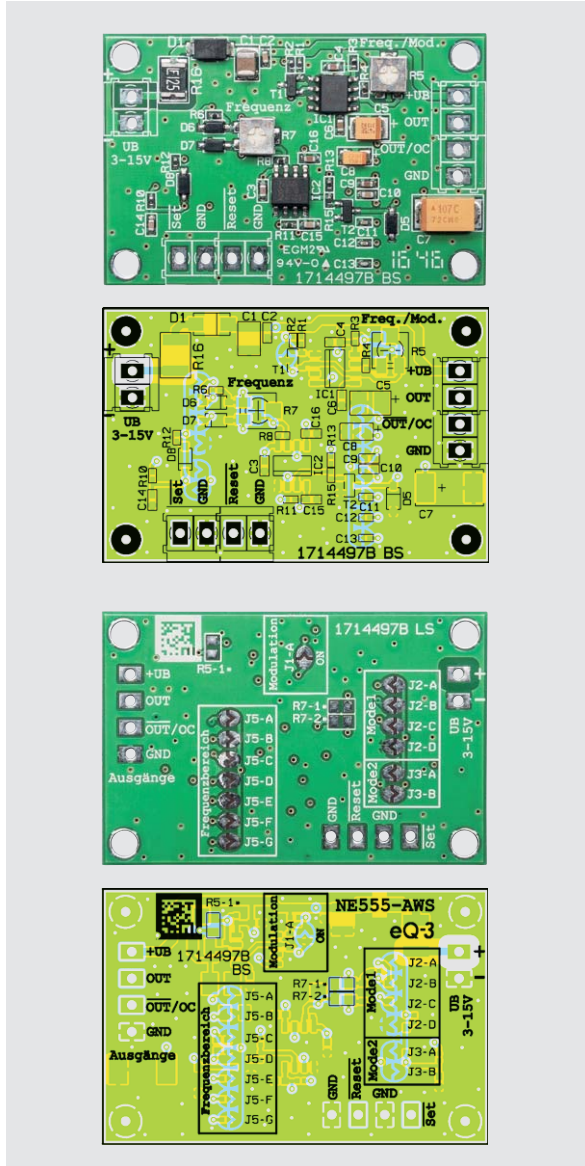


Bild 22: Platinfotos und Bestückungspläne der Anwendungsschaltung NE555-AWS, oben die Oberseite, unten die Unterseite der Platine

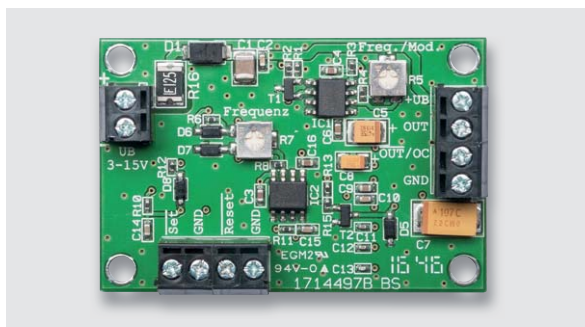


Bild 23: NE555-AWS mit Klemmleisten

Stückliste NE555-AWS

Widerstände:

100 Ω/SMD/0402	R12, R13
1 kΩ/SMD/0402	R3, R8, R15
10 kΩ/SMD/0402	R1, R2, R10, R11
22 kΩ/SMD/0402	R4
56 kΩ/SMD/0402	R6
Trimmer/1 MΩ/SMD	R5, R7
PTC/1,25 A/16 V/SMD/1812	R16

Kondensatoren:

47 pF/50 V/SMD/0402	C13
1 nF/50 V/SMD/0402	C12
10 nF/50 V/SMD/0402	C11
100 nF/50 V/SMD/0603	C2, C10, C14, C15
1 µF/16 V/SMD/0402	C3, C4, C6, C9, C16
4,7 µF/20 V/Tantal/SMD	C5
10 µF/10 V	C8
10 µF/50 V/SMD/1210	C1
100 µF/16 V/SMD/Tantal	C7

Halbleiter:

ICM7555/SMD	IC1, IC2
BC857C/SMD	T1
FMMT618/SMD	T2
SK14/SMD	D1
1N4148W/SMD	D5-D8

Weitere Infos:

[1] Datenblatt NE555/ICM7555:
www.nxp.com/documents/data_sheet/ICM7555.pdf
www.ti.com/lit/ds/symlink/ne555.pdf

[2] www.elv.de: Webcode #10055

[3] <https://de.wikipedia.org/wiki/Flipflop>

[4] <https://de.wikipedia.org/wiki/NE555>



Kompakt umgesetzt

Mini-USB-UART-Umsetzer

Infos zum Bausatz

im ELV Shop

#10054

Für die Kommunikation, das Debuggen oder die Programmierung von Mikrocontrollern wird von vielen Entwicklern häufig die UART-Schnittstelle des Mikrocontrollers verwendet. Da auf den heutigen Computern meist nur noch USB-Ports vorhanden sind, bedarf es eines Adapters, der die beiden Schnittstellenarten zueinander kompatibel macht. Mit dem neuen USB-UART-Umsetzer UM2102N lösen wir dieses Problem und ersetzen gleichzeitig das schon etwas in die Jahre gekommene Vorgängergerät.

Allgemeines

Der als Nachfolger des UM2102 konzipierte UM2102N setzt auf den neuen Schnittstellenwandler CP2102N der Firma Silicon Laboratories auf und bringt somit alle Vorteile des neuen Chips mit. In seinen Abmaßen ist der neue Umsetzer identisch mit dem Vorgänger und lässt sich somit wieder einfach in eigene Schaltungslayouts integrieren.

Heutzutage ist ein USB-RS232-Umsetzer kein Hightech-Produkt mehr, im (Internet-)Elektronikhandel sind sie zahlreich zu bekommen. Mit dem praktischen Umsetzerkabel oder Zwischenstecker (Bild 1) kann man einen Computer so sehr einfach mit einer EIA/RS232-Schnittstelle nachrüsten. Es ist dann aber noch ein RS232-UART-Wandler notwendig, um mit einem Mikrocontroller zu kommunizieren.

Für eigene Elektronikentwicklungen werden aber auch gern USB-UART-Module eingesetzt, da sie flexibler in der Handhabung sind und man sich den Umweg über die RS232-Schnittstelle spart. Diese Module sind jedoch häufig zu groß oder zu teuer, um sie einfach und häufig einzusetzen. Unter diesen

Aspekten hat ELV im Jahr 2010 das Mini-USB-Modul UM2102 entwickelt und optimiert. Auf einer Fläche von nur ca. 5 cm² bietet es diverse Steuersignale, zwei Ausgangsspannungen und vielfältige Anschlussmöglichkeiten. Dies ist nun ein paar Jahre her und der Chip-Hersteller des USB-UART-Umsetzers hat zwischenzeitlich einen Nachfolge-Chip entwickelt. Aus diesem Grund wird auch das UM2102 mit diesem neuen Chip aktualisiert.

Mini-USB-Modul UM2102N

Mit dem Nachfolgemodul UM2102N können identisch zum Vorgänger serielle Datenverbindungen zwischen einem Computer und einer elektronischen



Bild 1: Für PC-Betrieb ideal, für eigene Applikationen nicht flexibel genug – ein handelsüblicher USB-zu-RS232-Adapter

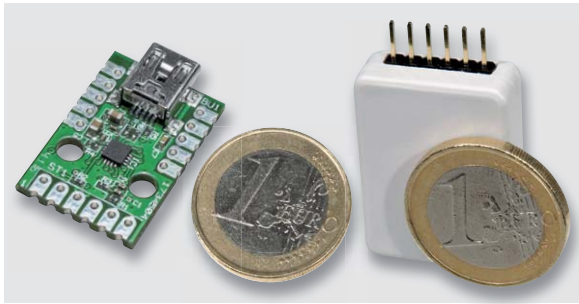


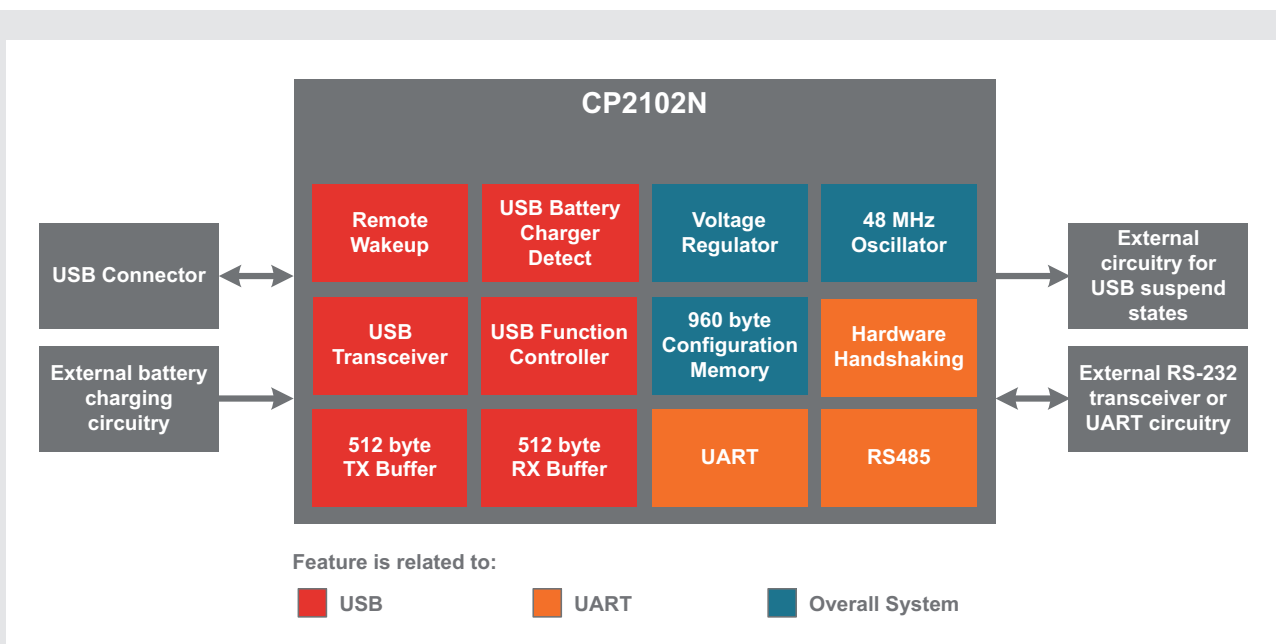
Bild 2: Größenvergleich des UM2102N mit einer 1-Euro-Münze, rechts als einfach ansteckbares Modul mit passendem Gehäuse

Schaltung über eine USB-Verbindung realisiert werden, ohne dass Änderungen an bestehender Software notwendig werden. Durch einen auf dem PC installierten VCP-Treiber (VCP: Virtual Com Port) wird ein echter COM-Port simuliert und im Gerätemanager des Betriebssystems eingerichtet.

Das Modul besteht im Wesentlichen aus dem Wandler-Chip CP2102N von Silicon Laboratories [1], einem USB-zu-UART-Umsetzer, der als Ein-Chip-Lösung konzipiert wurde. Dadurch ist keine aufwendige Peripherie notwendig und das Modul konnte sehr kompakt gehalten werden. Einem vielseitigen Einsatz

Technische Daten

Geräte-Kurzbezeichnung:	UM2102N
Schnittstelle:	USB 2.0
Übertragungsrate:	300–3.000.000 Baud
Treiber:	Virtueller COM-Port (VCP): Windows 2000/XP/Server 2003/Vista Windows 7/8/8.1/10 WinCE, Mac OS X, Linux, Android USB-Direkt-Treiber USBXpress: Windows 98SE/Windows 2000/XP/Server 2003/Vista Windows 7/8/8.1/10 WinCE
Versorgungsspannung:	USB powered
Spannungsausgang	5 V/max. 500 mA 3,3 V/max. 100 mA
Signale:	RXD, TXD, RTS, CTS
Signalpegel:	RXD, CTS, WAKEUP, GPIO.x als Eingang: 3,3 V (5-V-tolerant) TXD, RTS, /Suspend, Suspend, GPIO.x als Ausgang, TXT, RXT: 3,3 V
Kontakte:	16, auf 2,54-mm-Raster
Umgebungstemperatur:	5 bis 35 °C
Abmessungen (B x H x T):	28 x 18 x 6 mm
Abmessungen Gehäuse (B x H x T):	30 x 22 x 9 mm
Gewicht:	2 g



USB-zu-UART-Umsetzer CP2102N

Der CP2102N ist ein USB-zu-UART-Umsetzer (USB to UART Bridge), mit dem Mikrocontroller die weitverbreitete USB-Schnittstelle zur Kommunikation mit einem PC verwenden können. Auch heute wird dafür noch gern die RS232-Schnittstelle eingesetzt, da sie einfach zu nutzen ist. Moderne PCs und Notebooks haben aber in der Regel keine COM-Schnittstelle mehr, jedoch genügend USB-Ports.

Der CP2102N schließt diese Lücke, indem er sich auf der Mikrocontrollerseite wie ein UART-Baustein, PC-seitig aber wie ein USB-Gerät verhält. Mit dem passenden Treiber wird auf dem PC ein virtueller COM-Port eingerichtet, der sich wie ein echter COM-Port verhält. Somit können PC-Anwendung und Mikrocontroller auf einfache Weise über USB kommunizieren und

bestehende Programme weitgehend unverändert weiterbenutzt werden.

Im Gegensatz zu anderen USB-UART-Lösungen benötigt der CP2102N fast keine externe Beschaltung. In den Chip ist neben USB-Transceiver, Takterzeugung und Konfigurationsspeicher auch noch ein Spannungsregler integriert. Der CP2102N kann sich daher aus der Busspannung selbst versorgen und stellt zudem eine Spannung von 3,3 V für andere Schaltungsteile zur Verfügung. Kleine Schaltungen mit einer Stromaufnahme von maximal 100 mA können so ganz einfach ohne zusätzliche Spannungsregler versorgt werden.



steht daher auch dank des intelligenten Layouts nichts im Wege. Das Modul kann sowohl direkt über doppelseitige Lötflächen, Drahtstücke, Leitungen als auch über beidseitig einlötbare Stiftleisten an andere Schaltungen angebunden werden.

Bei Bedarf kann das Modul externe Schaltungskomponenten direkt versorgen, dafür stehen die USB-Spannung (5 V, max. 500 mA) und eine geregelte 3,3-V-Spannung (max. 100 mA) zur Verfügung.

Bild 2 zeigt das Modul im Größenvergleich mit einer 1-Euro-Münze.

Welcher Treiber?

Silicon Laboratories bietet für die CP210x-Serie VCP-Treiber für verschiedene Betriebssysteme an [2]. Ebenfalls verfügbar ist ein Direkttreiber namens USBXpress für verschiedene Windows-Betriebssysteme [3]. Der VCP-Treiber ermöglicht eine einfache Softwareimplementierung, da der Zugriff auf einen COM-Port recht simpel ist und fertige Bibliotheken im Internet erhältlich sind. Auch können mehrere Module problemlos parallel an einem Rechner eingesetzt werden, jedes Modul bekommt dann einen eigenen COM-Port zugewiesen.

Mit dem USBXpress-Treiber ist eine direkte Kommunikation mit dem UM2102N möglich, ohne den Umweg über einen virtuellen COM-Port. Die Implementierung ist aufwendiger, da die Schnittstellen des Treibers genutzt werden müssen. Eine Dokumentation (AN169) ist bei Silicon Laboratories verfügbar [4]. Man erhält damit auch Zugriff auf die im internen Konfigurationsspeicher des CP2102N abgelegten Daten wie VID (Herstellereerkennung), PID (Produktkennung), Seriennummer und Produktname und kann damit das Modul eindeutig identifizieren.

Hinweis:

Damit an einem Computer mehrere UM2102N gleichzeitig verwendet werden können, verfügen die Module über eine herstellerseitige individuelle Seriennummer. Anhand dieser Seriennummer können Geräte mit derselben VID-/PID-Kombination für den VCP-Treiber trotzdem eindeutig identifiziert werden.

Bei Verwendung des USBXpress-Treibers muss vorher noch die PID des Moduls geändert werden, dazu ist das Programm ID-Changer notwendig, das auf der Produktseite im Online-Shop erhältlich ist. Eine genaue Beschreibung des Programms steht dort ebenfalls zur Verfügung.

Der CP2102N

Der USB-Function-Controller erledigt den Datentransfer zwischen UART-Einheit und USB-Transceiver, kümmert sich um das Protokollhandling und kontrolliert die UART-Einstellungen sowie den Reset-Eingang und die Suspend-Ausgänge (siehe Blockschaltbild im Kasten „Elektronikwissen“). Für die Umsetzung der Daten in elektrische Signale und umgekehrt ist der USB-Transceiver zuständig. Die Empfangs- und Sendedaten werden in den jeweiligen Puffern zwischengespeichert, bis sie an den Computer oder den angeschlossenen Mikrocontroller gesendet werden können. So können auch Verzögerungen bei der Datenverarbeitung ausgeglichen werden.

Im Konfigurationsspeicher werden die Gerätedaten wie VID, PID, Seriennummer usw. abgelegt, die bei der Anmeldung des UM2102N-Moduls am PC notwendig sind. Da auch noch ein 48-MHz-Oszillator und ein 3,3-V-Spannungsregler integriert sind, kann auf externe Komponenten komplett verzichtet werden.

Der Spannungsregler ermöglicht einen direkten Betrieb über die USB-Spannung von 5 V, obwohl die Versorgungsspannung 3,3 V beträgt. Optional ist auch eine Versorgung mit 3,3 V möglich. Praktisch ist die Tatsache, dass die Ausgangsspannung des Reglers am Pin +UB verfügbar ist. Die High-Pegel der Ausgänge betragen 3,3 V, die Eingänge sind aber 5-V-tolerant.

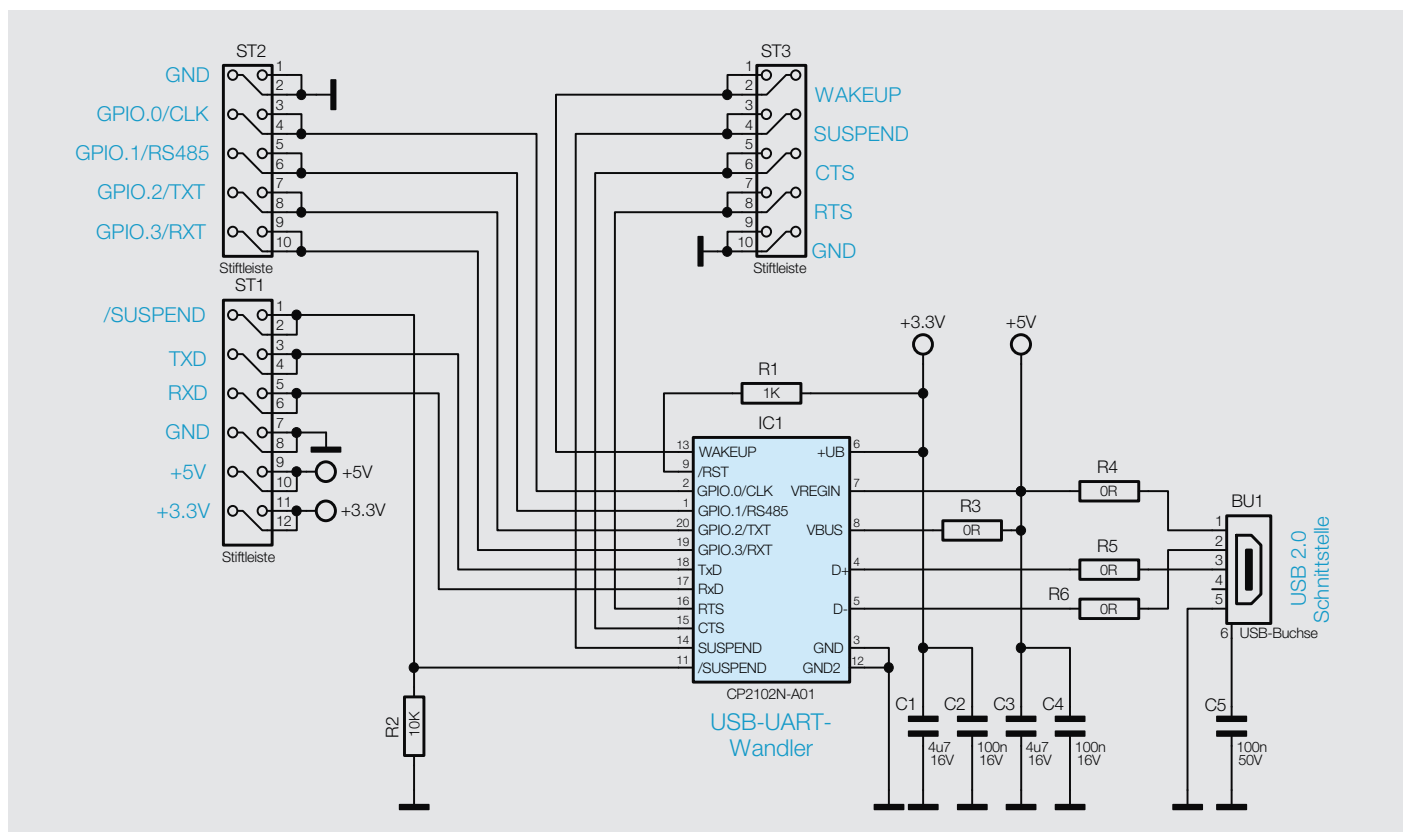


Bild 3: Die Schaltung des UM2102N

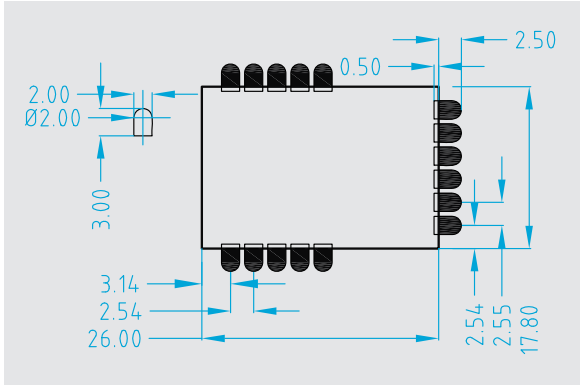


Bild 4: Layoutvorgabe für das UM2102N (Maßangaben in mm)

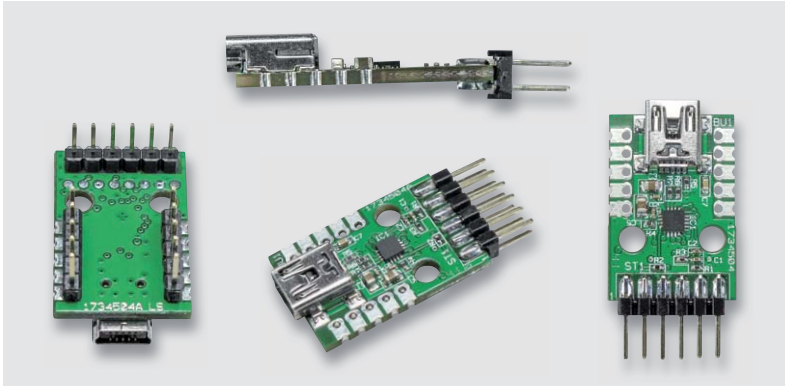


Bild 5: Das Modul kann je nach Bedarf mit Stiftleisten bestückt werden.

Der CP2102N bietet die Möglichkeit, einige Pins als allgemeine Ein- oder Ausgänge zu nutzen. Mithilfe einer passenden Software können so die aktuellen Pegelzustände als Eingang ausgelesen oder als Ausgang gesetzt werden.

Schaltung

Beginnen wir mit der Schaltungsbeschreibung. In Bild 3 ist die komplette Schaltung des UM2102N dargestellt. Wie hier zu sehen, ist der Schaltungsaufwand sehr gering und beschränkt sich auf das IC1 (CP2102N) samt Beschaltung und die USB-Buchse BU1.

Die Kondensatoren C3 und C4 dienen der Störunterdrückung auf der USB-Spannungsleitung. Für die 3,3 V des integrierten Spannungsreglers dienen C1 und C2 als Puffer- und Entstörkondensatoren. Der Widerstand R1 zieht den Reset-Pin des CP2102N definiert auf die intern erzeugten 3,3 V.

Das Suspend-Signal zeigt den aktuellen Betriebszustand des CP2102N an, also ob der Chip im aktiven Betrieb ist oder sich im Stand-by befindet. Das Suspend-Signal kann zum einen auf der Stiftleiste ST3 und in der invertierten Variante (/SUSPEND) auf ST1 ausgelesen werden. Im aktiven Betrieb liegt an dem Suspend-Pin auf ST3 ein Low-Signal an, nach einem erkannten Suspend-Befehl wechselt es auf ein High-Signal.

Der Suspend-Befehl wird vom Betriebssystem gesendet, wenn ein USB-Gerät in den Stand-by-Modus wechseln soll. Während eines Resets sorgt R2 für einen definierten Signalpegel.

Die RS232-Steuersignale RTS und CTS für das „Hardware-Handshaking“ werden vom CP2102N unterstützt und sind als Kontakte auf der Stiftleiste ST3 ausgeführt.

An der Stiftleiste ST2 befinden sich die Kontakte zu den als Ein- oder Ausgang nutzbaren Pins. Alle Kontakte liegen auf einem 2,54-mm-Raster, sodass

die Platine auch problemlos in einem Lochrasteraufbau einsetzbar ist. Die für die eigentliche UART-Kommunikation benötigten Signalleitungen zum Senden (TXD) und zum Empfangen (RXD) von Daten sind auf der Stiftleiste ST1 untergebracht. Zusätzlich stehen hier auch die 5-V-Busspannung und die 3,3-V-Spannung des internen CP2102N-Spannungsreglers zur Verfügung.

Nachbau

Der Nachbau der eigentlichen Platine entfällt, da alle Bauteile als SMD ausgeführt sind und bereits werkseitig bestückt werden. Die Schaltung kann daher direkt eingesetzt werden.

Optional können ST1, ST2 und ST3 mit Stiftleisten bestückt werden (siehe Einsatzmöglichkeiten).

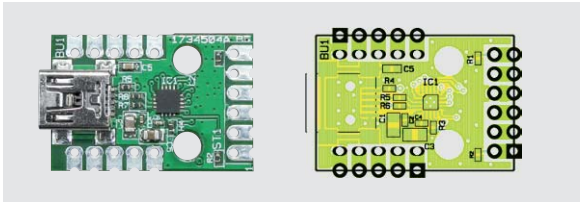
Die Inbetriebnahme des UM2102N beschränkt sich auf das Verbinden des Moduls mit einem Computer und die Installation des zuständigen Treibers. Wird das Modul vom angeschlossenen Rechner korrekt erkannt, so kann davon ausgegangen werden, dass auch das gesamte Modul ordnungsgemäß funktioniert. Die komplette Funktion lässt sich allerdings erst prüfen, wenn das Modul im Zielsystem eingesetzt ist und auch die gesamte Kommunikationsstrecke einem Test unterzogen werden kann. In der Regel reicht es aus, TXD, RXD und Masse des UM2102N mit einem Mikrocontroller zu verbinden, um eine Kommunikation zu ermöglichen.

Achtung!
Das UM2102N darf nur direkt auf eine andere Leiterplatte gelötet oder über Stiftleisten (siehe Bild 5 bis Bild 8) verbunden werden.

Um einen ausreichenden ESD-Schutz zu gewährleisten, muss die Platine in ein Gehäuse eingebaut werden. Das kann das Gehäuse eines anderen Gerätes oder aber auch das weiter unten erwähnte Gehäuse DRO3 sein.

Wichtiger Hinweis zum ESD-Schutz:

Bei den verwendeten Bauteilen des Mini-USB-UART-Moduls UM2102N handelt es sich um elektrostatisch gefährdete Bauteile. Das bedeutet, dass sie bereits durch bloßes Anfassen, z. B. beim Einbau oder im späteren Betrieb, zerstört werden können, sofern man vorher elektrisch geladen war, was beispielsweise durch Laufen über Teppiche passieren kann. Vor dem Handhaben bzw. dem Berühren dieser Bauteile ist es daher ratsam, Maßnahmen anzuwenden, die einen entsprechenden Schutz vor elektrostatischen Entladungen an diesen Bauteilen ermöglichen. Hierzu kann man sich z. B. mit einem Erdungsband erden oder zumindest ein Metallgehäuse eines Gerätes oder die Heizung anfassen.



Ansicht der fertig bestückten Platine des UM2102N mit zugehörigem Bestückungsplan



Einsatzmöglichkeiten

Die Kontakte des Moduls sind so ausgeführt, dass man es auf unterschiedliche Weise in eigene Schaltungen integrieren kann:

Auflöten auf die Platine

Das Modul kann direkt auf eine eigene Leiterplatte gelötet werden, wodurch eine stabile mechanische und elektrisch sichere Verbindung ermöglicht wird. Das CP2102-Modul kann dafür am einfachsten als Bauteil im CAD-Programm angelegt und dann ins Schaltbild aufgenommen werden. Bild 4 zeigt eine Maßzeichnung („Land Pattern“) des Moduls, die als Vorlage für ein eigenes Layout genutzt werden kann.

Verwendung von Stiftleisten

Die Kontakte des Moduls sind im 2,54-mm-Raster angeordnet und die Bohrungen nehmen ein- oder zweireihige Stift- oder Buchsenleisten auf, wie es in Bild 5 zu sehen ist. Durch diese Variationen kann das UM2102N genauso flexibel wie das Vorgängermodul eingesetzt werden, indem es je nach Bedarf in die eigenen Schaltungen gesteckt wird. Bild 6 zeigt nur einige dieser Möglichkeiten, wie sie beim Vorgängermodul umgesetzt wurden. In der Entwicklungsphase ist diese Art der Anbindung vorteilhaft, da das Modul bei einem Defekt der Applikationsleiterplatte problemlos wieder entfernt werden kann. Die Belegung der Stiftleiste ST1 vom UM2102N ist identisch mit der Belegung der Stiftleiste ST2 beim Vorgänger. Somit ist ein direkter Austausch der beiden Modulvarianten gewährleistet.

Einsatz im Gehäuse

Optional kann man das Modul auch in das Gehäuse DR03 (Best.-Nr. CO-07 60 12) einbauen (siehe Bild 7). Dafür wird ST1 mit einer abgewinkelten Stiftleiste bestückt, und es müssen Aussparungen für die USB-Buchse und die Stiftleiste in das Gehäuse gefeilt bzw. gefräst werden. So kann das Modul wie ein eigenständiger USB-UART-Umsetzer benutzt, aber auch schnell als optionale Debug-Schnittstelle zum Aufstecken verwendet werden.

Bild 8 zeigt diesen Einsatz. Ein derart ausgeführtes Modul ist auch der ideale Helfer bei der Entwicklung eigener Applikationen. **ELV**

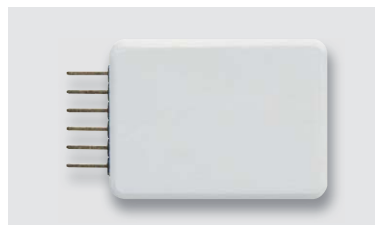


Bild 7: UM2102N im bearbeiteten Gehäuse

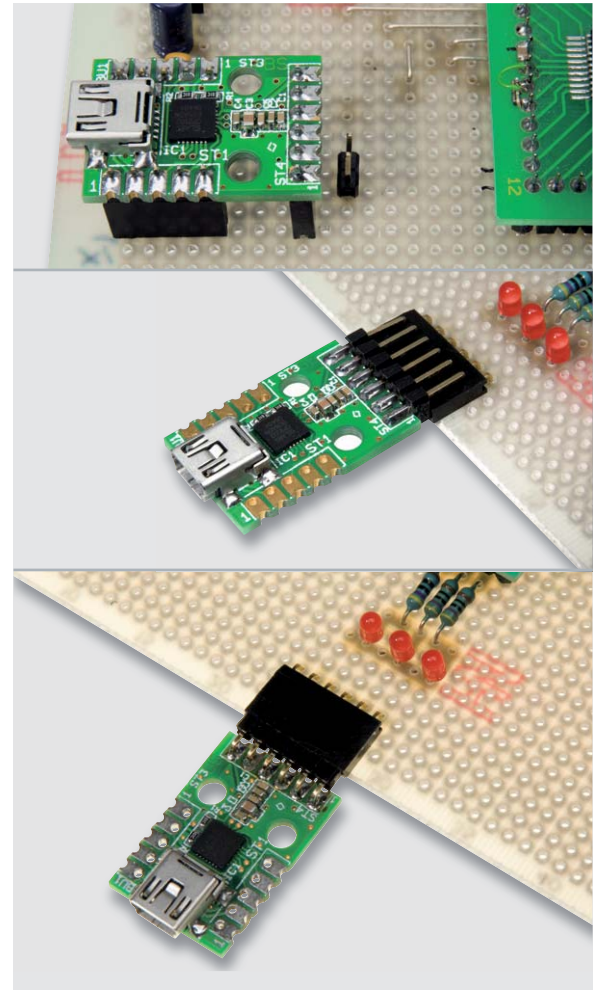


Bild 6: Das UM2102N lässt sich identisch in die eigene Applikation sehr flexibel einbinden bzw. an diese anschließen, wie das oben dargestellte Vorgängermodul.

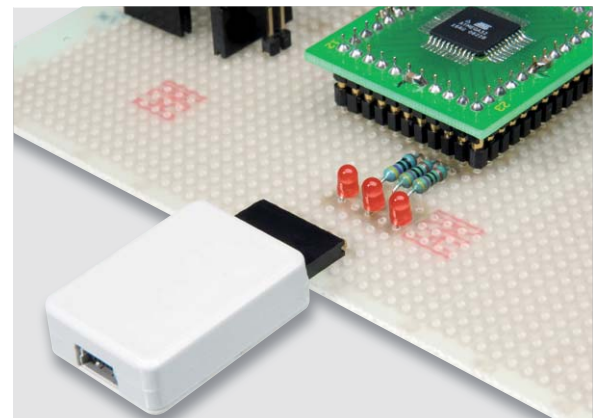


Bild 8: Auch im Gehäuse lässt sich das Modul noch flexibel einsetzen.

Widerstände:

0 Ω /SMD/0402	R3–R6
1 k Ω /SMD/0402	R1
10 k Ω /SMD/0402	R2

Kondensatoren:

100 nF/16 V/SMD/0402	C2, C4
100 nF/50 V/SMD/0603	C5
4,7 μ F/16 V/SMD/0805	C1, C3

Halbleiter:

CP2102N	IC1
---------	-----

Sonstiges:

USB-Buchse, Mini B, SMD	BU1
-------------------------	-----

Stückliste



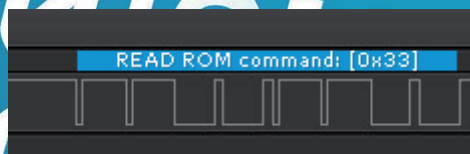
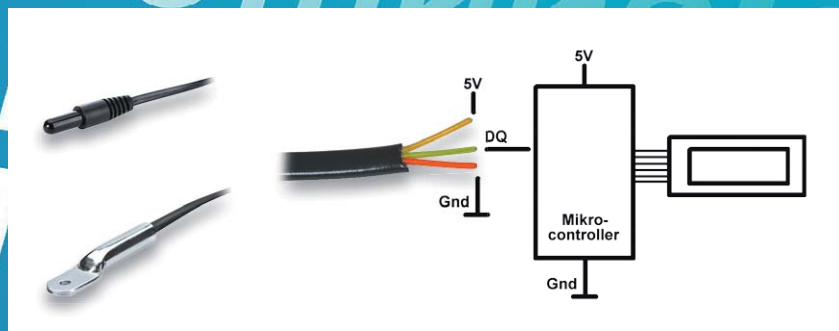
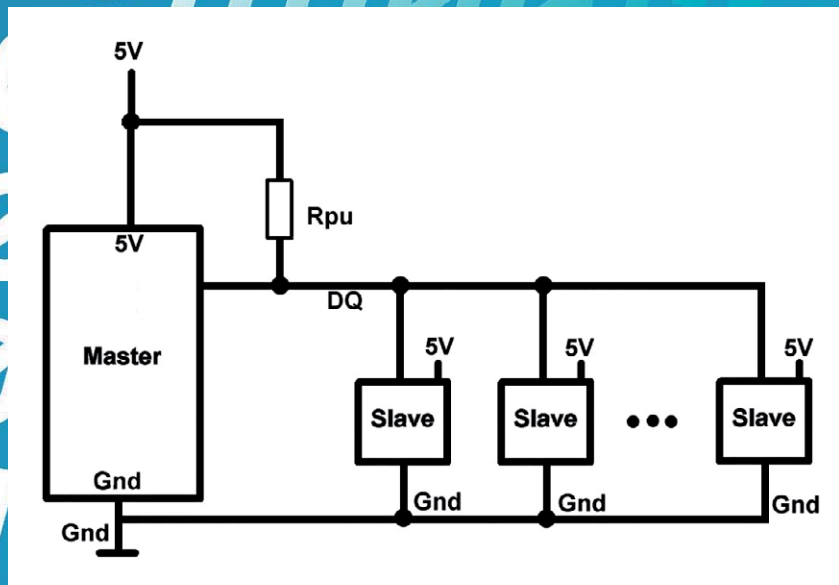
Weitere Infos:

- [1] www.silabs.com
- [2] www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers
- [3] www.silabs.com/products/development-tools/software/direct-access-drivers
- [4] www.silabs.com/documents/public/application-notes/AN169.pdf

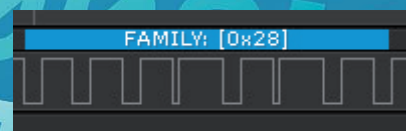


Digitale Hardwarechnittstellen

Teil 4: 1-Wire



RESET condition
 PRESENCE condition
 READ ROM command: [0x33]
 FAMILY CODE section from ROM: [0x28]
 ROM CODE section from ROM: [0x00000...]
 CRC section from ROM: [0xEC]





Die 1-Wire-Schnittstelle wurde von der Firma Maxim Integrated Products (vormals Dallas Semiconductor) entwickelt und definiert sowohl den Busaufbau als auch das Protokoll. Es handelt sich um eine asynchrone serielle Verbindung zwischen einem Master und nahezu beliebig vielen Slaves. Der Name 1-Wire wurde gewählt, weil die Datenverbindung mit lediglich einer Datenleitung (zusätzlich zur Gnd-Verbindung) auskommt. Von einer seriellen Verbindung wird gesprochen, weil die Bits hintereinander übertragen werden. Die Übertragung heißt asynchron, weil es keine separate Taktleitung zwischen dem Master und dem Slave oder den Slaves gibt, sondern die Kommunikation zwischen Master und Slave(s) durch eine detaillierte Spezifikation der zeitlichen Abläufe geregelt wird. Die Verbindung erfolgt bidirektional, das heißt, dass Daten sowohl vom Master zum Slave als auch umgekehrt übertragen werden können.

Es gibt eine große Anzahl verschiedener Slaves für unterschiedliche Zwecke. Am weitesten verbreitet sind die Temperatursensoren DS1820 bzw. dessen Nachfolger DS18S20 und der aktuellere DS18B20 mit der Möglichkeit, die Auflösung zu konfigurieren und damit die Messzeit zu verkürzen. Der DS18S20 kann als direkter Ersatz des ursprünglichen DS1820 verwendet werden, während der modernere DS18B20 in neuen Entwicklungen bevorzugt werden sollte. Beide Temperatursensoren (DS1820/DS18S20 und DS18B20) müssen unterschiedlich ausgewertet werden, aber da sie per Software anhand unterschiedlicher Kennungen (Family-Codes – siehe unten) unterscheidbar sind, ist auch ein gemischter Betrieb an einem Bus leicht möglich. Unterschiede zwischen den Sensoren sind in [1] beschrieben.

Außer den genannten Temperatursensoren gibt es für 1-Wire auch adressierbare elektronische Schalter (DS2413, DS2408), digitale Potenziometer (DS1805), Analog-Digital-Konverter (DS2450), 4k-RAM mit Zähler (DS2423), Batteriemonitor (DS2438), Echtzeituhren (DS2417), Speicher-ICs (DS24B33, DS1961), weitere Temperaturbausteine (DS28EA00, DS1921), eindeutige Silicon-Seriennummer (DS2401) und so weiter. Einige der 1-Wire-Slaves werden in einem Gehäuse angeboten, das einer Knopfzelle sehr ähnlich sieht – sogenannte iButtons –, bei denen Ober- und Unterseite je einen Kontakt zu dem eingebauten Baustein darstellt. Die meisten Bausteine sehen allerdings aus wie ein Transistor (meist im TO-92- oder SOT-223-Gehäuse) oder werden als Temperatursensoren inklusive angeschlossener Kabel und in speziellen Gehäusen angeboten.

Busaufbau

Wie Bild 1 zeigt, besteht ein 1-Wire-Bus aus genau einem Master, der auch stets die Kommunikation anstößt, und einem oder vielen Slaves. Der Master und die Slaves müssen eine gemeinsame Gnd-Leitung haben. Die Datenübertragung erfolgt auf einer Leitung mit der Bezeichnung DQ, an die jeder Slave angeschlossen ist. Für den gesamten Bus muss es genau einen Pull-up-Widerstand R_{pu} von ca. 4,7 k Ω geben, der die Datenleitung nach Plus (V_{cc}) zieht. Jeder einzelne 1-Wire-Teilnehmer

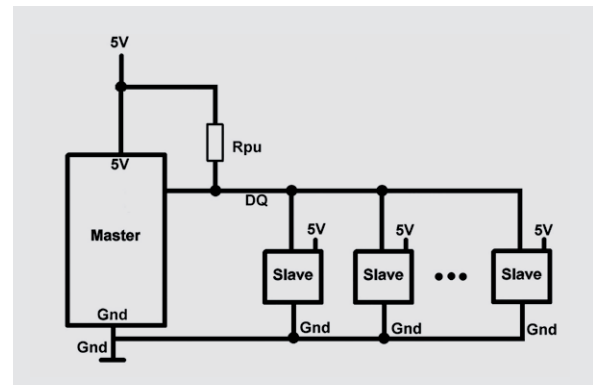


Bild 1: 1-Wire-Bus mit externer Spannungsversorgung

ist – wie in Bild 2 schematisch dargestellt – mit einer Open-Drain-Schaltung aufgebaut. Wenn ein 1-Wire-Baustein einen Spannungszustand auf die Datenleitung geben soll, zieht er den Spannungspegel entweder nach Gnd, indem der eingebaute MOSFET leitend geschaltet wird, oder er lässt die Datenleitung durch den Pull-up-Widerstand nach Plus (V_{cc}) ziehen. Zum Erkennen der an der Datenleitung anliegenden Spannung „lauscht“ der 1-Wire-Baustein (mit Rx, Bild 2) auf den Bus. Der 1-Wire-Bus kann auf jeden Fall 20 bis 30 Meter lang sein. Manchmal liest man sogar von bis zu 200 Metern.

Die meisten 1-Wire-Bausteine ermöglichen eine Spannungsversorgung über eine dedizierte Leitung für die Betriebsspannung (typisch 5 Volt oder 3,3 Volt) oder aber alternativ eine sogenannte parasitäre Spannungsversorgung, bei der es keine explizite Spannungsversorgung gibt, sondern die Spannung aus der Datenleitung gewonnen wird, indem ein Kon-

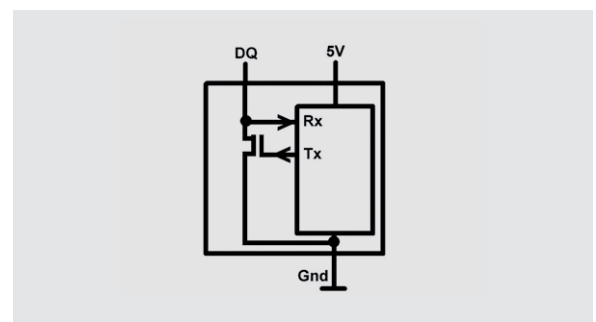


Bild 2: 1-Wire-Slave mit Open Drain



8-BIT CRC	48-BIT SERIAL NUMBER	8-BIT FAMILY CODE (28h)
MSB		LSB

Bild 3: 64-Bit ROM-Code

densator im Innern des Slaves sich während der Ruhezeiten (Spannung auf der Datenleitung durch den Pull-up-Widerstand auf Vcc) auflädt und die gespeicherte Spannung für den Betrieb wieder abgibt.

Bei externer Spannungsversorgung benötigt man für die Spannungsversorgung zwei Leitungen und für die Datenübertragung eine dritte Leitung. Bei einer parasitären Spannungsversorgung benötigt man außer einer Gnd-Leitung nur die Datenleitung, kommt also insgesamt mit nur zwei Leitungen aus – was auch die Bauform als iButtons ermöglicht. Bei der Bezeichnung 1-Wire wird die Gnd-Leitung nicht mitgezählt.

Adressierung

Jeder Slave am 1-Wire-Bus hat eine 64 Bit (8 Byte) lange ID – den sogenannten ROM-Code (Bild 3). 8 Bit des ROM-Codes sind der sogenannte Family-Code (Gerätfamilie), der den Typ des 1-Wire-Bausteins darstellt (zum Beispiel 28h für einen DS18B20 und 10h für einen DS18S20 oder DS1820) und per Programm abgefragt und ausgewertet werden kann. Dadurch kann man beispielsweise verschiedenartige Temperatursensoren an einem 1-Wire-Bus haben und im Programm die jeweils passende Umwandlungsroutine einsetzen. Die nächsten 48 Bit des ROM-Codes sind eine für den betreffenden Baustein weltweit eindeutige Identifikationsnummer. Mit diesen 48 Bit lassen sich $2 \text{ hoch } 48 = 281.474.976.710.656$ verschiedene 1-Wire-Bausteine eindeutig identifizieren!

Anders als zum Beispiel bei der I²C-Schnittstelle, wo teilweise nur 8 gleichartige Bausteine an einem Bus angeschlossen sein können, ist es dadurch bei 1-Wire problemlos möglich, zum Beispiel 30 gleichartige Temperatursensoren an einem 1-Wire-Bus angeschlossen zu haben und einzeln anzusprechen bzw. abzufragen. Außerdem enthält der ROM-Code einen 8 Bit langen Prüfcode (CRC = Cyclic Redundancy Check), mit dem die korrekte Übertragung der vorangegangenen 56 Bits des ROM-Codes geprüft werden kann.

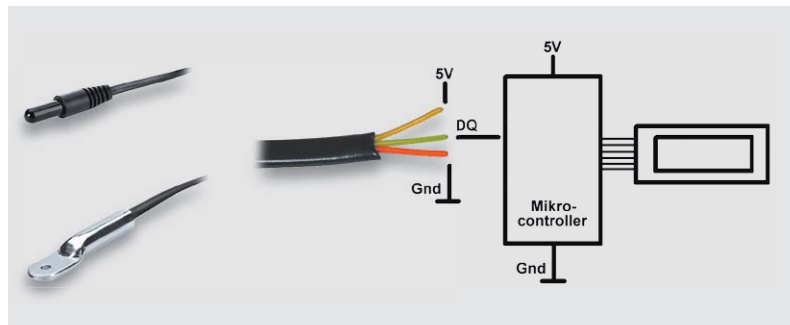


Bild 4: Gehäuseformen und Anschluss

DS18B20-Temperatursensor

Die bekannteste Anwendung des 1-Wire-Busses ist die Temperaturmessung mit DS18B20 (CN-10 93 37, CN-10 27 83) oder den älteren DS18S20. Deshalb wird die Arbeit mit 1-Wire im Folgenden anhand des DS18B20 erläutert. Mit einem DS18B20 lässt sich die Temperatur im Bereich von -55 °C bis $+125 \text{ °C}$ messen, wobei die Temperaturmessung zwischen -10 °C und $+85 \text{ °C}$ mit einer Genauigkeit von $0,5 \text{ °C}$ erfolgt. (Die Genauigkeit gibt an, wie genau die vom Sensor dargestellte Temperatur der wahren Temperatur entspricht.) Die Auflösung lässt sich zwischen 9 und 12 Bit einstellen. (Die Auflösung gibt an, wie fein abgestuft eine analoge Größe digital dargestellt wird.) Eine Temperaturumwandlung mit der maximalen 12-Bit-Auflösung erfolgt in 750 ms. Die Spannungsversorgung kann zwischen 3 Volt und 5,5 Volt sein, wobei wie oben beschrieben eine parasitäre Spannungsversorgung zum Einsatz kommen kann und dann außer der Gnd-Leitung nur eine einzige (Daten-) Leitung benötigt wird. Mit einem zweiadrigen Kabel kann man also verschiedene Sensoren an verschiedenen Orten an einem Bus einsetzen. Bild 4 zeigt links zwei verschiedene von ELV angebotene Gehäuseformen des DS18B20 – für Flüssigkeiten geeignet bzw. mit metallischer Befestigungsglasche. Rechts im Bild sieht man den Anschluss an einen Mikrocontroller. Bild 4 zeigt die nicht parasitäre Spannungsversorgung.

Der Blick auf Bild 5 zeigt die verschiedenen Blöcke im Innern des DS18B20. Der linke Block erkennt, ob der parasitäre oder der normale Spannungsversorgungsmodus verwendet wird. In diesem Block ist auch der Kondensator angedeutet, der bei parasitärer Spannungsversorgung für die Speicherung der Spannung zum Einsatz kommt. Der zweite Block zeigt den Speicherbereich für den 64-Bit-Code, der wie oben beschrieben den Family-Code beinhaltet und eine eindeutige Adressierung des Bausteines ermöglicht. Der Inhalt dieses Speicherbereiches lässt sich nicht durch den Benutzer verändern (ROM = Read Only Memory). In dem Speicherbereich mit dem Namen Scratchpad sind die Temperaturdaten und ein Konfigurationsregister gespeichert (vgl. auch Bild 11).

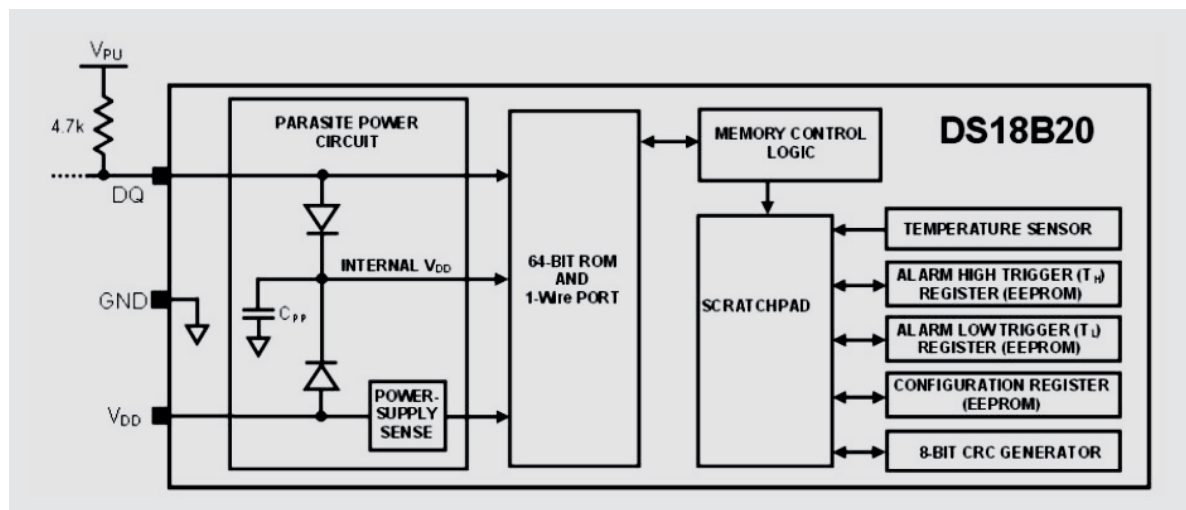


Bild 5: Aufbau DS18B20 Temperatursensor (Datenblattauszug)



1. ROM Commands (Adressierung)			2. Function Commands (Befehl)		
33	Read ROM	ROM Code eines Slaves auslesen	44	Convert T	Temperaturmessung anstoßen
F0	Search ROM	ROM Codes mehrerer Slaves auslesen	BE	Read Scratchpad	Scratchpad lesen
CC	Skip ROM	Alle Slaves ansprechen	4E	Write Scratchpad	Scratchpad beschreiben (T _H , T _L und Konfigurationsregister)
55	Match ROM	Nur einen bestimmten Slave ansprechen	B4	Read Power Supply	Art der Spannungsversorgung auslesen
EC	Alarm Search	Slaves mit gesetztem Alarmflag ansprechen	48	Copy Scratchpad	T _H , T _L und Konfigurationsregister in EEPROM schreiben
			B8	Recall	T _H , T _L , Konfigurationsregister aus EEPROM in Scratchpad

Tabelle 1: Kommandos

Im Scratchpad gibt es auch je ein Register für Temperaturmaximum und Temperaturminimum, mit deren Hilfe der DS18B20 als Thermostat eingesetzt werden kann, indem er bei Über- bzw. Unterschreiten eines definierten Temperaturwertes einen Alarm signalisiert, der vom Benutzerprogramm ausgewertet werden kann. Rechts im Blockbild sieht man EEPROM-Speicherbereiche, deren Inhalte auch ohne Spannung am Sensor gespeichert bleiben. Dadurch kann man die Konfiguration und den Maximum-/Minimumwert für den Thermostat dauerhaft speichern. (Tipp: Wenn man die Thermostatfunktion nicht benötigt, kann man die beiden letzteren Register (T_H und T_L) auch zur dauerhaften Speicherung von zwei beliebigen (Byte-großen) Werten benutzen!)

Protokoll

Das Ansprechen eines 1-Wire-Slaves erfolgt immer nach dem gleichen Schema: Zunächst wird ein ROM Command (Adressierung) vom Master auf den Bus gesendet, welches angibt, welche Slaves angesprochen werden, und dann wird ein Function Command (Befehl) vom Master gesendet, welcher angibt, was auf Slave-Seite getan werden soll. [Tabelle 1](#) zeigt die möglichen Kommandos des DS18B20. Ein Standardablauf ist beispielsweise, dass zunächst der hexadezimale Wert CC vom Master auf den Bus gesendet wird, was angibt, dass alle Slaves gleichzeitig angesprochen werden sollen, und danach vom Master eine hexadezimale 44 auf den Bus gesendet wird, wodurch alle Slaves dazu veranlasst werden, eine Temperaturmessung vorzunehmen. Eine weitere Möglichkeit wäre, dass der Master eine hexadezimale 33 auf den Bus gibt, was den (einen) angeschlossenen Slave dazu veranlassen würde, seinen ROM-Code auf den Bus zu geben. Diese und weitere Kombinationen von ROM Command und Function Command werden im Folgenden detailliert analysiert.

```
ROM Family: 28
EC000000485FBCC28
```

ROM-Code auslesen

Beim Aufbau einer Anlage ist es sinnvoll, zunächst zu prüfen, wie die ROM-Codes der teilnehmenden Temperatursensoren sind. Dazu schließt man *einen* DS18B20 am Bus an und der Master gibt eine hexadezimale 33 (Read ROM Kommando) auf den Bus. Eine genaue Analyse mit einem Logikanalysator ([Bild 6](#) und der Blick in das Datenblatt [\[2\]](#)) zeigt, dass am Anfang jeder 1-Wire-Kommunikation eine sogenannte Reset Condition vom Master auf den Bus gegeben wird, die von Slave-Seite mit einem „Presence“ beantwortet wird. Der Signalverlauf aus [Bild 6](#) wird nun anhand [Bild 7](#) im Einzelnen besprochen: Im Ruhezustand liegt der Bus durch den Pull-up-Widerstand auf logisch High. Um eine Kommunikation zu beginnen, zieht der Master den Spannungspegel auf der Datenleitung für mindestens 480 µs auf Low (Reset Condition/Reset Pulse) und gibt den Bus dann wieder frei (High durch Pull-up-Widerstand). Der Slave wartet 15 bis 60 µs (damit saubere Signale vorliegen) und zieht dann den Bus seinerseits für 60 µs bis 240 µs auf Low und signalisiert damit seine Bereitschaft zur Kommunikation (vgl. [Abbildung 13](#) im Datenblatt [\[2\]](#)).

Danach kann der Master nun das Read-ROM-Kommando (eine hexadezimale 33) auf den Bus geben.

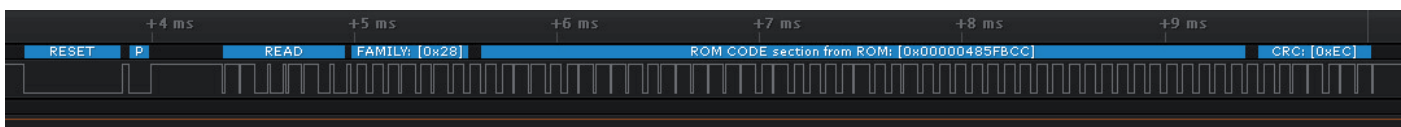


Bild 6: Signalverlauf ROM-Code lesen (Gesamtdarstellung)

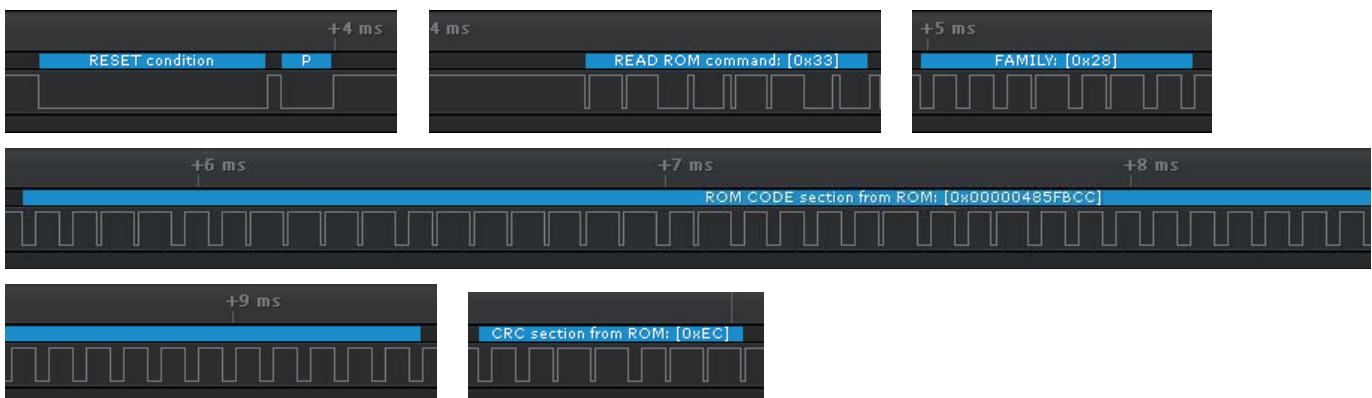


Bild 7: Signalverlauf ROM-Code lesen (in Teile zerlegt)



```

RESET condition
PRESENCE condition
READ ROM command: [0x33]
FAMILY CODE section from ROM: [0x28]
ROM CODE section from ROM: [0x00000...
CRC section from ROM: [0xEC]

```

Bild 8: Dekodierte Zeichen: ROM-Code lesen

Das macht der Master, indem er den Bus für eine 1 für maximal 15 μ s und für eine 0 für mindestens 60 μ s auf Low zieht. In Bild 7 kann man im zweiten Teilbild die seriell vom Master gesendete Bitfolge 1 1 0 0 1 1 0 0 erkennen. Wenn man nun (aus dem Datenblatt) weiß, dass stets das niedrigstwertige Bit zuerst übertragen wird, dann kann man diese Bitfolge manuell als 33 dekodieren (weil 0011 0011 binär = 33 hexadezimal ist). Der Slave wertet im Zeitraum zwischen 15 μ s und 60 μ s nach einer fallenden Flanke aus, ob ein High-Pegel (1) oder ein Low-Pegel (0) am Bus anliegt (Abbildung 14 oben im Datenblatt [2]).

Nachdem der Slave nun diese 33h empfangen hat, gibt der Slave den Family-Code auf den Bus (drittes Teilbild in Bild 7). Dafür zieht der Master für jedes Bit den Bus für mindestens 1 μ s auf Low und gibt ihn dann wieder frei, woraufhin der Slave für eine logische 1 den Bus sofort auf High gehen lässt und für eine logische 0 den Bus für ca. 14 μ s auf Low gezogen hält, was der Master jeweils erkennen kann (Abbildung 14 im Datenblatt [2]). Im Beispiel wird als Family-Code seriell 0 0 0 1 0 1 0 0 übertragen. Da wiederum das niedrigstwertige Bit zuerst übertragen wurde, wurde also 0010 1000, also 28h übertragen. Nach dem Family-Code 28h wird die einzigartige 48 Bit lange Seriennummer in gleichartiger Technik vom Slave an den Master signalisiert. Abschließend wird, wie in Bild 7 im letzten Teilbild zu sehen, die Prüfsumme (CRC) über die vorangegangenen Bits übertragen.

Noch einmal zur Übung: In diesem Fall sieht man am Signalverlauf, dass die Bits 0 0 1 1 0 1 1 1 übertragen werden. Da wieder „rückwärts“, also das niedrigstwertige Bit zuerst übertragen wird, handelt es sich um 1110 1100 = Ech. Damit ist eine komplette Kommunikation zwischen Master und Slave beendet. Der Spannungspegel auf dem Bus wird durch den

Pull-up-Widerstand auf High gezogen gehalten. Der Master kennt nun den ROM-Code des angeschlossenen Slaves. Die komplette Kommunikation aus Bild 6 sieht man in Bild 8 noch einmal in Form von (vom Logikanalysator) dekodierten Zeichen: RESET vom Master – PRESENCE vom Slave – Read ROM Kommando vom Master – Family-Code vom Slave – Seriennummer vom Slave – CRC vom Slave.

```

18B20 29.1250°C
0000000111010010

```

Temperatur messen und auslesen – ein Sensor

Um die Temperatur mit einem angeschlossenen DS18B20 messen zu lassen und auszulesen, wird zunächst vom Master mit CCh 44h jeder angeschlossene DS18B20 veranlasst, die Temperatur zu messen (vgl. Tabelle 1). Danach muss man mindestens 750 ms (bei höchster Auflösung) warten, bis die Temperatur intern im DS18B20 bereitsteht. Nach mindestens 750 ms (bei 12-Bit-Auflösung) kann der Master dann CCh BEh (vgl. Tabelle 1) auf den Bus geben, um alle (in diesem Fall den einzigen) DS18B20 zum Übertragen der Temperatur auf den Bus aufzufordern. Die Temperatur ist in einem Speicherbereich mit dem Namen Scratchpad im DS18B20 gespeichert. Bild 9 zeigt den Signalverlauf der zwei Schritte. Im oberen Teilbild ist zu sehen, dass die Kommunikation wie oben beschrieben wieder mit Reset vom Master und Presence vom Slave beginnt. Danach gibt der Master CCh 44h auf den Bus, wodurch alle angeschlossenen DS18B20 angewiesen werden, die Temperatur zu messen. Nach Ablauf von mindestens 750 ms (bei 12-Bit Auflösung) erfolgt eine zweite Kommunikation (Bild 9 unten), die wiederum mit Reset/Presence beginnt. Der Master sendet dann CCh BEh woraufhin der Slave die Bits des Scratchpads nacheinander auf den Bus gibt. Bild 10 zeigt die übertragenen dekodierten Daten auf dem 1-Wire-Bus. Das Scratchpad besteht aus 9 Bytes (Bild 11).

Der Wert der Temperatur befindet sich in Byte 0 und Byte 1. Bild 12 zeigt den Aufbau der Temperaturregister. Die Bits 15 bis 11 stellen das Vorzeichen der Temperatur dar. Die restlichen Bits stellen den Wert der Temperatur dar. Bild 13 zeigt beispielhafte Temperaturwerte. Um die Fehlerfreiheit der übertragenen Bits zu überprüfen, kann man die Prüfsumme (CRC) aus Byte 8 benutzen.

Read Power Mode

Wie oben beschrieben kann man einen DS18B20 (wie auch andere 1-Wire Slaves) „normal“ (mit einer positiven Spannung und Gnd) oder im parasitären Spannungsversorgungsmodus (Spannung wird aus Datenleitung gewonnen) betreiben. Der Master kann per Software abfragen, in welcher der beiden Betriebsarten sich ein angeschlossener Slave befindet.

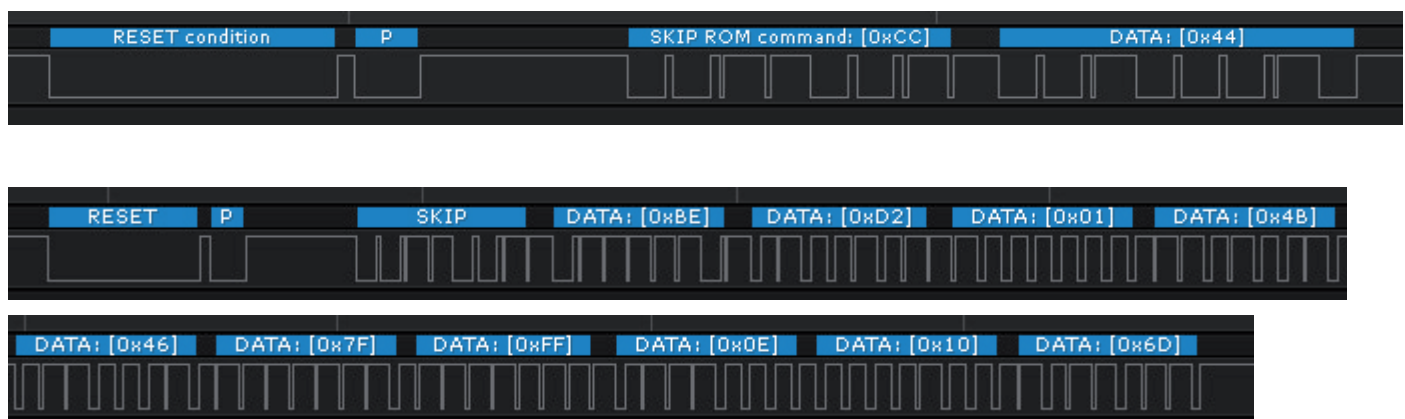


Bild 9: Temperaturmessung anstoßen (oben) und Temperatur auslesen (unten) (Signalverlauf)



Bild 10: Temperaturmessung anstoßen und auslesen (dekodierter Text)

```

RESET condition
PRESENCE condition
SKIP ROM command: [0xCC]
DATA: [0x44]
-----
RESET condition
PRESENCE condition
SKIP ROM command: [0xCC]
DATA: [0xBE]
DATA: [0xD2]
DATA: [0x01]
DATA: [0x4B]
DATA: [0x46]
DATA: [0x7F]
DATA: [0xFF]
DATA: [0x0E]
DATA: [0x10]
DATA: [0x6D]
    
```

SCRATCHPAD	
Byte 0	Temperature LSB
Byte 1	Temperature MSB
Byte 2	T _H Register or User Byte 1
Byte 3	T _L Register or User Byte 2
Byte 4	Configuration Register
Byte 5	Reserved (FFh)
Byte 6	Reserved
Byte 7	Reserved (10h)
Byte 8	CRC

Bild 11: DS18B20 Memory Map (Scratchpad) (Datenblattauszug)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

S = SIGN

Bild 12: Temperaturregister DS18B20 (Datenblattauszug)

Er sendet dafür (bei einem angeschlossenen Slave) CCh B4h (vgl. Tabelle 1) an den Slave, woraufhin der Slave seinen Spannungsversorgungsmodus zurückgibt. FFh steht für nichtparasitäre/konventionelle Spannungsversorgung und 00h steht für parasitäre Spannungsversorgung. In Bild 14 ist der Signalverlauf der Kommunikation zu sehen.

Die Kommunikation beginnt wie immer mit einem RESET vom Master und einer Bestätigung (PRESENCE Condition) vom Slave. Danach folgt CCh (Skip ROM) und Read Power Supply B4h vom Master und die Antwort vom Slave (FFh bzw. 00h). Bild 15 zeigt die dekodierte Kommunikation.

TEMPERATURE (°C)	DIGITAL OUTPUT
+125	0000 0111 1101 0000
+85	0000 0101 0101 0000
+25.0625	0000 0001 1001 0001
+10.125	0000 0000 1010 0010
+0.5	0000 0000 0000 1000
0	0000 0000 0000 0000
-0.5	1111 1111 1111 1000
-10.125	1111 1111 0101 1110
-25.0625	1111 1110 0110 1111
-55	1111 1100 1001 0000

Bild 13: Temperaturwerte (Beispiele) (Datenblattauszug)

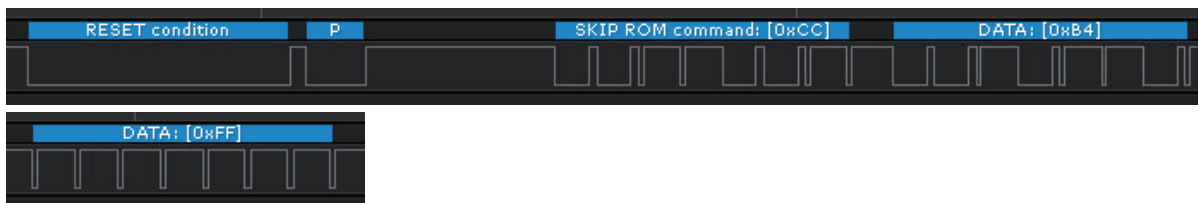


Bild 14: Signalverlauf Power Mode (geteilt)

```

RESET condition
PRESENCE condition
SKIP ROM command: [0xCC]
DATA: [0xB4]
DATA: [0xFF]
    
```

Bild 15: Power Mode (dekodiertes Signal)

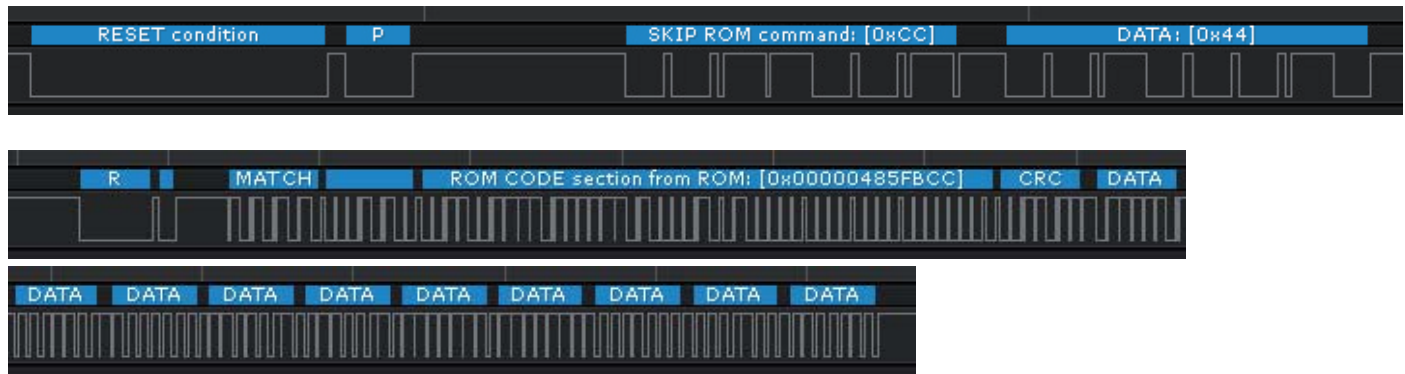


Bild 16: Signalverlauf Match ROM (unten geteilt)

```
18B20 25.5000°C
0000000110011000
```

Match ROM

In den bisherigen Beispielen war jeweils nur ein DS18B20 Sensor am Bus angeschlossen. In den meisten Fällen hat man mehrere 1-Wire-Slaves am Bus angeschlossen, die man direkt über die jeweilige eindeutige Seriennummer aus dem ROM-Code anspricht. Selbstverständlich können die Slaves nicht nur mehrere DS18B20-Temperatursensoren sein, sondern es kann auch eine Mischung aus DS18B20 und DS18S20 oder auch völlig anderen 1-Wire-Slaves sein. Über die Seriennummer und den Family-Code (beides im ROM-Code) ist jeder Slave eindeutig identifizierbar.

Bild 16 zeigt die Kommunikation für das Anstoßen der Temperaturmessung aller am Bus befindlichen Sensoren (CCh 44h) und das anschließende Auslesen der Temperatur eines bestimmten Sensors. Dieser Ablauf ist in der Praxis sehr nützlich, weil man im

Programm oft einen Ein-Sekunden-Interrupt hat, an dessen Ende man einfach pauschal ALLE Sensoren die Temperatur messen lässt und an dessen Anfang man gezielt die verschiedenen bekannten Sensoren (Außentemperatur, Innentemperatur 1, Innentemperatur 2 o. Ä.) auslesen und auswerten kann. In Bild 17 sieht man die dekodierte Kommunikation.

Die hier am DS18B20 gezeigten Beispiele von 1-Wire-Abläufen lassen sich auf andere 1-Wire-Bausteine übertragen. Der Ablauf ist immer gleich und die Details (Befehle, Register usw.) müssen dem jeweils verwendeten Slave-Datenblatt entnommen werden. Die meisten Programmiersprachen bieten 1-Wire-Libraries bzw. 1-Wire-Befehle für die einfache Einbindung von 1-Wire-Bausteinen an. So kann man elegant die Anzahl aller angeschlossenen 1-Wire-Slaves zählen lassen, die ROM-Codes aller angeschlossenen Slaves anzeigen lassen, Daten zu den Slaves senden und Daten von den Slaves empfangen. Da das 1-Wire-Timing exakt definiert ist, kann man sogar spezialisierte eigene Slaves programmieren, die sich an einen 1-Wire-Bus hängen lassen.

Fazit

Die 1-Wire-Schnittstelle ist eine gut verständliche asynchrone serielle Schnittstelle, die es ermöglicht, sehr viele – auch unterschiedliche – Slaves an einem Bus aus 3 bzw. sogar 2 Leitungen an unterschiedlichen Stellen im Labor oder im Haus über weltweit einzigartige Seriennummern anzusprechen. Im nächsten Teil dieser Artikelserie geht es um die Grundlagen der USB-Schnittstelle, die bekanntermaßen eine enorme Verbreitung hat. **ELV**

```
RESET condition
PRESENCE condition
SKIP ROM command: [0xCC]
DATA: [0x44]
RESET condition
PRESENCE condition
MATCH ROM command: [0x55]
FAMILY CODE section from ROM: [0x28]
ROM CODE section from ROM: [0x00000...
CRC section from ROM: [0xEC]
DATA: [0xBE]
DATA: [0x98]
DATA: [0x01]
DATA: [0x4B]
DATA: [0x46]
DATA: [0x7F]
DATA: [0xFF]
DATA: [0x08]
DATA: [0x10]
DATA: [0x22]
```

Bild 17: Dekodiertes Signal Match ROM



Weitere Infos:

- [1] DS18B20 vs. DS18S20/DS1820: <http://www.maximintegrated.com/en/app-notes/index.mvp/id/4377>
- [2] Datenblatt DS18B20: <http://www.elv.de>: Webcode #10096

Preisstellung April 2017 – aktuelle Preise im ELV Shop

Empfohlene Produkte	Best.-Nr.	Preis
H-Tronic TS 1 Temperatursensor DS18B20	CN-10 93 37	€ 12,95
H-Tronic TS 2 Temperatursensor DS18B20	CN-10 27 83	€ 14,95
Kupplung für RJ45-Stecker	CN-10 65 34	€ 1,-
ISDN-Kabel mit RJ45-Steckern	CN-10 65 35	€ 6,50
Delock Adapter Terminalblock mit Drucktaste > RJ45 Buchse	CN-11 52 88	€ 5,95
Modular-Einbaubuchse 2x RJ45	CN-11 21 47	€ 0,88



MITMACHEN & GEWINNEN

ELV Journal-Leser testen und gewinnen

Ihre Meinung interessiert uns! Bewerben Sie sich als Tester und schreiben Sie für die nächste Ausgabe einen Testbericht! Was gefällt Ihnen, was gefällt Ihnen nicht? Was kann man verbessern? Unter allen Bewerbern lösen wir die glücklichen Tester aus, die dann natürlich das jeweilige Testgerät behalten dürfen!

5x Batavia-Wowstick-Elektronik-Präzisions-Akkuschrauber mit Li-Ion-Akku



Der innovative Mini-Akkuschrauber lässt sich wie ein Stift halten und ermöglicht komfortables und genaues Arbeiten in allen Bereichen, in denen es auf Präzision ankommt, z. B. bei der Reparatur von Smartphones und anderen elektronischen Geräten mit kleinen Schrauben.

Best.-Nr. CN-12 58 15

Wert € 69,95

10x ELV X-BLAZE 800, LED-Taschenlampe mit Akku, aufladbar per USB




Die hochwertig verarbeitete, wasserdichte Taschenlampe ist dank Lithium-Ionen-Akku immer einsatzbereit. Die richtige Empfehlung für harten, professionellen Einsatz.


Best.-Nr. CN-12 35 38

Wert € 44,95

So werden Sie ELV Journal-Leser-Tester und können gewinnen!*

ELV Journal verlost unter allen Bewerbern je ein Exemplar von 5x Batavia-Wowstick-Elektronik-Präzisions-Akkuschrauber oder 10x ELV X-BLAZE 800, LED-Taschenlampe mit Akku. Bewerben Sie sich jetzt!

 **Per E-Mail** an lesertest@elvjournal.de – geben Sie als Betreff bitte den Produktnamen an.

 **Online** auf www.lesertest.elvjournal.de – wählen Sie einfach Ihr Wunschprodukt aus.

Einsendeschluss
18.06.2017

Bitte geben Sie für Rückfragen Ihre Kontaktdaten an: Telefon, E-Mail-Adresse und (falls vorhanden) Ihre ELV Kundennummer.

Mehrfache Teilnahmen derselben Person werden als nur eine Teilnahme gezählt und erhöhen nicht die Gewinnchance.

Sie erhalten zum Testprodukt eine ausführliche Bedienungsanleitung, gegebenenfalls weitere Informationen zum Produkt sowie einen Fragebogen, den Sie innerhalb von 4 Wochen nach Erhalt des Produkts und nach Abschluss des Tests an uns zurückschicken müssen.

Wir freuen uns auch über Fotos! Das Testprodukt dürfen Sie nach Abschluss des Tests natürlich behalten.

Die Gewinner aus dem letzten ELV Journal:

Solar-LED-Hausnummernleuchte Steinel XSolar LH-N

Alexandra Link aus 24811 Owschlag
Andreas Kühn aus 09217 Burgstädt
Arthur Follmer aus 12527 Berlin
Georg Rötzer aus 93458 Eschlkam
Philipp Pohl aus 82515 Wolfratshausen



Hama-Digitaltuner DIT1000MBT – UKW-/DAB+/-/Internetradio

Andreas Langer aus 98693 Ilmenau
Volker Horn aus 21465 Wentorf
Wolfgang Schwarz aus 97490 Poppenhausen
Erwin Schmidt aus 91238 Engelthal
Wolfgang Kleber aus 19294 Neu Kalitz
Gesine Schuppe aus 14169 Berlin
Uwe Kantor aus 70176 Stuttgart
Lothar Ehrmann aus 74915 Waibstadt



* ELV ist berechtigt, die Testergebnisse unter der Nennung Ihres Namens im ELV Journal und auf www.elvjournal.de sowie www.elv.de zu veröffentlichen. Teilnahmeberechtigt sind Personen über 18 Jahre. Nicht teilnahmeberechtigt sind Mitarbeiter der ELV Elektronik AG und der e0-3-AG-Gruppe, der beteiligten Unternehmen und deren Angehörige sowie Gewinnspiel-Vereine und automatisierte -Dienste. Unter allen fristgerecht eingegangenen Einsendungen entscheidet das Los. Mehrfache Teilnahmen derselben Person werden als nur eine Teilnahme gezählt und erhöhen nicht die Gewinnchance. Eine Barauszahlung oder ein Tausch gegen andere Produkte ist nicht möglich. Der Rechtsweg ist ausgeschlossen.



Unsere Leser testeten

eBlocker Pro – privater Netzwerkschutz

Anleitung/Bedienung



Inbetriebnahme/Konfiguration



Betrieb/Nutzung



Werbeblocker



CN-12 73 65

€ 199,-

Oliver Wiegand:*„Eine übersichtliche Menüführung und die leichte Installation machen das Gerät sehr einsteigerfreundlich.“*

Unsere Leser bewerteten

2,0

Durchschnitt

**Unsere Leser bewerteten:
Durchschnittsnote 2,0**

In die Gesamtnote sind weitere Kriterien unseres Fragebogens einbezogen.



Anschließen und auf Knopfdruck anonym im Internet surfen – die kompakte Box blockiert unerwünschte Online-Werbung, erschwert Tracking und verschleiert via Anon-Funktion die eigene IP-Adresse bei allen Geräten eines Heimnetzwerks. So hinterlässt man im Internet keine digitalen Fingerabdrücke und kann genau festlegen, welche eigenen Daten man freigibt. Über die Anon-Funktion kann man anonym im Tor-Netzwerk surfen, und das ohne störende Werbung auf den besuchten Seiten.

Wir baten fünf Leser um einen Test aus Anwendersicht.

So muss ein modernes Netzwerkgerät funktionieren – wir springen mitten in das Testfazit der Tester hinein, das vor allem die einfache Installation und Einrichtung sowie den problemlosen Umgang hervorhebt. So muss das sein! Bedienungsanleitung und Inbetriebnahme erhielten ebenso gute Noten wie die Menüführung und die Einstellung des eBlockers. Zwei der fünf Tester konnten das Gerät per Plug & Play in Betrieb nehmen, drei Tester hingegen haben die Inbetriebnahme manuell ausgeführt.

Bei einem solchen recht komplexen Gerät gibt es einiges, das detailliert erklärt werden muss, um die Funktionen auch ohne IT-Studium in Betrieb nehmen zu können. Deshalb haben wir im Detail gefragt, ob bestimmte Funktionen gut genug beschrieben sind. Die Frage „Ist die Verbindungsanalysefunktion für Apps hinreichend erklärt?“ beantworteten vier Tester mit „Ja“, und: „Ist das Hinterlegen des SSL-Zertifikats gut erklärt?“ wurde von drei Testern mit „Ja“

beantwortet. Besonders gut gefiel den Testern der Werbeblocker und das Abfangen von Tracking-Cookies bereits am Router, dies auch für mobile Geräte. Immer wieder wurden die einfache Installation, die problemlose Endgeräteerkennung und die Funktionalität hervorgehoben. Der Werbeblocker funktioniert, jedoch fanden einige Werbeseiten die Lücke, trotzdem bis zum Endgerät durchzukommen. Hier fehlte dem betroffenen Tester die Möglichkeit, diese manuell im eBlocker sperren zu können.

Ein Wermutstropfen ist die jährlich gegen eine Gebühr zu erneuernde Nutzungslizenz, allerdings wird dafür auch der eBlocker automatisch stets auf dem neuesten Softwarestand gehalten. Gerade im sich rasant entwickelnden Internet-Umfeld eine unschätzbare Option, die zudem kein Zutun des Nutzers erfordert.

Auch die bei manchen Routermodellen erforderliche manuelle Konfiguration wurde bemängelt – IT-Laien können hier schnell an Grenzen stoßen.

Einem Tester fiel die verringerte Surfgeschwindigkeit bei der Tor-Nutzung auf – dies dürfte aber eher dem Tor-Browsersystem mit dessen verschlungenen Zugangswegen anzulasten sein als dem eBlocker.

Was fehlt den Testern? Ein Reset-Button zum Zurücksetzen auf die Werkseinstellungen, ein Ein-/Ausschalter samt Betriebskontrolle, eine für manche Experten-Konfigurationen von außen sichtbare MAC-Adresse. Auch die erwähnte Eintragungsmöglichkeit, durchgekommene Werbeseiten zukünftig auszusperrern, wurde hier nochmals genannt. Ein Tester schlug ein interessantes Feature vor: die Nutzung des eBlockers auf entfernten Endgeräten, die per VPN auf den heimischen Router zugreifen.

Fazit: Der eBlocker tut, was er tun soll, und das ohne wirklich gravierende Probleme und Mängel. Sowohl Inbetriebnahme, Konfiguration, Bedienung als auch der Betrieb sind zum größten Teil problemlos, es bleiben nur wenige Wünsche offen.

ELV



Unsere Leser testeten

ELV Funk-Kamerasystem KS200 HD mit App

ELV

Anleitung



Inbetriebnahme



Aufnahme-/Bildqualität



App

**Günter Rehra:***„Leicht zu installieren; guter Monitor mit Akkubetrieb; hochwertige Kamera.“***Unsere Leser bewerteten:
Durchschnittsnote 2,0**

In die Gesamtnote sind weitere Kriterien unseres Fragebogens einbezogen.



CN-12 37 22

€ 199,95

Unsere Leser bewerteten

2,0

Durchschnitt

Das Funk-Kamerasystem ermöglicht die einfache Installation eines Überwachungssystems mit bis zu vier Funk-Kameras. Der Monitor dient gleichzeitig als Empfänger und zudem als Recorder. Er verfügt über einen besonders einfach bedienbaren Touchscreen und besitzt einen Netzwerkanschluss. Zusätzlich erlaubt das Videoüberwachungssystem den Fernzugriff über das Internet und eine Mobilgeräte-App für Android und iOS. Wir baten drei Leser, das System zu testen.

Das generelle Fazit der Tester war einhellig: leicht zu installieren, intuitive Bedienung, gute Bildqualität. Damit erweist sich dieses Kamerasystem als das, was es sein soll: eine unkomplizierte Videoüberwachungsanlage.

Das erste Kennenlernen inklusive Inbetriebnahme erhielt gute bis sehr gute Bewertungen, ebenso die Verarbeitung der Geräte, das Preis-Leistungs-Verhältnis, die Monitor-Bildqualität und die Funkreichweite der Kamera. Auch der integrierte Infrarot-Scheinwerfer der Kamera kam insbesondere mit seiner Reichweite bei den Testern gut an.

Besonders die Qualität der Kamera wurde mit guten Bewertungen bedacht. Hervorgehoben wurden hier die hohe Bildqualität, die gute Gegenlichtkompensation und die gute Nachtsicht mit Infrarot-Unterstützung. Den Testern gefiel auch der durch Akkubetrieb mobile Monitor, die Akku-Laufzeit von drei Stunden reicht dann auch für einen längeren Betrieb auf Terrasse oder Balkon.

Weniger gut gefiel die für den Fernzugriff auf das System gehörende App, hier herrscht noch Verbes-

serungsbedarf, insbesondere bei den Konfigurations- und Einstellmöglichkeiten über die App sowie bei Alarmierungs-Ereignissen.

Die Tester haben sich sehr ausführlich mit dem System beschäftigt, das sieht man an den formulierten Wünschen. So werden erweiterte Kameraoptionen vermisst, wie eine Bilddrehung oder eine Ausblendmöglichkeit von nicht zu überwachenden Bereichen. Auch wünscht man sich erweiterte Einstellmöglichkeiten wie z. B. für die Bewegungserkennung. Schließlich wünscht man sich die gleiche Funktionalität in der App, wie sie der Monitor bietet.

Ein Tester machte auch auf ein Installationsproblem aufmerksam – er wünscht sich eine andere Art des Stromversorgungsanschlusses, um nicht den recht dicken Entstörfilter im Stromkabel durch eine Wand führen zu müssen, wie es in einigen Einbausituationen vorkommen kann. Auch wäre heute eine USB-Buchse statt eines Rundsteckers am Monitor eine Option, damit man u. a auch eine Powerbank zur Verlängerung des mobilen Betriebs des Monitors anschließen kann. Immerhin dauert das Aufladen des Akkus ebenfalls drei Stunden.

Fazit: Ein gutes System, das unkompliziert in Betrieb zu nehmen und zu nutzen ist. Besonders die einfache Bedienung, die gute Bildqualität am Tag und in der Nacht sowie die Möglichkeit, den Monitor mobil benutzen zu können, gefielen im Test. Dass tiefergehende Einstellmöglichkeiten fehlen, gehört zum Konzept der besonders einfachen Bedienung des Systems, vielleicht kann man diese in folgenden Generationen als Experteneinstellung hinterlegen. Die Bedienungsanleitung könnte mehr zum Thema „Alarmauslösung und -Behandlung“ aussagen, hier bleibt eine Unsicherheit im Betrieb zum Thema Alarmierung.

ELV**Sie wollen es genau wissen?**Die Testberichte unserer Leser finden Sie auch unter:
www.lesertesten.elvjournald.de



Die große Chance mit

Warum der Anschluss von Mini-Anlagen immer einfacher wird



Die Idee ist so großartig, dass es einfach nur eine Frage der Zeit war, bis sie sich durchsetzte: Mini-PV-Anlagen mit einer Leistung von wenigen hundert Wp erfreuen sich immer größerer Beliebtheit. Doch es gibt auch kritische Stimmen, wie die des VDE, der vor zahlreichen technischen und rechtlichen Risiken warnt. Was steckt dahinter? Was ist zu beachten? Gibt es Neuigkeiten zu den Themen Anschluss und Betrieb sowie Eigenverbrauch und Einspeisung? Um es vorwegzunehmen: ja, sehr erfreuliche – zumindest für potenzielle Mini-Anlagen-Betreiber.



kleinen Solarstromanlagen

Dezentrale Energieversorgung als zentrales Zukunftsthema

„So dezentral wie möglich, so zentral wie nötig.“ So lautet die Kernforderung, die der Bundesverband mittelständische Wirtschaft (BVMW) in Bezug auf die Energiewende und eine zukunftssichere Energieversorgung erhebt. Für manche mag diese Forderung nach Dezentralität in der Energieversorgung befremdlich erscheinen – Fakt ist jedoch: Auch der BVMW steht seit Jahren auf der Seite der Erneuerbaren Energien und einer dezentralen Stromerzeugung und nennt hier Vorteile wie Kosteneinsparungen, höhere Effizienz, Entlastung der Netze, Teilhabe der Kommunen und Bürger etc. [1].

In Deutschland sind die Zahlen bei Photovoltaik-Anlagen eindeutig: Mehr als 98 % der Solarstromanlagen speisen als dezentrale Solarstromanlagen ins Niederspannungsnetz ein [2]. Es geht aber längst nicht mehr hauptsächlich um die Einspeisung durch Kopplung mit dem öffentlichen Stromnetz: Gerade die Unterstützung des Eigenbedarfs an Strom ist durch das kontinuierliche Sinken der Stromgestehungskosten und die (fast) jährliche Preissteigerung des vom Energieversorger gekauften Stroms äußerst attraktiv. Und mehr ist da nicht in jedem Fall mehr. Es kommt darauf an, was einem wichtig ist.

Groß im Nutzen, klein im Aufwand

Im Gegensatz zu großen, komplexen PV-Anlagen über viele Quadratmeter des Hausdaches sowie intelligenter Speicher- und Hausautomationstechnologie kommen Mini-Anlagen geradezu „vorsintflutlich“ daher: ein Modul, ein Wechselrichter, ein Stecker. Im Prinzip braucht es nicht mehr als diese Komponenten. Nachdem das Solarmodul an der Balkonbrüstung, auf der Terrasse oder auf dem Dach befestigt worden ist, muss in der Regel nur noch der Wechselrichter angeschraubt werden. Dann folgt noch der Anschluss an die AC-Seite des Hausstromnetzes. Und genau da liegt der Hase im Pfeffer. Aber dazu kommen wir gleich.

Convenient Technologie: nicht für Tüftler, sondern ausgetüftelt

Während es vor einigen Jahren noch viel technisches Handwerkzeug und Wissen erforderte, eine PV-Anlage zu installieren und zu betreiben, ist es heute relativ simpel geworden. Denn auch hier gehen die Bemühungen der Hersteller dahin, Plug-&-Play-Lösungen für den Massenmarkt zu bieten. Welcher



Eigenverbrauch mit Mini-PV-Anlagen: sauberer Strom für die Grundlast (Foto: SUNSET Energietechnik GmbH)

klassische Endverbraucher will schon Komponenten zusammenschrauben, Einstellungen konfigurieren, Kabel anbringen und unter Umständen auch noch den Energiezähler („Stromzähler“) austauschen lassen? Wie bei anderen Produkten geht der Trend wohl auch bei Elektronik und Energie in Richtung „Convenient Technologie“ – oder wie auch immer man es nennen will. Ready to run: auspacken, platzieren, Stecker in die Steckdose und loslegen. Warum auch nicht?

Anschluss über Kabel oder Dose?

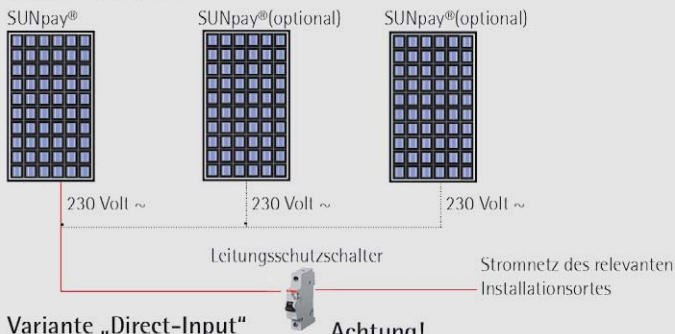
Bei der Verwendung elektrischer „Verbraucher“ oder Erzeugungsanlagen stellt sich immer die entscheidende Frage: Wie wird das angeschlossen? Und hier schieden sich bisher die Geister. Ging man bis dato davon aus, dass hiesige Normen und Gesetze unbedingt die feste Installation in die elektrischen Kreisläufe des Haushalts vorschreiben, scheint sich daran aktuell einiges zu ändern. So beschreiben die meisten Anbieter von Mini-PV-Anlagen die (theoretisch möglichen) zwei simplen Anschlussvarianten.

Der deutsche Hersteller SUNSET etwa bietet für seine SUNpay-Anlage die „Regular“- und die „Direct Input“-Lösung an: Entweder wird beim Anschluss an die AC-Seite das AC-Kabel an der Anschlussstelle über eine Verteilerdose (FI-geschützt) in den Sicherungskasten geführt – wodurch eine feste Installation des SUNpay 250 in die elektrischen Kreisläufe des Installationsortes entsteht. Oder aber es wird beschrieben, dass der mitgelieferte Stecker am AC-Kabel angebracht werden kann. Selbstverständlich wird an dieser Stelle immer darauf hingewiesen, dass

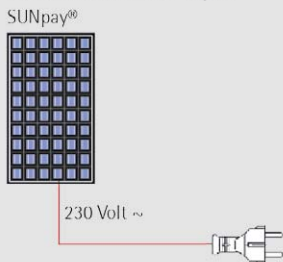


Blockschaltbild

Variante „Regular“



Variante „Direct-Input“



Achtung!

Der vorgeschriebene NA-Schutz ist in dem SUNpay® standardmäßig enthalten.

Vorsicht!

Variante „Direct Input“ muss mit dem jeweiligen Netzbetreiber auf Netzkonformität geprüft werden. Sollte diese Variante seitens des Netzbetreibers oder der Gesetzgebung nicht rechtens sein, kann diese Variante nicht zum Einsatz kommen.

Anschluss via Leistungsschutzschalter oder Schuko stecker: Variante „Regular“ bzw. „Direct Input“ (Quelle: SUNSET)

diese Anschlussart zwar in Ländern wie Österreich, der Schweiz oder den Niederlanden erlaubt ist, jedoch nicht in Deutschland.

Betrieb von Plug-in-Anlagen ist rechtens

Neben dem rein technischen Aspekt stehen auch die bisherigen rein rechtlichen Hindernisse auf dem Prüfstand: Und es scheint sich bei genauerer Betrachtung zu bestätigen, dass dieser Sachverhalt juristisch längst nicht mehr so spannungsgeladen ist. Denn der oft zitierte § 49 EnWG, der die Anforderungen an Energieanlagen bzw. Erzeugungsanlagen regelt, kann nicht mehr gegen den Anschluss von Mini-Anlagen ins Feld geführt werden. Dr. Jörn Bringewat, Jurist der Greenpeace Energy AG, erklärt: „Netzbetreiber haben nur die Möglichkeit, Vorschriften der Nie-

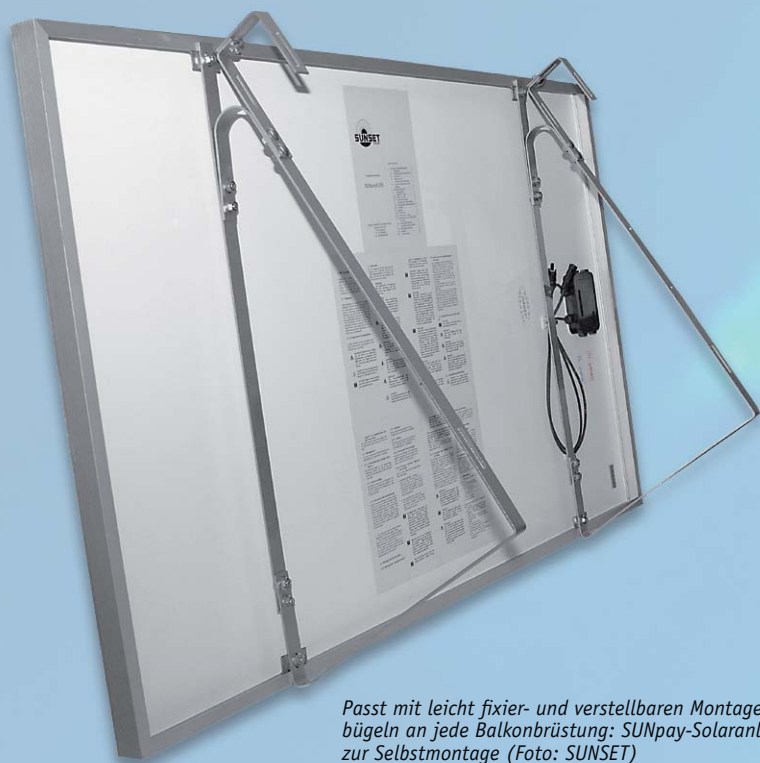
derspannungsanschlussverordnung (NAV) geltend zu machen. Diese Verordnung ist allerdings alleine dafür relevant, den reibungslosen Netzbetrieb zu gewährleisten. Netzbetreiber können also nur eingreifen, wenn schädliche Netzzrückwirkungen zu erwarten sind.“ [3]

Sicherer Wechselrichter mit NA-Schutz

Hinzu kommt, dass die vom VDE geforderte Fähigkeit der PV-Anlagen, die Stabilität von Spannung und Frequenz zu gewährleisten, erfüllt wird. So ist beispielsweise der Wechselrichter SUNpay 260 der SUNpay-Anlage mit einem Netz- und Anlagenschutz (Entkopplungsschutz) ausgestattet. Nach der VDE-Anwendungsregel VDE-AR-N 4105 muss die Anlage die Netzsicherheit, also die Stabilität von Spannung und Frequenz, sicherstellen. Bei etwaigen Störungen wie Spannungsabweichungen, Frequenzabweichungen, Fehlerströmen oder Reparaturen am Hausstromnetz wird die Anlage selbsttätig vom Netz getrennt.

Die Chancen für die Erlaubnis von Plug-in-Anlagen steigen

Bei dieser Nachricht dürften nicht nur Interessenten von Mini-Anlagen erfreut sein: Im Streit um die Nutzung kleiner Plug-&-Play-Solarmodule für Balkone und Terrassen gab erst Deutschlands größter Verteilernetzbetreiber Westnetz (ehemals RWE) den Anschluss der Module über die Steckdose frei, machte dann aber auf Druck des Mutterkonzerns einen Rückzieher. Der Hintergrund: Die Innogy-Tochter Westnetz hatte einer deutschen Kundin den Betrieb einer Mini-PV-Anlage mit Stecker verboten, weil sie gegen gültige Sicherheitsvorschriften des VDE verstoße. Bei dem anschließenden Verfahren vor der Bundesnetzagentur schlossen Gutachten unabhängiger Institute solche Risiken aus. Westnetz räumte abschließend ein, dass es nach Prüfung der vorgelegten Unterlagen an



Passt mit leicht fixier- und verstellbaren Montagebügeln an jede Balkonbrüstung: SUNpay-Solaranlage zur Selbstmontage (Foto: SUNSET)



Erfüllt mit integrierter Freischalteinrichtung (NA-Schutz) nach EN 50438 und VDE-AR-N 4105 alle Sicherheitsvorschriften: Wechselrichter SUNpay 260 (Foto: SUNSET)



seiner bisherigen Einschätzung nicht weiter festhalten und den Betrieb der Plug-in-Anlage erlaube, weil schädliche oder störende Rückwirkungen auf das Elektrizitätsversorgungsnetz nicht erkennbar seien. Und außerdem: Unterhalb der 300-Watt-Grenze verzichtete der westdeutsche Netzbetreiber zunächst zudem auf den Einbau von Stromzählern mit Rücklaufsperrung. Im Versorgungsgebiet von Westnetz sollte es für Nutzer genügen, Namen und Adresse, Leistung und Fabrikat der Module zu melden. Leider aber folgte nur kurze Zeit später das Dementi auf Druck des Mutterkonzerns Innogy.

Aber trotz des Hin und Her scheint eins sicher: Es ist eine Frage der Zeit, wann auch in Deutschland die strengen Vorschriften für Mini-Anlagen mit Stecker gelockert werden [4]: Denn die technischen Normen und gesetzlichen Regelungen für den Einsatz von Stecker-Solaranlagen sind veraltet. Noch in diesem Jahr ist aber damit zu rechnen, dass es eine anwenderfreundliche neue Norm geben wird, die dann auch die letzten unsinnigen Hürden für sichere Mini-Photovoltaik-Anlagen aus dem Weg räumen wird [5].

Betrieb von Mini-Anlagen ohne Finanzamt

Die erste Ankündigung von Westnetz, auf den zeitlich und finanziell aufwendigen Einbau von Stromzählern mit Rücklaufsperrung zu verzichten, ist auch steuerrechtlich von Bedeutung: Bisher galt, dass die etwaige, doch ins öffentliche Netz eingespeiste überschüssige Strommenge nach § 9 Abs. Nr. 1 oder 3 StromStG steuerbefreit und wegen eines fehlenden Zweirichtungszählers nicht nachweisbar war. Nun kommt hinzu, dass durch den Verbleib auch der alten Stromzähler der Nutzer gar keine Möglichkeit mehr hat, die unter Umständen zu versteuernde Strommenge zu messen. De facto bleibt also das Finanzamt außen vor, Nutzer müssen kein Gewerbe anmelden und den allenfalls in kleinen Mengen eingespeisten Strom auch nicht versteuern [6].

Signal für Interessenten von Balkon-Kraftwerken

Die Signalwirkung dieser Entscheidung (und des Rückziehers) von Westnetz ist immens: Jeder, der bisher mit der Vorstellung spielte, seinen Balkon oder seine Terrasse mit einem Solarmodul auszustatten und mit dem selbst produzierten, sauberen Strom den Stromzähler langsamer drehen zu lassen, kann dies jetzt in die Tat umsetzen. Dabei ist zu beachten, dass es bei einer 200- bis 300-Wp-Anlage in der Regel keine Einspeisung geben kann, da die Grundlast (je nach Haushaltsgröße und angeschlossenen Geräten) meistens um die 200 bis 400 W liegt. Diesem Modell folgend, könnten allein in Deutschland mehrere Gigawatt an Stecker-Solar-Geräten installiert werden. Dies entspricht der Leistung großer Braunkohlekraftwerke mit hohem CO₂-Ausstoß. Das wäre nicht nur eine saubere Leistung für die Umwelt, sondern auch eine Vielzahl kleiner Schritte für eine demokratische Energieversorgung, für mehr Unabhängigkeit von den (immer noch) mächtigen Stromkonzernen und für die Energiewende. **ELV**



Quellen:

- [1] BVMW: Energiewende. Forderungen und Positionen des Mittelstands. April 2014, S. 5
- [2] Aktuelle Fakten zur Photovoltaik in Deutschland, zusammengestellt von Dr. Harry Wirth, Fraunhofer ISE, 24.01.2017, S. 32
- [3] Jörn Bringewat: Neues zur rechtlichen Zulässigkeit von Mini-PV-Anlagen/PlugIn-PV-Anlagen (www.juop.org)
- [4] Presseerklärung Greenpeace Energy, 17.3.2017
- [5] www.pv-magazine.de/nachrichten/details/beitrag/innogy-fordert-von-westnetz-weiterhin-ablehnung-von-photovoltaik-balkonmodulen_100026192/
- [6] Jörn Bringewat: Rechtliche Aspekte der Nutzung von PlugIn-(Mini)-PV-Anlagen (www.juop.org)



Weitere Infos:

<https://www.photovoltaikeu/Gentner.dll/Archiv/Meldungsarchiv/Westnetzerlaubt-Anschluss-von-Balkonmodulen/QULEPTc1NzkwMiZNSUQ9MTEwOTQ5.html?AID=757902&MID=110949>

<http://www.pv-magazine.de/2017/03/02/dgs-nimmt-stellung-zu-warnhinweisen-fr-stecker-solar-gerte/>



HomeMatic Know-how

Teil 21: Homematic Sicherheit

In unserer Reihe „Homematic Know-how“ zeigen wir anhand von kleinen Detaillösungen, wie man bestimmte Aufgaben im Homematic System konkret lösen kann. Dies soll insbesondere Homematic Einsteigern helfen, die Einsatz- und Programmiermöglichkeiten besser zu nutzen. In dieser Ausgabe zeigen wir, welche Möglichkeiten es gibt, ein Homematic System möglichst sicher zu betreiben.

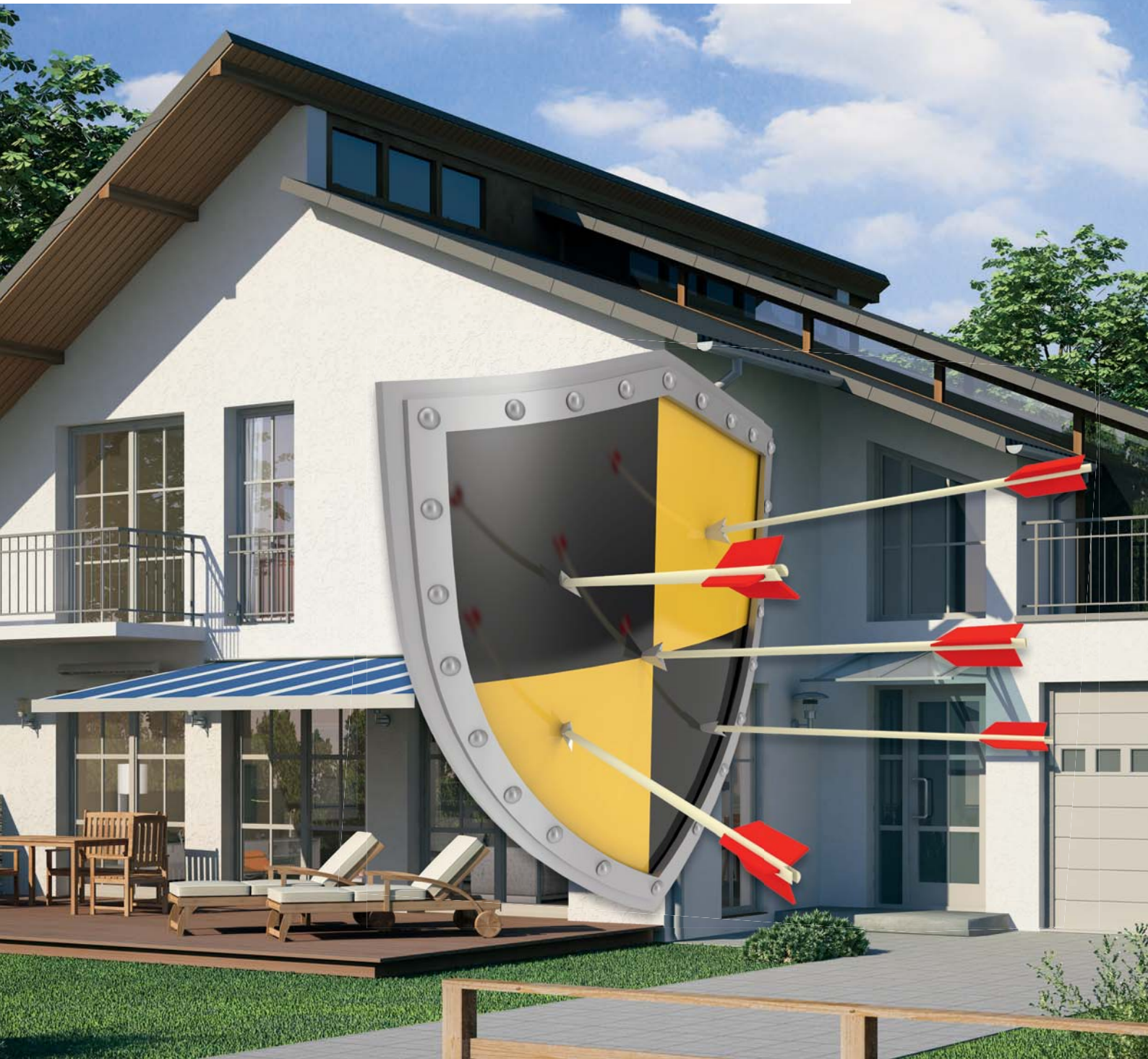




Bild 1: WebUI-Startseite mit dem rechts angezeigten Kennwort-Hinweis

Fragen zur Sicherheit eines Smart Home Systems kommen gerade in der heutigen „Immer-und-alles-online-Zeit“ sehr häufig auf. Nicht zuletzt auch, weil die Medien dieses immer wieder thematisieren, wobei allerdings die entsprechenden Lösungsansätze zur Optimierung der Sicherheit auf der Strecke bleiben.

Im Folgenden werden die wichtigsten Punkte in Bezug auf das Homematic System und der Verwendung der Homematic Zentrale CCU2 im lokalen Netzwerk sowie beim Fernzugriff und auch die gesicherte Funkkommunikation näher erläutert.

Lokaler Netzwerkzugriff auf die Homematic Zentrale

Bereits bei der Ersteinrichtung der CCU2 ist eine Verbindung mit dem heimischen Netzwerk erforderlich, um die aktuelle Firmware-Version einzuspielen sowie das System bequem per PC-Browser einrichten und programmieren zu können. Die folgenden Punkte sollten direkt bei der Einrichtung des Systems beachtet und entsprechend konfiguriert werden, um Unbefugten den Zugriff auf das System zu verwehren.

Gerade in Bezug auf das lokale Netzwerk ist allerdings auch anzumerken, dass in erster Linie der Betreiber des Netzwerks die Fäden selbst in der Hand hält. Man sollte sich daher zunächst einmal Gedanken machen, wer ggf. auch aus dem privaten Umfeld Zugang zum lokalen Netzwerk hat. Nahezu jeder besitzt heutzutage ein Smartphone und möchte gerne immer und überall surfen. Daher wird das WLAN-Passwort zum privaten Netzwerk schnell mal weitergereicht, wodurch dann bereits ein Zugriff auf alle Netzwerkgeräte möglich ist. Es empfiehlt sich, sofern der WLAN-Router es zulässt, für solche Zwecke ein WLAN-Gast-Netzwerk einzurichten, welches lediglich den Zugriff auf das Internet bietet, aber den Zugriff auf lokale Netzwerkgeräte verwehrt.

1. Benutzerpasswort einrichten

Nach dem Einspielen der aktuellen Firmware-Version der CCU2 wird man auf der Startseite der Homematic WebUI freundlich mit einem roten Hinweis „Kein Kennwort gesetzt“ begrüßt (siehe Bild 1).

Zur Einrichtung des persönlichen Benutzerkennworts fährt man mit dem Mauszeiger über den Button „Einstellungen“ und klickt anschlie-



Bild 2: Eingabemaske für das Benutzerpasswort

Bild 3: Einstellung für das automatische Anmelden an der WebUI

Bild 4: Nach abgeschlossener Programmierung sollte die WebUI immer per Klick auf „Abmelden“ verlassen werden.

Bild 5: Eingabefelder für das persönliche SSH-Passwort

Bild 6: Zeigt beispielhaft eine Firewall-Konfiguration. Die erste Zeile (192.168.6.10/20;) stellt einen Adressbereich da. Darunter drei Einzeladressen. Wichtig: In der letzten Zeile darf am Ende kein Semikolon stehen.

ßend auf „Benutzerverwaltung“. Im folgenden Fenster erscheinen alle angelegten Benutzer, für welche man jeweils durch einen Klick auf „Bearbeiten“ ein persönliches Benutzerkennwort festlegen kann (siehe Bild 2).

Möchte man zudem auch den auf der Anmeldeseite angezeigten Button mit dem Benutzernamen entfernen, kann der Haken für „Benutzername-Button in der Anmeldung“ entfernt werden.

Das gesetzte Passwort wird nach einem Klick auf „Einstellungen übernehmen“ gespeichert und bei jedem Einloggen auf der WebUI abgefragt.

Sofern weitere Nutzer das System über die WebUI lediglich bedienen sollen, lassen sich über die Benutzerverwaltung auch reine Benutzer- oder auch Gastkonten einrichten.

2. Automatisches Anmelden deaktivieren

Das automatische Anmelden auf der WebUI ist zwar sehr bequem, da einem die vielleicht lästige Eingabe der Zugangsdaten erspart bleibt. Man sollte dabei jedoch bedenken, dass jede Person, die Zugriff auf das lokale Netzwerk hat, einfach durch die Eingabe der Zentralen-IP-Adresse direkt und ohne Eingabe von Zugangsdaten auf die WebUI gelangt und sich hier frei bewegen kann.

Um das automatische Anmelden zu deaktivieren, klickt man in der Übersicht der Benutzerverwaltung unten auf den Button „Automatisches Anmelden“, wählt links in der Auswahl „nicht gewählt“ und speichert mit einem Klick auf „OK“ ab (siehe Bild 3).

3. Von der WebUI abmelden

Sofern man seine zu erledigenden Arbeiten auf der WebUI abgeschlossen hat, sollte immer eine saubere Abmeldung erfolgen. Hierdurch wird nicht nur die aktuelle Konfigurationssitzung abgespeichert, sondern auch die geöffnete Sitzungs-ID geschlossen (siehe Bild 4).

4. SSH-/SFTP-Zugriff

Ein „normaler“ Heimanwender wird den Konsolen- oder Dateisystem-Zugriff der Zentrale vermutlich weniger nutzen. Für fortgeschrittene Anwender oder gar Entwickler von Homematic Zusatzsoftware ist diese Zugriffsmöglichkeit unerlässlich.

Der SSH-Zugriff (Secure Shell) erfolgt über den Netzwerk-Port 22 und ermöglicht es dem Nutzer, über ein Konsolenprogramm wie z. B. Putty Shell-Befehle auf bzw. über die CCU auszuführen.

Des Weiteren kann mit SFTP-Programmen wie z. B. WinSCP auf die Dateisystemebene zugegriffen werden, wodurch das Kopieren/Erstellen/Ändern oder auch das Löschen von Dateien/Ordern möglich ist.

Ab der Firmware Version 2.7.8 ist der SSH-Zugriff nicht mehr werkseitig aktiv, dieser muss vom Nutzer durch das Setzen eines Passworts zunächst aktiviert werden, das ursprüngliche Standard-Passwort ist nicht mehr gültig.

Um SSH zu aktivieren, ist unter „Einstellungen → Systemsteuerung → Sicherheit“ ein Haken für „SSH aktiv“ zu setzen, ebenso ist die Eingabe eines Passworts erforderlich (siehe Bild 5).



5. Firewall konfigurieren

In der Zentralen-Firmware-Version 2.27.x und neuer ist die CCU-Firewall optimiert und automatisch aktiviert. Die Firewall ermöglicht den Netzwerkzugriff auf die XML-RPC-API-Schnittstelle sowie die Einschränkung der Remote-Script-API-Schnittstelle. Über die XML-RPC-API-Schnittstelle kann auf einzelne Geräte zugegriffen werden, die Remote-Script-API-Schnittstelle ermöglicht den Zugriff auf Räume, Gewerke und Programme. Die Firewall schränkt allerdings nicht den Zugriff auf den Webserver (die WebUI) der Zentrale ein.

In der WebUI werden hierzu unter „Einstellungen → Systemsteuerung → Firewall konfigurieren“ nur die Netzwerkgeräte eingetragen, welche tatsächlich mit der Zentrale kommunizieren sollen bzw. müssen.

Um den Netzwerkgeräten den Zugriff auf diese Schnittstellen zu gewähren, ist die Eingabe der Geräte-IP-Adressen bzw. Adressbereiche erforderlich. Damit der IP-Adressen-Filter vollständig wirksam wird, sollten die beiden Schnittstellen „HomeMatic XML-RPC API“ sowie „Remote HomeMatic-Script API“ auf „Eingeschränkt“ eingestellt werden (siehe Bild 6).

ACHTUNG: Sofern man diese Einstellung getätigt hat, muss einem bewusst sein, dass der Zugriff ggf. durch fehlerhafte Konfiguration oder bei Änderungen der Netzwerkstruktur (z. B. Router-Wechsel) nicht mehr von allen Netzwerkgeräten bzw. den verwendeten Software- oder App-Lösungen möglich ist.

In die Filterliste sollten für den bestmöglichen Schutz alle Geräte-IP-Adressen eingetragen werden, welche auf die Homematic Zentrale zugreifen müssen.

Dies sind in der Regel folgende Geräte:

- Computer (sofern Programme Zugriff benötigen: z. B. mediola NEO, contronics Homeputer)
- Smartphones/Tablets (welche per App Zugriff erhalten sollen)
- Fernzugriff per meine-homematic/CloudMatic, folgender Eintrag erforderlich: 10.192.0.0/12;
- Drittanbieter-Systeme bzw. Software (z. B. Openhab, FHem, ioBroker, IP-Symcon)

Fernzugriff auf die Homematic Zentrale

Im Folgenden wird erläutert, welche Zugriffsmöglichkeiten zur Steuerung des Homematic Systems aus der Ferne bestehen, was zu beachten ist und welche Vor- bzw. Nachteile diese Möglichkeiten haben.

Sofern man keinen Fernzugriff benötigt oder Bedenken hat, das System im Netzwerk zu betreiben, ist es nicht erforderlich, eine der folgenden Zugriffsmöglichkeiten einzurichten.

Nur bei einem eingerichteten Fernzugriff ist auch der Zugriff aus dem Internet auf die CCU2 möglich.

Allerdings kann die Zentrale, sofern sie per Netzkabel mit einem Internet-Router verbunden ist, Informationen abrufen bzw. senden. Dies sind z. B. Informationen über neue Firmware-Updates sowie der Zeitabgleich mit dem NPT-Zeitserver. Zudem können so z. B. mittels Zusatzsoftware oder per Skript auch Nachrichten (Mail/Push) versendet werden.

1. Port-Weiterleitung (Port Forwarding)

Von der Einrichtung einer Port-Weiterleitung ist strikt abzuraten!

Um per Port-Weiterleitung auf das System Zugriff zu erhalten, ist es erforderlich, die Ports der Homematic Zentrale, welche zunächst nur im internen Netzwerk zugänglich sind, über die Konfiguration im Internet-Router auch für den externen Zugriff, also aus dem Internet, zugänglich zu machen. Dies birgt ein extrem hohes Risiko, dass ggf. existierende oder auch aktuell noch unbekannt Sicherheitslücken der Zentrale für einen unbefugten Zugriff missbraucht werden könnten.

Aus diesem Grund wird hier nicht näher auf die dafür notwendige Einrichtung eingegangen.

2. VPN-Tunnel

VPN (Virtual Private Network) bzw. der VPN-Tunnel gilt als eine der sichersten Fernzugriffsmöglichkeiten. Bei einer VPN-Verbindung wird ein gesicherter, AES-verschlüsselter Tunnel zwischen einem mobilen Endgerät (z. B. Smartphone, Tablet, PC) und dem lokalen Heimnetzwerk hergestellt. Hierzu wird ein DDNS-Dienst benötigt, damit der Fernzugriff, auch bei den in Deutschland üblichen privaten DSL-Anschlüssen, mit wechselnder öffentlicher IP-Adresse jederzeit möglich ist. Diese Verbindung bietet zudem den Vorteil, dass nicht nur auf die Homematic Zentrale, sondern auch auf alle anderen im heimischen Netzwerk befindlichen Geräte zugegriffen werden kann.

Als eventuelle Nachteile kann man die für technische Laien ggf. komplex erscheinende Einrichtung sowie die bei gewünschtem Zugriff zunächst aufzubauende Verbindung aufführen.

Aufgrund der vielen verschiedenen DDNS-Anbieter und Internet-Router kann innerhalb der kostenlosen Technischen Kundenberatung von ELV kein Support zu dieser Einrichtung erfolgen. Allerdings findet man im ELV Shop unter dem Webcode #60063 hierzu eine Musteranleitung, welche eine solche Einrichtung erläutert.

3. Reverse Proxy

Die Verbindung über einen Reverse Proxy bietet sich als durchaus interessante Alternative zum VPN-Tunnel an. Allerdings ist die Einrichtung im Vergleich zum VPN-Tunnel noch komplexer und erfordert eine zusätzliche Hardware, auf welcher der Reverse-Proxy-Server aufgesetzt wird.

Warum sollte man überhaupt diesen Aufwand betreiben?

- Das vor dem Zugriff eines VPN-Tunnels erforderliche Aufbauen der Verbindung entfällt.
- Ermöglicht wie beim VPN-Tunnel nicht nur den Zugriff auf die Homematic Zentrale.
- Man vermeidet Cloud-Dienste und somit Netzwerk-Verbindungen, welche zum eigenen Netzwerk geöffnet werden.
- Apps können sich einfach wie beim VPN-Tunnel über eine IP-Adresse intern wie extern verbinden.
- Der Reverse-Proxy-Server lässt sich auf unterschiedlicher Hardware sowie vielen Betriebssystemen installieren.

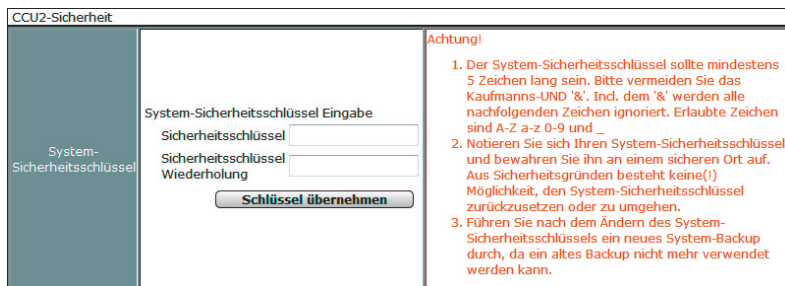


Bild 7: Eingabefelder für den persönlichen Systemsicherheitsschlüssel

Allerdings sind bei dieser Zugriffsmöglichkeit auch folgende Punkte zu berücksichtigen:

- Der Reverse Proxy sollte zwingend eine sichere Authentifizierung (Benutzername/Passwort) vornehmen, sowie eine per HTTPS gesichert Verbindung aufbauen.
- Da für einen Reverse Proxy meist OpenSource Software verwendet wird, muss diese ständig vom Nutzer aktualisiert werden. In OpenSource Software werden regelmäßig Sicherheitslücken entdeckt und meist auch zügig geschlossen.

Die Einrichtung eines Reverse-Proxy-Servers für einen sicheren Fernzugriff auf die Homematic Zentrale wird z. B. unter [1] erläutert.

4. Fernzugriff über eine Cloud-Lösung

Für Nutzer, die sich nicht mit der Einrichtung eines VPN-Tunnels oder Reverse-Proxy-Servers beschäftigen wollen, sind die Partnerlösungen von CloudMatic (meine-homematic.de) oder Orbylon empfehlenswert und einfach einzurichten.

Diese Partner bieten ähnlich wie beim VPN-Tunnel eine gesicherte Verbindung, welche auch ohne Netzwerktechnik-Kenntnisse einzurichten ist. Der Unterschied zum eigens eingerichteten VPN-Tunnel ist allerdings, dass diese getunnelte Verbindung nicht direkt zwischen dem Endgerät (Smartphone, Tablet, PC) und dem Heimnetzwerk, sondern zwischen der Zentrale und dem Cloud-Server des Partners aufgebaut wird. Das Endgerät muss sich somit zunächst mit dem Cloud-Server des Partners verbinden und erhält erst nach entsprechender Authentifizierung den Zugriff über die getunnelte Verbindung auf die Zentrale.

Vorteil ist, wie bereits erwähnt, die einfache Einrichtung, zudem sind nahezu alle Homematic Apps bereits für die Verwendung mit dem Dienst von CloudMatic (meine-homematic.de) vorbereitet. Ausnahme ist hier Orbylon, dieser Partner bringt neben dem Fernzugriff auch seine eigene App mit. Des Weiteren umfassen diese Lösungen weitere Funktionen. Welche dies im Einzelnen sind, kann unter [2] bzw. [3] nachgelesen werden.

Als eventuelle Nachteile kann man ggf. die folgenden Punkte auslegen: Trotz der mehrjährigen Partnerschaft zwischen diesen Anbietern und dem Hersteller eQ-3 sowie der ständigen Wartung und Pflege der für den Dienst erforderlichen Infrastruktur läuft die Verbindung, auch wenn diese verschlüsselt ist, über einen Dritten. Des Weiteren sind beide Lösungen nicht kostenfrei nutzbar, allerdings sind die anfallenden Kosten mit ca. 24 Euro im Jahr unter Berücksichtigung der vom Anbieter notwendigen Pflege sowie der enthaltenen Mehrwertdienste durchaus überschaubar und auch berechtigt.

Gesicherte Funkkommunikation

Neben den bereits behandelten netzwerkseitigen Sicherheitsthemen besteht im Homematic System auch die Möglichkeit, die Funkkommunikation zwischen der Homematic Zentrale und den Geräten zusätzlich zu verschlüsseln. Hierfür wird das symmetrische Kryptoverfahren AES 128 bit (Advanced Encryption Standard) verwendet.

Werkseitig sind alle Homematic Geräte bereits mit einem systemweiten Standard-Sicherheitsschlüssel versehen, sodass zumindest ab Werk alle sicherheitsrelevanten Geräte wie z. B. KeyMatic, WinMatic, Alarmsensoren und weitere Sender die gesicherte Verbindung nutzen.

Wie bei anderen Netzgeräten mit werkseitigen Passwörtern bzw. Schlüsseln auch (z. B. WLAN-Router) sollte der Schlüssel für die Funkkommunikation individuell gesetzt werden.

Folgend wird erläutert, wie dieser persönliche Schlüssel gesetzt wird, was generell bei der Verwendung zu beachten ist und welche Vor- bzw. Nachteile dieser ggf. haben kann.

1. Persönlicher Systemsicherheitsschlüssel

Bevor ein persönlicher Systemsicherheitsschlüssel eingetragen bzw. eine weitere Änderung des bereits gesetzten Schlüssels vorgenommen wird, sollte immer ein System-Backup erstellt werden. Es empfiehlt sich, diese Backup-Datei entsprechend umzubenennen, damit man bei ggf. auftretenden Problemen noch weiß, welche Sicherung der Systemkonfiguration ohne Schlüssel bzw. mit einem alten Schlüssel verwendet wurde. Die Vergabe eines persönlichen Systemsicherheitsschlüssels und damit das Überschreiben des Standardschlüssels wird in der WebUI unter „Einstellungen → Systemsteuerung → Sicherheit → System-Sicherheitsschlüssel“ durchgeführt (siehe Bild 7).

ACHTUNG: Den in der WebUI nebenstehenden und rot markierten Hinweisen sowie den hier beschriebenen sollte der Nutzer in jedem Fall Beachtung schenken, um nachfolgende Probleme zu vermeiden.

Nachdem der persönliche Systemsicherheitsschlüssel eingetragen wurde, sind die folgenden Punkte zwingend zu beachten:

- Der neu gesetzte Systemsicherheitsschlüssel muss von der Zentrale an alle bereits angelernten Geräte übertragen werden. Daher muss nach einer Schlüsseländerung zwingend in der WebUI unter Servicemeldungen kontrolliert werden, für welche Geräte noch „Konfigurationsdaten stehen zur Übertragung an“ gemeldet wird (siehe Bild 8). Werden diese Daten und somit der Schlüssel nicht an die Geräte übertragen, kann dies zur Folge

Erste Meldung Datum/Zeit	Letzte Meldung Datum/Zeit	Name	Bild	Seriennummer	Gewerk	Raum	Letzte Änderung	Servicemeldung	Aktion
18.04.2017 18:33:11	18.04.2017 18:33:11	Handsender KeyMatic:0		NEQ0683293:0			18.04.2017 18:33:11	Konfigurationsdaten stehen zur Übertragung an	Bestätigen

Bild 8: Noch zur Übertragung anstehende Konfigurationsdaten, welche durch kurze Betätigung der Config-Taste am Gerät übernommen werden müssen



Name	Typenbezeichnung	Bild	Bezeichnung	Seriennummer	Interface/Kategorie	Übertragungsmodus	Gewerke	Räume				Aktionen
Filter	Filter		Filter	Filter	Filter	Filter	Filter	Filter				
Handsender KeyMatic	HM-RC-Key4-2		HM-RC-Key4-2	NEQ0683293	BidCos-RF	Gesichert	Taster		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Einstellen Löschen Direkte Programme
Handsender KeyMatic:1 Handsender KeyMatic:2	HM-RC-Key4-2		HM-RC-Key4-2	NEQ0683293:1 NEQ0683293:2	Sender	Gesichert	Taster		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Einstellen Direkte Programme
Handsender KeyMatic:1 Tasterkanal	HM-RC-Key4-2		HM-RC-Key4-2	NEQ0683293:1	Sender	Gesichert	Taster		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Einstellen Direkte Programme
Handsender KeyMatic:2 Tasterkanal	HM-RC-Key4-2		HM-RC-Key4-2	NEQ0683293:2	Sender	Gesichert	Taster		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Einstellen Direkte Programme

Bild 9: Beispiel für ein Gerät mit seinen einzelnen Kanälen

haben, dass diese nicht mehr mit der Zentrale kommunizieren bzw. nach mehrfachem Schlüsselwechsel oder gar dem Verlust des Schlüssels sich nicht mehr in den Werkszustand versetzen lassen bzw. nicht mehr angelernt werden können.

- Für alle netzspannungsversorgten Geräte wird die Übertragung des Schlüssels, sofern die Geräte erreichbar sind, automatisch und sofort durchgeführt.
- Für batteriebetriebene Geräte gilt dies nur teilweise. Geräte wie z. B. KeyMatic, Heizkörper- sowie Wandthermostaten, welche nach dem Burst-Empfängerverfahren arbeiten, bekommen den neuen Schlüssel in der Regel auch direkt mitgeteilt.
- Für alle anderen batteriebetriebenen Sensoren wie z. B. Bewegungsmelder, Fensterkontakte sowie Wand- und Handsender, welche aus Energiespargründen nur bei einer Betätigung der Config-Taste bzw. ihrer zyklischen Statusmeldung oder bei Auslösung mit der Zentrale kommunizieren, ist es besonders wichtig, die Servicemeldungen in der WebUI zu beachten. Die ausstehenden Konfigurationsdaten und somit der neue Schlüssel werden wie in der Geräte-Bedienungsanleitung beschrieben nach einer kurzen Betätigung der Config-Taste übertragen (siehe Bild 8).
- Die Übertragung der noch ausstehenden Konfigurationsdaten sollte Gerät für Gerät erfolgen.

Allgemeine Kanaleinstellungen: NEQ0683293:1

HM-RC-Key4-2

Name: Handsender KeyMatic:1

Typenbezeichnung: HM-RC-Key4-2

Seriennummer: NEQ0683293:1

Kategorie:

Übertragungsmodus: Gesichert

Bedienbar: Standard

Sichtbar: Gesichert

Protokolliert:

Räume

Gewerke

Funktionstest

 --:--:--

Im Rahmen des Funktionstests wird geprüft, ob die Kommunikation mit dem Kanal fehlerfrei funktioniert.

Bild 10: Beispiel für die Kanaleinstellungen eines Handsenders

Nach der Betätigung der Config-Taste eines Geräts muss also kurz gewartet werden, bis die Daten übertragen wurden. Dass eine Übertragung stattfindet, ist an der schnell blinkenden Geräte-LED erkennbar. Ist die Übertragung abgeschlossen, erlischt die LED. War die Übertragung erfolgreich, verschwindet die Servicemeldung in der WebUI automatisch.

Name	Typenbezeichnung	Bild	Bezeichnung	Seriennummer	Interface	Firmware
Handsender KeyMatic	HM-RC-Key4-2		HM-RC-Key4-2	NEQ0683293	BidCos-RF	Version: 1.2
Geräteparameter						
Parameter						
Reset per Gerätetaste sperren <input type="checkbox"/>						

Bild 11: Parameter „Reset per Gerätetaste sperren“

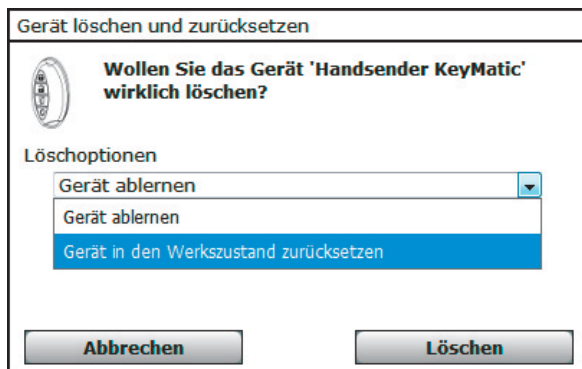


Bild 12: Korrekte Löschoption zum Entfernen des persönlichen Systemsicherheitsschlüssels

- Sofern ein umfangreiches System mit vielen Funkkomponenten betrieben wird, ist es zudem nicht ausgeschlossen, dass aufgrund der Anzahl zu übertragender Konfigurationsdaten das Sendelimit (Duty-Cycle) der Zentrale erreicht wird. Da in der aktuellen Version der Zentralen-Software eine entsprechende Hinweismeldung beim Erreichen des Duty-Cycles fehlt, ist dies für den Nutzer lediglich daran erkennbar, dass trotz der Betätigung der Geräte-Config-Taste die Daten nicht erfolgreich zu übertragen sind. In diesem Fall kann der Nutzer lediglich eine Stunde warten bzw. die Zentrale neu starten.

2. Übertragungsmodus

Damit die gesicherte Funkkommunikation mit dem persönlichen Systemsicherheitsschlüssel greift, muss je nach Gerätetyp der Übertragungsmodus in der Geräte-Konfiguration angepasst werden. Wie bereits erwähnt, ist dieser für sicherheitsrelevante Geräte bereits werkseitig konfiguriert. Möchte man diesen auch für weitere Geräte konfigurieren, ist hierzu die Geräteleiste über „Einstellungen -> Geräte“ zu öffnen. Anschließend öffnet man links durch einen Klick auf das Plus-Symbol die Unterkanäle des Geräts (siehe Bild 9). Klickt man nun in der Namensspalte auf den Kanalnamen, öffnet sich das Fenster für die allgemeinen Kanaleinstellungen (siehe Bild 10).

Hier kann unter Übertragungsmodus nun von „Standard“ auf „Gesichert“ umgestellt werden, anschließend sind je nach Gerätetyp erneut die Servicemeldungen zu beachten. Zu überlegen ist allerdings, ob es notwendig ist, jegliche auch nicht sicherheitsrelevante Gerätekommunikation auf „Gesichert“ umzustellen. Ein Handsender, welcher beispielsweise direkt mit einem Dimmer verknüpft wurde, benötigt die gesicherte Kommunikation ggf. nicht zwingend.

3. Geräte-Werksreset bei gesetztem Systemsicherheitsschlüsseln

Sofern ein persönlicher Systemsicherheitsschlüssel gesetzt wurde, ist der Geräte-Werksreset nicht mehr möglich, allerdings gilt dies nicht für alle Geräte des Homematic Systems. Primär „ältere“ Geräte-Generationen sind gegen den Reset bei gesetztem Schlüssel gesperrt. Bei neueren Geräten ist der Reset möglich, sofern der Nutzer nicht den in der Gerätekonfiguration ersichtlichen Parameter „Reset per Gerätetaste sperren“ aktiviert hat (siehe Bild 11).

4. Folgen bei Schlüsselverlust

Es ist wichtig, dass, wie im Hinweistext der WebUI sowie dem WebUI-Handbuch [4] beschrieben, eine Dokumentation über die vom Nutzer gesetzten Systemsicherheitsschlüssel geführt wird. Hierzu gehören wie eingangs beschrieben das Erstellen und entsprechende Bezeichnen der Backups sowie das Notieren des zugehörigen Sicherheitsschlüssels. Des Weiteren sollte nicht nur der aktuell vergebene Schlüssel, sondern auch die vorhergehenden Schlüssel notiert sowie die zugehörigen Backups gespeichert bleiben. Somit ist es jederzeit möglich, sofern z. B. einmal der Schlüssel nicht an ein Gerät übertragen wurde, auf einen älteren Stand zurückzuspringen. Anschließend kann das Gerät über die Geräteleiste mit der Option „Aus Homematic Zentrale löschen“ sauber abgelernt/gelöscht werden, wodurch dann nach erfolgreicher Übertragung der Konfigurationsdaten der Schlüssel aus dem Gerät entfernt wird (siehe Bild 12).

Sollte der Systemsicherheitsschlüssel trotz Beachtung der genannten Punkte nicht bekannt und auch der Reset der Geräte nicht möglich sein, bleibt lediglich das Einsenden der Geräte zum kostenpflichtigen Firmware-Flash. Die Abwicklung sowie anfallenden Kosten sind in einem Hinweisblatt, zu finden unter Webcode #60064 im ELV Shop, aufgeführt.

5. Vor-/Nachteile

Der Hauptvorteil einer verschlüsselten Kommunikation liegt klar auf der Hand. Einem unbefugten Dritten wird die Möglichkeit genommen, die Funkkommunikation von außen zu beeinflussen bzw. letztlich die Geräte zu steuern. Zudem sind Geräte, die ggf. durch Dritte entwendet wurden, aufgrund des eigenen Sicherheitsschlüssels gegen das Zurücksetzen in den Auslieferungszustand sowie das Anlernen an eine andere Zentrale gesperrt (siehe Punkt 3).

Als eventuelle Nachteile lassen sich hier ggf. die folgenden Punkte aufführen:

- Je nach Programmierung kann es zur Erhöhung des Zentralen-Duty-Cycles führen.
- Die Batterielaufzeiten werden geringfügig verkürzt.
- Schaltbefehle werden minimal verzögert ausgeführt.
- Zur Vermeidung von Kommunikationsstörungen müssen in Zentralenprogrammen Verzögerungszeiten zwischen den Aktivitäten eingefügt werden. Dieser Hinweis gilt allerdings auch bei Verwendung des Standard-Übertragungsmodus (siehe Webcode #60062 im ELV Shop).

Die aufgeführten Punkte sind durch die für die Verschlüsselung notwendige, mehrfache Kommunikation zwischen Sender und Empfänger zu begründen.

Abschließend bleibt zu sagen, dass mit diesem Artikel lediglich die wichtigsten Punkte bezüglich der Netzwerk Sicherheit der Zentrale sowie der Funk-Kommunikation zwischen der Zentrale und den Homematic Komponenten behandelt wurden. Viele weiterführende Informationen zum Thema findet man z. B. im Homematic Forum [5].

Die Sicherheit des Homematic IP Systems werden wir in einem separaten Artikel beleuchten.



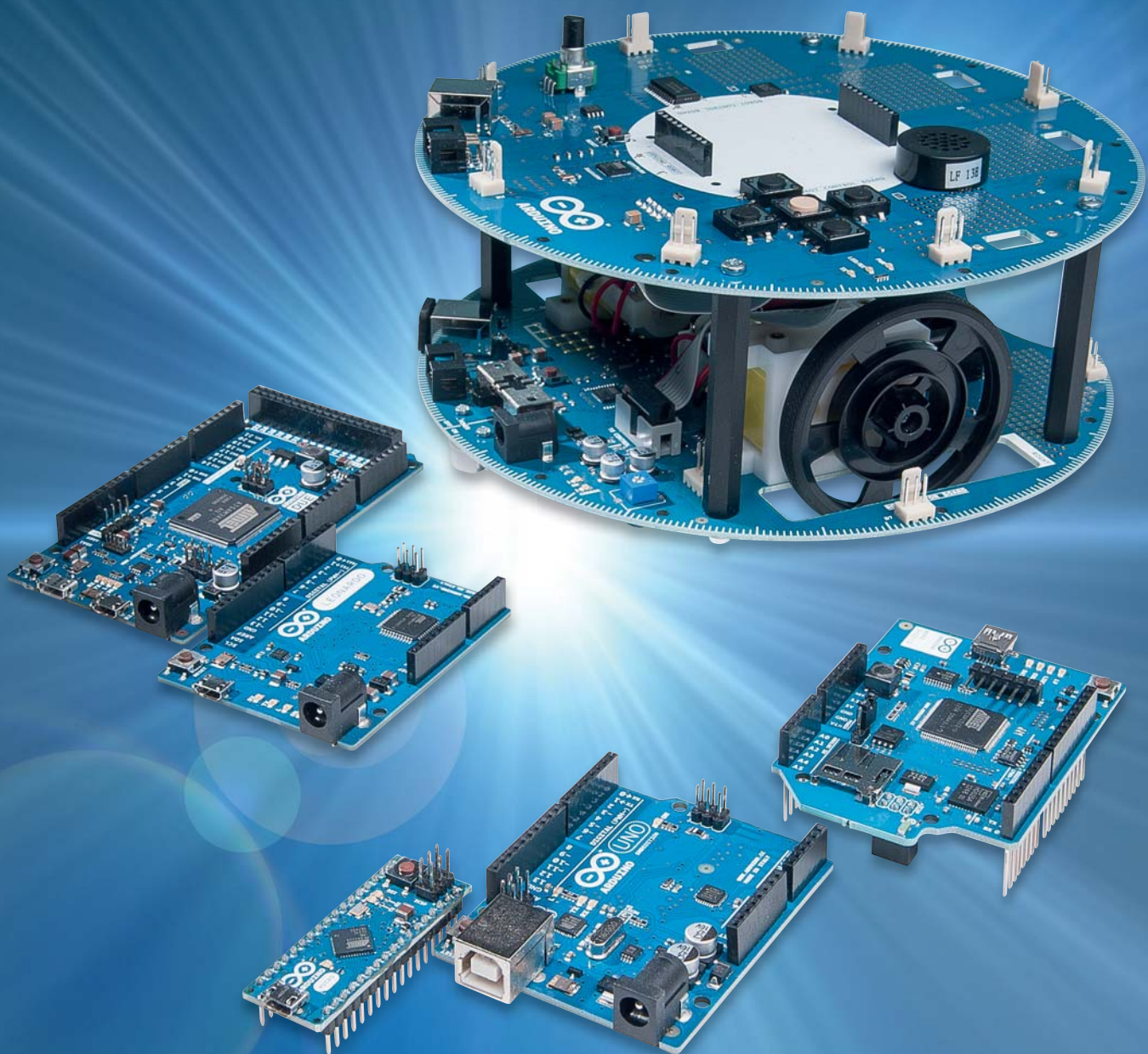
Weitere Infos:

- [1] www.elv.de: Webcode #60065
- [2] www.meine-homematic.de
- [3] www.myorbylon.de
- [4] www.elv.de: Webcode #60066
- [5] www.elv.de: Webcode #60067



Arduino verstehen und anwenden

Teil 22: I²C – der Inter-IC-Bus – Grundlagen und Anwendungen



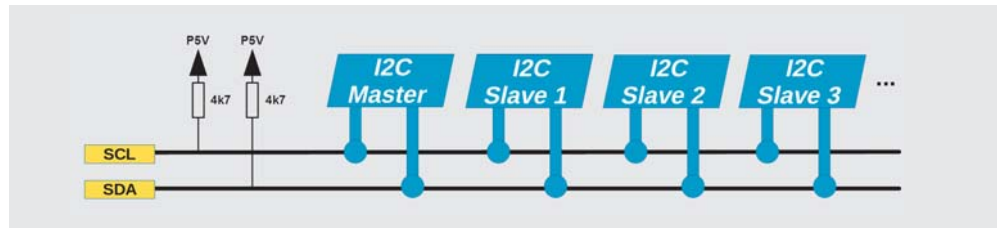


Bild 1: Prinzipieller Aufbau des I²C-Busses

Der Arduino verfügt zwar über 14 digitale I/O-Pins und diese sind für viele kleinere Projekte durchaus ausreichend. Bei größeren Praxisanwendungen stößt man jedoch schnell an Grenzen. Insbesondere wenn Shields, wie etwa das im letzten Beitrag vorgestellte MP3-Shield, benutzt werden, sind oft nur noch wenige Pins frei verfügbar, da die Shields selbst bereits viele Pins belegen.

Eine Lösung wäre, eine größere Arduino-Version einzusetzen, etwa den Arduino Mega oder den Due. Dies ist allerdings mit erheblichen Kosten verbunden und in vielen Fällen auch gar nicht notwendig.

Sogenannte Bus-Systeme bieten die Möglichkeit, umfangreiche Funktionen mit nur wenigen Pins zu steuern. Ein klassisches Beispiel ist die Verwendung eines I²C-Displays. Dieses benötigt nur zwei I/O-Pins im Gegensatz zu den sechs Pins, die ein klassisches HD44780-LCD belegt.

In diesem Beitrag soll daher der I²C-Bus im Vordergrund stehen. Weitere häufig verwendete Bus-Systeme wie etwa SPI oder One-Wire werden in späteren Beiträgen behandelt.

Der I²C-Bus

Der I²C-Bus (Inter Integrated Circuit, meist als „I-squared-C-Bus“ oder deutsch „I-Quadrat-C-Bus“ ausgesprochen) ist ein serieller Zweidraht-Bus, der vor über 30 Jahren in der Entwicklungsabteilung der Firma Philips, heute NXP, entstand. Rasch entwickelte sich das System zum Industriestandard für Steuerungs-, Diagnose- und Überwachungslösungen in unzähligen Embedded-Applikationen.

Durch die einfache Implementierung, niedrige Kosten und eine Übertragungsrate von bis zu 3,4 MBit/s hat der Bus bis heute nichts von seiner Aktualität verloren.

Ursprünglich sollte ein Bus-System entstehen, über das mehrere ICs auf einer Leiterplatte mit geringem Aufwand miteinander kommunizieren sollten. Daher wurde eine serielle Struktur bevorzugt, um mit wenigen Leiterbahnen auszukommen.

So entstand ein bidirektionaler Zweidraht-Bus in Master/Slave-Architektur mit integriertem Übertragungsprotokoll und Software-Adressierung, der nur zwei Verbindungsleitungen erfordert:

- die Taktleitung SCL (Serial Clock),
- die Datenleitung SDA (Serial Data).

Ein Mikrocontroller kann so ein ganzes Netzwerk von Komponenten mit nur zwei I/O-Pins ansteuern. Bevorzugte Anwendungen finden sich insbesondere im Bereich der Unterhaltungselektronik, z. B. in der Abstimmung von Autoradios oder TV-Geräten oder bei vollelektronischen Lautstärkereglern.

Zunächst genügte hierfür eine Übertragungsrate von lediglich 100 kBit/s. Aufgrund zunehmender Leistungsanforderungen wurde die Übertragungsrate dann aber immer weiter angehoben. Ein großer Vorteil des I²C-Busses besteht darin, dass auch langsamere Komponenten am Bus betrieben werden können. Hierfür kann das sogenannte Clock-Stretching eingesetzt werden (s. u.).

Inzwischen ist der I²C-Bus nicht mehr auf einzelne Platinen beschränkt. Vielmehr kommt er auch in größeren Systemen mit mehreren Boards zum Einsatz. Die einfache Steuerungssoftware macht den Bus sehr flexibel. Da keine festen Taktzeiten eingehalten werden müssen, können sowohl langsame als auch sehr schnelle Busteilnehmer simultan betrieben werden. Auf der Softwareseite ist so auch der Einsatz langsamerer Programmiersprachen wie etwa Python möglich.

Elektrischer Anschluss und Takt

I²C ist als Master-Slave-Bus konzipiert. Ein Datentransfer wird immer durch einen Master gestartet. Anschließend reagiert der über seine Adresse angesprochene Slave auf die Anfrage. Im sogenannten Multimaster-Mode können aber auch mehrere Master an einem Bus betrieben werden. Hier können dann auch zwei Master direkt miteinander kommunizieren, indem ein Gerät kurzzeitig als Slave arbeitet. Die Zugriffsregelung auf den Bus ist über eine entsprechende Spezifikation detailliert geregelt.

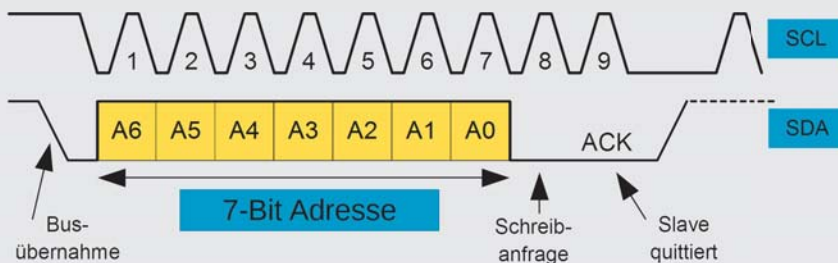
Bild 1 zeigt, wie die einzelnen Buskomponenten hardwaremäßig miteinander verbunden werden. Beide Busleitungen (Takt und Daten) liegen mit den Pull-up-Widerständen (typischerweise 4,7 bis 10 k Ω) an der Versorgungsspannung V_{DD}, die üblicherweise 5 V beträgt. Sämtliche daran angeschlossene Geräte haben Open-Collector-Ausgänge. Zusammen mit den Pull-up-Widerständen ergibt sich so eine Wired-AND-Schaltung. Der High-Pegel der Bussignale sollte mindestens 0,7×V_{DD} betragen, der Low-Pegel soll höchstens bei 0,3×V_{DD} liegen. Falls es zu Übertragungsfehlern kommt, ist es immer eine gute Idee, wenn man diese Spannungen mit einem Oszilloskop nachmisst und eventuell die Pull-up-Widerstände anpasst. Einige Controller haben auch bereits interne Pull-ups eingebaut, sodass die externen Widerstände entfallen können. Im Zweifelsfall hilft hier ein Blick ins Datenblatt weiter.

Der I²C-Bus arbeitet mit positiver Logik. Das bedeutet, dass ein High-Pegel auf der Datenleitung einer logischen „Eins“, der Low-Pegel einer „Null“ entspricht.

Der Bustakt wird immer vom Master ausgegeben. Für die verschiedenen Modi ist jeweils ein maximal erlaubter Bustakt vorgegeben. In der Regel können

Bild 2: Aufbau eines I²C-Telegramms

	Adresse				Geräte Subadresse			R/W	ACK	Daten							Stopp		
Start	0	1	0	0	x	x	x	1	ACK	x	x	x	x	x	x	x	x	x	Stopp

Bild 3: I²C-Timing

aber auch beliebig langsamere Taktraten verwendet werden, falls diese vom Master-Interface unterstützt werden. Einige ICs (z. B. Analog-digital-Umsetzer) benötigen jedoch eine bestimmte minimale Taktfrequenz, um ordnungsgemäß zu funktionieren. Die folgende Tabelle listet die gängigsten Taktraten auf:

100-kHz-Takt	Standard-Mode
400-kHz-Takt	Fast-Mode
1,0-MHz-Takt	Fast-Mode-Plus
3,4-MHz-Takt	High-Speed-Mode

Wenn der Slave mehr Zeit benötigt, als durch den Takt des Masters vorgegeben ist, kann er zwischen der Übertragung einzelner Bytes die Taktleitung auf „low“ halten. Über dieses im letzten Abschnitt bereits erwähnte Clock-Stretching kann der Mastertakt reduziert werden, falls dies erforderlich sein sollte.

In einem I²C-System sind Daten nur gültig, wenn sich ihr logischer Pegel während einer Clock-High-Phase nicht ändert. Ausnahmen von dieser Regel sind das Start- und Stoppsignal. Das Startsignal wird durch eine fallende Flanke auf SDA gekennzeichnet, während SCL „high“ ist. Das Stoppsignal dagegen wird durch eine steigende Flanke auf SDA signalisiert, die SCL-Leitung bleibt dabei auf High-Pegel.

Eine Dateneinheit besteht aus 8 Datenbits und wird häufig auch als Oktett bezeichnet. Ein Oktett kann entweder als Wert oder als Adresse interpretiert werden. Dazu kommt noch ein ACK-Bit als Bestätigung (von engl. Acknowledge). Dieses wird vom Slave durch einen Low-Pegel auf der Datenleitung erzeugt.

Protokoll und Adressierung

Eine Standard-I²C-Adresse ist das erste vom Master gesendete Byte, wobei die ersten sieben Bit die eigentliche Adresse darstellen und das achte Bit (R/W-Bit) dem Slave mitteilt, ob er Daten vom Master empfangen soll (LOW) oder Daten an den Master zu übertragen hat (HIGH). In I²C-Systemen ist daher ein Adressraum von 7 Bit verfügbar. Da $16 \text{ der } 2^7 = 128$ möglichen Adressen für Sonderzwecke reserviert

sind, können so bis zu 112 Teilnehmer an einem Bus betrieben werden. Dazu hat jedes I²C-fähige IC eine vom Hersteller festgelegte Adresse, von der häufig drei Bits, die sogenannte Subadresse, über drei Steuerepins individuell festgelegt werden können (siehe Bild 2 und 3). In diesem Fall können bis zu acht ICs des gleichen Typs an einem I²C-Bus betrieben werden. Im Praxisteil wird dies anhand eines Temperatursensors vom Typ LM75 genauer erläutert.

Für Hobbyanwendungen sind 112 Teilnehmer sicherlich in den meisten Fällen ausreichend. Bei großen, professionellen Systemen dagegen kann es durchaus zu Adressknappheit kommen. Deshalb wurde eine 10-Bit-Adressierung eingeführt. Sie ist abwärtskompatibel zum 7-Bit-Standard durch Nutzung von 4 der 16 reservierten Adressen. Beide Adressierungsarten sind gleichzeitig verwendbar, sodass nun bis zu 1136 Komponenten auf einem Bus erlaubt sind.

Die Übertragung beginnt mit einem Startsignal des Masters. Dann folgt die Adresse. Diese wird durch das ACK-Bit des angesprochenen Slaves bestätigt. Abhängig vom R/W-Bit werden nun Daten byteweise geschrieben oder gelesen. Beim Schreiben wird das ACK vom Slave gesendet, beim Lesen vom Master. Eine Übertragung wird durch das Stoppsignal beendet.

Die Datenübertragung startet mit dem höchstwertigen Bit (MSB first). Für den High-Speed-Mode wird zuerst im Fast- oder Standard-Mode ein Master-Code geschickt, bevor auf die erhöhte Frequenz umgeschaltet wird.

Anwendungen

Einer der wichtigsten Vorteile von Bus-Systemen wie I²C ist, dass ein Mikrocontroller ein ganzes Netzwerk an integrierten Schaltungen mit nur zwei I/O-Pins und einfacher Software ansteuern kann.

Systeme mit einer möglichst geringen Anzahl von erforderlichen I/O-Pins sind deshalb so vorteilhaft, da ein erheblicher Teil der Kosten einer integrierten Schaltung und der verwendeten Leiterplatte von der Größe des IC-Gehäuses und der Anzahl der Pins abhängt. Ein großer Chip mit vielen Pins benötigt mehr Platz auf der Leiterplatte. Zudem stellen zusätzliche Verbindungen immer ein Ausfallrisiko dar und erfordern so einen erhöhten Prüfaufwand. All das steigert die Entwicklungs-, Produktions- und Testkosten.

Obwohl das System langsamer ist als neuere Busse, ist die I²C-Schnittstelle wegen des geringen Aufwands vorteilhaft für Peripheriegeräte, die keine hohen Datenraten erfordern. Häufig wird sie für die Übertragung von Steuer- und Konfigurationsdaten verwendet. Beispiele sind Lautstärkereglern, Analog-digital- oder Digital-analog-Wandler mit niedriger Abtastrate, Echtzeituhren, kleine, nichtflüchtige Speicher oder



bidirektionale Schalter und Multiplexer. Auch elektronische Sensoren haben oft einen Analog-digital-Wandler mit I²C-Schnittstelle integriert.

In der folgenden Tabelle sind einige der am meisten verwendeten I²C-fähigen Chips zusammengefasst.

IC / Komponente	Funktion
LM75	Temperatursensor
BMP180	Barometrischer Luftdrucksensor
DS1307	Echtzeituhr mit Batteriepufferung
MAX127	A/D-Wandler
PCF8574	Portexpander
PCF8583	Echtzeituhr
LC-I2C	4x14-Segment-LC-Display
AZDelivery-Display	OLED-Graphik-Display 0,96"
AT24C256	EEPROM

Natürlich ist diese Tabelle bei Weitem nicht vollständig, sie liefert jedoch einen guten Überblick über die wichtigsten und am häufigsten verwendeten I²C-Bausteine.

Ein weiterer Vorteil des Busses ist, dass Bauelemente während des Betriebs zum Bus hinzugefügt oder entfernt werden können (sog. Hot-Plug-Fähigkeit). Allerdings muss diese Eigenschaft durch besondere Vorkehrungen in der Programmierung berücksichtigt werden. Ansonsten kann es auch zu unerwünschten Programmabbrüchen kommen, wenn Elemente vom Bus getrennt werden.

Eine gewisse Bedeutung hat das I²C-Protokoll auch im Chipkartenbereich. So wurde das System für Krankenversichertenkarten eingesetzt. Unter den Kontaktflächen der Chipkarte befand sich ein einfaches I²C-EEPROM, das vom Kartenleser ausgelesen und beschrieben werden konnte. Allerdings werden hier zunehmend auch andere Technologien eingesetzt.

Sensoren am I²C-Bus

Zu den wichtigsten Anwendungen des I²C-Busses zählt die Erfassung von Sensorwerten. Hierfür steht eine fast unüberschaubare Vielfalt von Komponenten zur Verfügung. So lassen sich praktisch alle physikalischen Größen mit I²C-Sensoren erfassen. Im nichtprofessionellen Bereich haben sich allerdings einige Bauelemente besonders etabliert. Die wichtigsten davon sollen als Anwendungsbeispiele für das Bus-System im Folgenden vorgestellt werden.

Präzise Temperaturmessung

Eine der einfachsten Möglichkeiten, Temperaturen zu erfassen, ist die Verwendung von NTC-Sensoren (Negative Temperature Coefficient). Diese Methode ist allerdings mit mehreren Nachteilen verbunden.

Zunächst sind diese einfachen Bauelemente herstellungsbedingt mit großen Toleranzen behaftet. Deshalb muss man jeden Sensor individuell kalibrieren, wenn eine gewisse Genauigkeit erreicht werden soll. Was im Hobbybereich durchaus noch möglich ist, wäre bei professionellen Anwendungen viel zu aufwendig.

Ein weiterer Nachteil ist, dass die Messungen sehr ungenau werden können, sobald zwischen dem Sensor und dem Controller bzw. dem Arduino größere Distanzen liegen. Eine Ferntemperaturerfassung wird mit einfachen analogen Sensoren sehr unzuverlässig.

Thermospannungen, Leitungswiderstände oder elektromagnetische Einstreuungen führen zu erheblichen Messfehlern. Digitale Sensoren sind hier deutlich weniger stör anfällig. Der Messwert wird damit direkt am Ort der Messung in ein digitales Signal umgewandelt. Dieses ist gegenüber den oben genannten Einflüssen wesentlich unempfindlicher. Auf diese Weise kann man auch größere Distanzen zwischen dem Sensor und einer zentralen Messstation überbrücken.

Ein weit verbreiteter Sensortyp mit integrierter A/D-Wandlung ist der LM75. Dieses Bauelement wandelt die aktuell gemessene Umge-

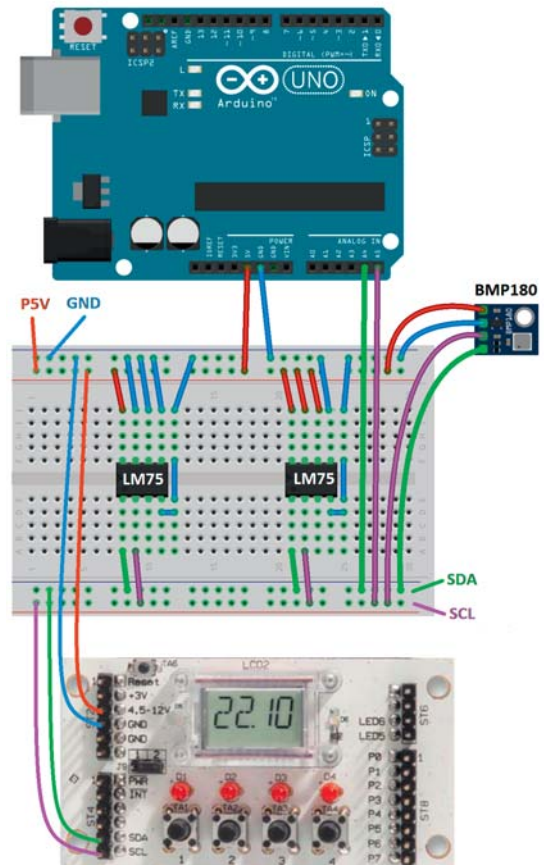


Bild 4: Sensoren und Display via I²C-Bus am Arduino

bungstemperatur direkt in ein I²C-Signal um. Damit wird es auch sehr einfach möglich, mehrere Sensoren mit einem einzigen Controller auszuwerten. Auf diese Weise kann man auch eine Vielzahl von Sensoren an einen Arduino anschließen, ohne dass die Analogeingänge belegt werden. Diese bleiben so für andere Anwendungen frei.

Bild 4 zeigt, wie zwei LM75-Sensoren parallel am I²C-Bus eines Arduinos betrieben werden können. Zusätzlich ist hier bereits auch noch ein I²C-fähiges LCD-Modul und ein Drucksensor eingezeichnet. Diese beiden Komponenten werden weiter unten genauer erläutert.

Die beiden Temperatursensoren müssen auf unterschiedliche Adressen konfiguriert werden. Dazu werden die Adresspins A0 bis A2 in geeigneter Weise mit Vcc oder GND verbunden. Bild 5 zeigt ein Beispiel.

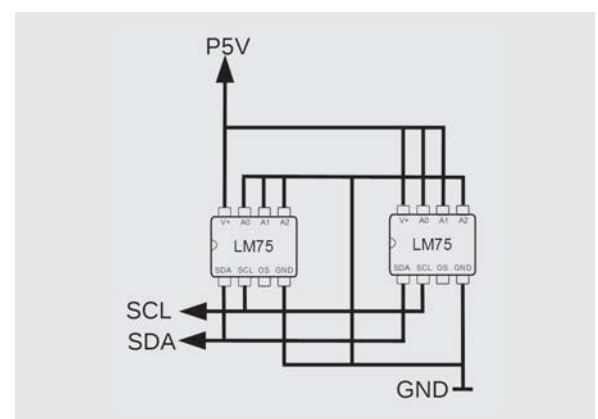


Bild 5: Einstellung von I²C-Adressen am LM75



Daraus ergeben sich die folgenden Adressen für die beiden Bausteine:

LM75 – 1:

A2 = 0, A1 = 0, A0 = 0

LM75 – 2:

A2 = 0, A1 = 1, A0 = 1

Da der fest eingestellte Teil der Adresse 1001 lautet, ergeben sich die vollständigen Adressen zu:

LM75 – 1:

1001000 = 48 hex

LM75 – 2:

1001011 = 4B hex

Sollen weitere Sensoren angeschlossen werden, müssen für diese natürlich andere Adressen gewählt werden.

I²C-fähiger Luftdrucksensor

Wie bereits aus der oben stehenden Tabelle ersichtlich ist, steht mit dem BMP180 auch für die Messung des Luftdrucks ein I²C-fähiger Baustein zur Verfügung. Dieser kann problemlos gemeinsam mit den beiden LM75-Sensoren am Arduino betrieben werden, da seine Adresse fest auf den Wert

1110111 = 77 hex

eingestellt ist.

Zusätzlich zum Luftdruck liefert dieser Sensor auch noch einen präzisen Temperaturmesswert. Zusammen mit den beiden LM75-ICs stünden damit dann bereits drei Temperatursensoren zur Verfügung, mit denen Messwerte an verschiedenen Orten erfasst werden könnten. Beispielsweise ist es so möglich, die Temperatur in einem Wohnraum, im Keller und im Außenbereich zu erfassen.

Einfacher Anschluss von Displaymodulen

Auch Displaymodule verfügen häufig über einen I²C-Anschluss. So kann beispielsweise ein alphanumerisches LC-Display über lediglich zwei aktive Pins gesteuert werden. Ein solches Display wurde bereits im letzten Beitrag zu dieser Artikelserie eingesetzt. Beim Audio-Player waren nur noch wenige freie Pins verfügbar, sodass der I²C-Bus dort seine Vorteile voll ausspielen konnte.

Natürlich könnte ein derartiges Display auch hier zum Einsatz kommen, allerdings soll an dieser Stelle eine andere Version vorgestellt werden. Das ELV I²C-Displaymodul bietet eine vierstellige 14-Segmentanzeige, die bestens zur Darstellung von Messwerten geeignet ist. Möchte man lediglich Zahlenwerte ausgeben, ist ein derartiges Display voll-

kommen ausreichend. Die Platine ist unter anderem auch direkt als Arduino-Shield einsetzbar.

Der Anschluss des Displays ist ebenfalls aus [Bild 4](#) ersichtlich.

Die I²C-Adresse des ELV Displays ist auf

0111011 = 3B hex

eingestellt und kann ebenfalls nicht verändert werden.

Test des Bus-Systems

Insbesondere wenn eine Vielzahl von Komponenten an einem Bus angeschlossen sind, ist es empfehlenswert, das Gesamtsystem vor der Inbetriebnahme zu testen.

Die folgende Tabelle fasst alle im System vorhandenen Adressen zusammen:

	binär	hex
LM75 – 1	0b1001000	0x48
LM75 – 2	0b1001011	0x4B
BMP180	0b1110111	0x77
I²C-Display	0b0111011	0x3B

Alle am Bus vorhandenen Sensoren können also aufgrund ihrer individuell verschiedenen Adressen problemlos angesprochen werden.

Um die Funktion des Busses zu testen, kann ein sogenannter I²C-Scanner verwendet werden.

Für den Arduino existiert dazu ein spezielles Programm, das unter [\[1\]](#) aus dem Internet geladen werden kann. Sind die Komponenten korrekt angeschlossen, zeigt der Scanner die jeweiligen Bus-Adressen im seriellen Monitor an ([Bild 6](#)).

Klimastation mit I²C-Komponenten

Die hier vorgestellten Komponenten ermöglichen den Aufbau einer präzisen und flexibel einsetzbaren Klimastation. Sind alle Komponenten erst einmal mit dem Arduino verbunden, fehlt nur noch ein passendes Auswertungsprogramm.

Das ELV I²C-Displaymodul verfügt neben der LCD-Anzeige auch noch über vier LEDs. Durch eine entsprechende Zuordnung und Beschriftung kann man damit jeden aktuell dargestellten Messwert kennzeichnen. Für die Anwendung als Klimastation wurde folgende Zuordnung gewählt:

LED 1: aktueller Luftdruck (P)

LED 2: Innentemperatur 1 (T1)

LED 3: Innentemperatur 2 (T2)

LED 4: Außentemperatur (T3)

Da die LEDs ebenfalls über den I²C-Bus angesteuert werden können, werden keine zusätzlichen Arduino-Pins belegt.

Um das Display anzusteuern, ist wieder eine passende Library erforderlich. Diese kann unter [\[2\]](#) von der ELV Produktseite des Moduls geladen werden. Das Programm dazu ist im folgenden Listing in verkürzter Form dargestellt. Der vollständige Code kann dem Download-Paket (siehe „Download-Paket zum Artikel“ am Ende des Beitrags) entnommen werden.

Zunächst werden im Sketch alle erforderlichen Bibliotheken eingebunden. Dann erfolgt die Deklaration der Variablen:

P: Luftdruck in mbar

T1: z. B. Innentemperatur 1

T2: z. B. Innentemperatur 2

T3: z. B. Außentemperatur

Im Set-up werden die einzelnen Module (Wire, LCD und Serial) gestartet. In der Hauptschleife werden dann die Werte aus den jeweiligen Modulen der Reihe nach ausgelesen und sowohl an das Displaymodul als auch an die serielle Schnittstelle ausgegeben. Weitere Details dazu können bei Bedarf in [\[3\]](#) nachgelesen werden.

Wenn die Klimastation ohne Verbindung zu einem PC laufen soll, kann auf die Ausgabe zur seriellen Schnittstelle verzichtet werden und

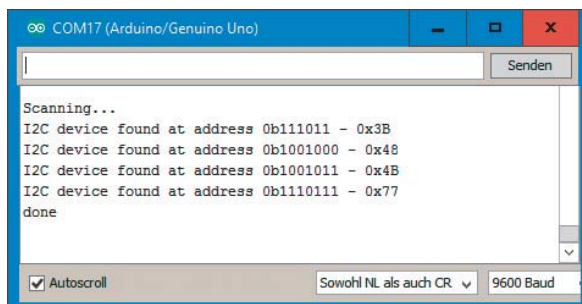


Bild 6: Der I²C-Scanner findet vier Komponenten am Bus



die entsprechenden Programmzeilen können entfallen. Die Leuchtdioden des Displays werden dabei so angesteuert, dass für jeden Ausgabewert eine einzelne LED aufleuchtet. Zusammen mit einer entsprechenden Frontplattenbeschriftung kann man so die angezeigten Werte den jeweiligen Sensoren leicht zuordnen.

```
// I2C_climate_station.ino

#include "TwoWireLCD.h"
#include <Wire.h>
#include <Adafruit_BMP085.h>
#define LM75_1 0b1001000 // LM75 #I 7-bit adresse
#define LM75_2 0b1001011 // LM75 #II 7-bit adresse
Adafruit_BMP085 bmp;
float P, T1, T2, T3;

void setup()
{ // Initialize I2C and I2C-LCD-Moduls
  Wire.begin(); LCD.begin(); Serial.begin(115200);
  while (!Serial) {}
  Serial.println(F("Init I2C"));
  // Initialisierung des Sensors
  if (!bmp.begin())
  { Serial.println("No BMP085 sensor found!");
    while (1) {}
  }
}

void loop()
{ byte msb_1,lsb_1=0, msb_2,lsb_2=0;
  float Degrees_1=0, Degrees_2=0;
  // read LM75_1
  Wire.beginTransmission(LM75_1);
  Wire.write((byte)0x00);
  Wire.endTransmission();

  Wire.requestFrom(LM75_1, 2);
  while(Wire.available() < 2);
  msb_1 = Wire.read();
  lsb_1 = Wire.read();
  ...
  // read LM75_2
  ...

  P = bmp.readPressure();
  T1 = bmp.readTemperature();
  T2 = Degrees_1;
  T3 = Degrees_2;

  Serial.println(P); Serial.println(T1);
  Serial.println(T2); Serial.println(T3);

  // show pressure P in display
  ...
  // show temperature T1 in display
  ...
  // show temperature T2 in display;
  ...
  // show temperature T3 in display
}
```

Wenn man den Aufbau in ein geeignetes Gehäuse einbaut, hat man eine praxistaugliche Klimastation vor sich. Bild 7 zeigt einen Vorschlag dazu. Neben der Anzeige der Werte im Display kann man diese auch



Bild 7: Eine Klima-Messstation mit I²C-Sensoren im Eigenbau

noch über die USB-Schnittstelle und den seriellen Monitor aufzeichnen. Man erhält so einen recht komfortablen Temperatur- und Luftdrucklogger für kontinuierliche Wetterbeobachtungen.

Ausblick

Nachdem in diesem Beitrag Grundlagen und erste Anwendungen des I²C-Busses vorgestellt wurden, folgen im nächsten Artikel weitere Praxisanwendungen. Dabei soll insbesondere der Einsatz von Portexpandern im Vordergrund stehen. Diese nützlichen Bauelemente gestatten es, die I/O-Ports eines Mikrocontrollersystems zu erweitern. So können alleine mit den beiden seriellen Signalen SCL und SDA nahezu beliebig viele weitere „Portpins“ angesprochen werden. Damit ist es dann sogar möglich, ein alphanumerisches, HD44780-kompatibles Flüssigkristall-Display mit kostengünstigen Bausteinen an den I²C-Bus anzuschließen und so mit lediglich zwei Pins das gesamte Display zu steuern. **ELV**



Weitere Infos:

- [1] <http://playground.arduino.cc/Main/I2cScanner>
 - [2] <https://www.elv.de>: Webcode #10094
 - [3] Grundlagen zur elektronischen Schaltungstechnik finden sich in der E-Book-Reihe „Elektronik!“ (www.amazon.de/dp/B000XNCB02)
- FRANZIS AVR-Mikrocontroller in C Programmieren, Best.-Nr. CN-09 73 52
 - Elektor-Praxiskurs AVR-XMEGA-Mikrocontroller, Best.-Nr. CN-12 07 62
 - FRANZIS Arduino-Projects-Lernpaket, Best.-Nr. CN-11 51 22
 - FRANZIS Physical Computing, Best.-Nr. CN-12 21 81
 - FRANZIS Lernpaket Motoren & Sensoren mit Arduino, Best.-Nr. CN-12 74 74

Preisstellung April 2017 – aktuelle Preise im ELV Shop

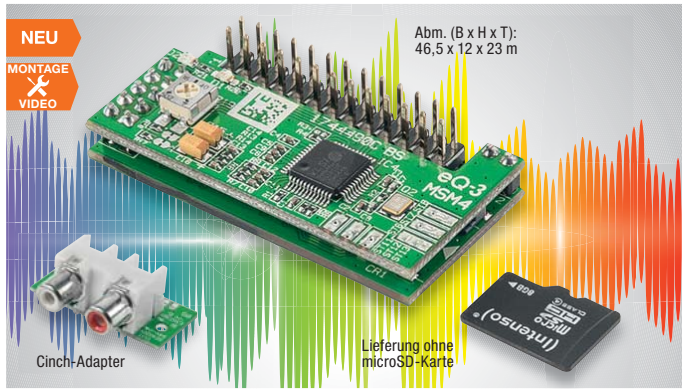
Empfohlene Produkte	Best.-Nr.	Preis
Arduino UNO	CN-10 29 70	€ 27,95
Komplettbausatz		
ELV I ² C-Bus-Displaymodul I ² C-LCD	CN-09 92 53	€ 13,95

Alle Arduino-Produkte wie Mikrocontroller-Platinen, Shields, Fachbücher und Zubehör finden Sie unter: www.arduino.elv.de

Download-Paket zum Artikel:

Die Sketche und Beispieldateien zu diesem Artikel können unter www.elv.de: Webcode #10095 heruntergeladen werden.

Vielseitig einsetzbar – MP3-Soundmodul MSM4



Das MSM4 ist ein sehr kompakter MP3-Player für den universellen Einsatz. Es ist in der Lage, bis zu 10 MP3-Soundfiles in verschiedenen Konfigurationen von einer microSD-Karte wiederzugeben. Die Auswahl der MP3-Files erfolgt wahlweise durch externe Schaltsignale oder Schaltkontakte.

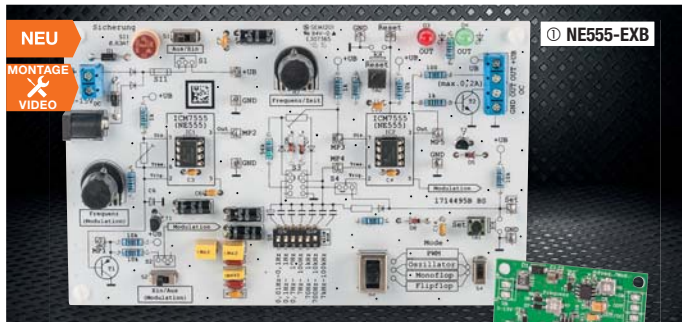
- Ein integrierter Verstärker erlaubt den direkten Lautsprecheranschluss, die Auskopplung eines Stereo-Signals ist über eine Cinch-Buchsenplatine möglich.
- Sehr schlankes Design für einfachen Einbau in eigene Geräte
- Einfacher File-Abruf durch einfache, gegen Masse schaltende Tasteringänge, auch durch Transistoren, MOSFETS, Optokoppler oder Mikrocontroller ansteuerbar
- Integrierter Mono-Leistungsverstärker: 0,5 W an 8 Ω
- Sehr einfache Soundfile-Zuordnung für den Titelabruf
- Abspielen kompletter Playlists möglich (bis zu 10 Playlists mit je bis zu 255 Soundfiles)

- Wiederholfunktion, Endlosschleife, Sprungfunktion, mehrere Funktionen kombinierbar
- Verschiedene Abspielmodi
- Autoplay-Funktion – spielt eine definierte Playlist ab, sobald der Player eingeschaltet wird
- Zufallswiedergabe oder der Reihe nach
- Lautstärkeinstellung über Tasten und per Trimpoti

Komplettbausatz MP3-Sound-Modul MSM4
CN-15 08 41 € 24,95

TECHNISCHE DATEN	
Versorgungsspannung	4,5–5,5 VDC
Stromaufnahme (Betrieb/Stand-by)	50–500 mA/ 2,5 mA (Idle-Modus), 50 µA (Sleep-Modus)
NF-Ausgangsleistung	0,5 W an 8 Ω

Experimentieren und anwenden – Experimentierboard NE555-EXB



Der altbekannte NE555 ist auch heute noch ein aktuelles und vielseitig einsetzbares Bauelement, das sich immer noch problemlos in eigenen Schaltungen einsetzen lässt. Das NE555-Experimentierboard erlaubt es, sich mit den Möglichkeiten des vielseitigen Schaltkreises ausführlich zu beschäftigen und die gefundene Lösung dann auch auf dem optional erhältlichen Anwenderboard anzuwenden.

- Das großzügig dimensionierte und mit praktischer und hilfreicher Bedruckung versehene Experimentierboard ermöglicht das Experimentieren und die Schaltungsdimensionierung der verschiedensten Schaltungen mit dem NE555. Die erarbeitete Lösung können Sie auf das Anwenderboard NE555-AWS (nicht inkl.) übertragen und dieses kompakte Board dann in seine eigene Applikation integrieren.
- NE555-Experimentierboard mit 2x ICM7555 (CMOS-Version) und Grundperipherie für die 4 am häufigsten eingesetzten Grundsaltungen:

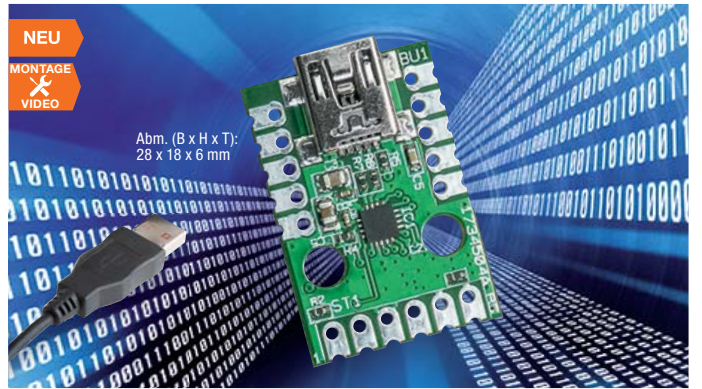
- Oszillator, PWM-Modulator, Monoflop und Flipflop
- 2 Schaltausgänge: direkt über Schutzwiderstand (100 mA) und Open-Collector-Transistorausgang (0,2 A)
- Spannungsversorgung über Batterie oder Steckernetzteil möglich

1 Komplettbausatz Experimentierboard NE555-EXB
CN-15 08 07 € 27,95

TECHNISCHE DATEN	
Versorgungsspannung	3–15 Vdc
Abm. (B x H x T) NE555-EXB	160 x 100 mm

2 Anwenderschaltung NE555-AWS (SMD-Bauteile vorbestückt)
Abm. (B x H x T): 51 x 34 mm
CN-15 08 08 € 9,95

Einfach seriell angebunden – USB-Modul UM2102N



Das kompakte USB-Umsetzermodule ermöglicht die einfache Anbindung serieller Datenschnittstellen, z. B. von UART-Mikrocontroller-Schnittstellen an ein PC-Betriebssystem. Dabei erfolgt die Verbindung über USB. Ein passender Treiber sorgt wieder für die Umsetzung in eine serielle Datenschnittstelle.

Das USB-Modul UM2102N ist der moderne Nachfolger des bereits seit dem Jahr 2010 erfolgreichen USB-Moduls UM2102.

Kern des Moduls ist der modernisierte USB-UART-Umsetzer-Chip CP2102N, der nur noch eine sehr minimale äußere Beschaltung erfordert und sehr vielseitig einsetzbar ist.

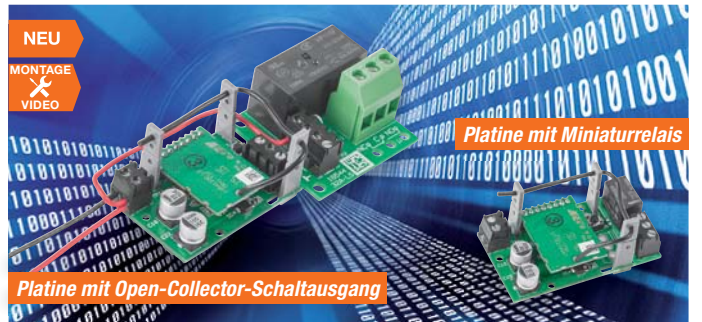
Das kompakte Modul entspricht in seinen Abmessungen dem Vorgängertyp UM2102 und kann sehr einfach über Lötflächen oder Stiftheisten in eigene Applikationen eingebunden werden.

- USB-2.0-UART-Umsetzer mit aktuellem Chip CP2102N
- Betrieb an 5 V und 3,3 V möglich
- 2 Spannungsausgänge für eigene Schaltungen: 5 V/500 mA und 3,3 V/100 mA
- Signale: RXD, TXD, RTS, CTS
- Remote-Wakeup- und Suspend-Pins nach außen geführt
- Treiber für VCP und USBxpress einsetzbar

Komplettbausatz USB-Modul UM2102N
CN-15 09 52 € 5,95

TECHNISCHE DATEN	
Bezeichnung	UM2102N
Schnittstelle	USB 2.0
Übertragungsrate	300–3.000.000 Baud
Versorgungsspannung	USB-powered
Spannungsausgang	5 V/max. 500 mA 3,3 V/max. 100 mA

Universeller Aktor für Kleinspannung – Schaltplatine HmIP-PCBS



Der kompakte Schaltaktor für das Homematic IP System vereinfacht Eigenbauprojekte im Kleinspannungsbereich, er stellt dabei entweder einen Open-Collector-Schaltausgang oder einen potentialfreien Relais-Umschaltkontakt bereit.

Die kleine Schaltplatine lässt sich mit anderen Homematic IP Sendern steuern und schaltet dabei Ströme bis zu 0,5 A am Open-Collector-Ausgang und bis zu 1 A am potentialfreien Wechslerkontakt des Kleinspannungsrelais. Reicht hingegen die Schaltleistung des Miniaturrelais nicht für die vorgesehene Aufgabe aus, kann der Aktor auch sehr einfach mit dem Relais-Schaltmodul RSM1 verbunden werden. Die Klammern wurden deshalb so positioniert und belegt, dass beide Schaltungen direkt aneinanderpassen.

Wie alle anderen Homematic IP Geräte lässt sich auch der Aktor über die Zentrale des Homematic Systems ansteuern, u. a. mit dem Vorteil, dass hier eine sehr hohe Funkreichweite erzielbar ist.

- Kompakte Schaltplatine für die Einbindung in das Homematic IP System
- Open-Collector-Schaltausgang bis 0,5 A
- Relais-Schaltausgang, bis zu 1 A, potentialfreier Wechslerkontakt
- Passend zur Relaisplatine RSM1
- In das Homematic System einbindbar

Komplettbausatz Schaltplatine
CN-15 07 75 € 22,95

TECHNISCHE DATEN	
HM-Bezeichnung	HmIP-PCBS
Versorgungsspannung	5–25 Vdc
Funkreichweite	bis zu 350 m (Freifeld)
Abm. (B x H x T)	48 x 21 x 28 mm (mit Relais)

Komplettbausatz Relais-Schaltmodul RSM1
CN-15 06 08 € 8,95

Mehr Wissen in Elektronik – Das ELV Journal Online



Laden Sie sich unsere Platinenvorlagen (auch doppelseitige) bequem per Internet herunter! Wir halten alle aktuellen Platinenvorlagen als PDF-File auf unserer Website des ELV Journal für Sie bereit. Damit stehen in der Regel alle, auch die übergroßen und doppelseitigen Platinenvorlagen, der aktuellen Ausgaben des ELV Journal zur Verfügung. Sie können gegenüber der herkömmlichen Folie beliebig oft eingesetzt, auf dem von Ihnen gewünschten Medium ausgedruckt und bequem jahrelang im Computer archiviert werden. Die Daten liegen im gebräuchlichen PDF-Format vor, sind also mit jedem Acrobat Reader ab Version 3.0 zu öffnen und auszudrucken. Beste Ergebnisse erreichen Sie beim Ausdruck auf den von uns angebotenen Laser- und Inkjet-Folien. Hiermit sind besonders hohe Maßhaltigkeit und Kantenschärfe erreichbar.

www.elvjournals.de

Wir wollen es wissen! Ihre Anwendungen und Applikationen

Welche eigenen kreativen Anwendungen und Applikationen haben Sie mit den ELV Haustechnik-Systemen, aber auch anderen Produkten und Bausätzen realisiert? Schreiben Sie uns, fotografieren Sie Ihre Applikation, berichten Sie uns von Ihren Erfahrungen und Lösungen. Die interessantesten Anwendungen werden redaktionell bearbeitet und im ELV Journal mit Nennung des Namens vorgestellt.

Leserwettbewerb



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

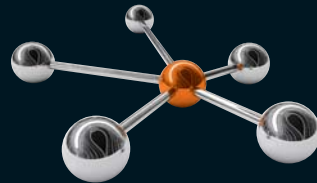
Warengutschein
€ 200,-

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 themenähnlichen 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 Einsenden liegen. Die eingesandten Unterlagen und Aufnahmen verbleiben bei der ELV Elektronik AG und können von dieser für Veröffentlichungen und zu Werbezwecken genutzt werden. Ihre Einsendungen senden Sie per Brief oder Mail mit Stichwort „Leserwettbewerb“ an:

ELV Elektronik AG, Leserwettbewerb, 26787 Leer bzw. leserwettbewerb@elvjournals.de

ELV-Technik-Netzwerk – jetzt kostenlos anmelden

Knüpfen Sie wertvolle Kontakte – schon über 58.000 Mitglieder* im ELV Technik-Netzwerk. Profitieren Sie von der Kompetenz unserer Kunden und Mitarbeiter! Werden Sie Mitglied und diskutieren Sie interessante Themen und Produkte.



Kontakte knüpfen!

Notruf-Funktion

*Stand: 24.04.2017

- Über 49.900 Beiträge* zu über 12.120* Themen
- Notruf-Funktion: Ihr Technik-Notruf ist 24 Stunden an prominentester Stelle sichtbar – so steigt die Wahrscheinlichkeit für schnelle Hilfe
- Mitglieder- und Expertensuche: Finden Sie Mitglieder und Experten mit gleichen Interessen oder dem Fachwissen, das Sie suchen



www.netzwerk.elv.de

Aktuelle News und Infos für noch mehr Wissen:



www.facebook.com/elvelektronik



plus.google.com/+ElvDeShop



www.twitter.com/elvelektronik

Service

Technische Anfragen

Für spezielle technische Fragen nutzen Sie bitte unseren Technischen Kundendienst, der Ihnen gerne umfassende und qualifizierte Auskünfte erteilt. Damit es schneller geht: Bitte nennen Sie uns ggf. Bestellnummer, Artikelbezeichnung und Katalogseite. Danke! Die Kontaktdaten finden Sie in der Tabelle unten.

Reparatur-Service

Für ELV Markenprodukte, aber auch für Geräte, die Sie aus ELV Bausätzen selbst herstellen, bieten wir Ihnen einen kostengünstigen Reparatur-Service an. Im Sinne einer schnellen Abwicklung führen wir eine Reparatur sofort durch, wenn die Reparaturkosten den halben Artikelpreis nicht überschreiten. Bei einem größeren Defekt erhalten Sie vorab einen unverbindlichen Kostenvorschlag. Die Kontaktdaten:

- ELV / Reparatur-Service / 26789 Leer
- ELV / Reparatur-Service / Pakettfach ELV 1 / 5005 Salzburg
- ELV / Reparatur-Service / Postfach 100 / 4313 Möhlin

Qualität/Sicherheit

Komplettbausätze von ELV beinhalten sämtliche zum Aufbau erforderlichen elektronischen und mechanischen Teile einschließlich Platinen, Gehäuse mit gebohrter und bedruckter Frontplatte, Netztrafos, Schrauben, Muttern usw. Es finden ausschließlich hochwertige Markenbauteile Verwendung. Fertigergeräte werden mit Gehäuse betriebsfertig und komplett abgeglichen geliefert. Sämtliche ELV Bausätze und ELV Fertigergeräte sind mit 1%-Metallfilmwiderständen ausgerüstet. Technische Änderungen vorbehalten.

Hinweis

Bitte beachten Sie beim Aufbau von Bausätzen die Sicherheits- und VDE-Bestimmungen. Netzspannungen und Spannungen ab 42 V sind lebensgefährlich. Bitte lassen Sie unbedingt die nötige Vorsicht walten und achten Sie sorgfältig darauf, dass Spannung führende Teile absolut berührungssicher sind. Zahlreiche ELV Bausätze, insbesondere solche, bei denen für den Betrieb der fertigen Geräte Netzspannung erforderlich ist, dürfen ausschließlich von Profis aufgegeben werden, die aufgrund ihrer Ausbildung dazu befugt und hinreichend mit den einschlägigen Sicherheits- und VDE-Bestimmungen vertraut sind.

Bitte nennen Sie uns bei Bestellungen:

	▶ Kundenummer	▶ Bestell-Nummer	▶ Zahlungswunsch		
	Deutschland		Österreich		Schweiz

Bestellen

	Deutschland	Österreich	Schweiz
Telefon (Montag bis Freitag 8.00–19.00 Uhr)	0491/6008-88	0662/624-084	061/9711-344
Fax	0491/7016	0662/624-157	061/9711-341
Internet	www.elv.de	www.elv.at	www.elv.ch
E-Mail	bestellung@elv.de	bestellung@elv.at	bestellung@elv.ch
Versandkosten	€ 5,95	€ 5,95	CHF 9,95
Versandkostenfrei*	ab € 150,-	ab € 150,-	ab CHF 150,-

Technische Beratung

	Deutschland	Österreich	Schweiz
Telefon (Montag bis Freitag 9.00–19.00 Uhr)	0491/6008-245	0662/627-310	061/8310-100
Fax	0491/6008-457	0662/624-157	061/9711-341
E-Mail	technik@elv.de	technik@elv.at	technik@elv.ch

Kundenservice

	Deutschland	Österreich	Schweiz
Telefon (Montag bis Freitag 8.00–19.00 Uhr)	0491/6008-455	0662/624-084	061/9711-344
Fax	0491/6008-459	0662/624-157	061/9711-341

Kontostand

	Deutschland	Österreich	Schweiz
E-Mail	konto@elv.de	konto@elv.at	konto@elv.ch
Fax	0491/6008-316	0662/624-157	061/9711-341



Fledermausdetektor

Der Fledermausdetektor setzt die Ortungssignale von Fledermäusen in den hörbaren Bereich um und bietet damit vielfältige interessante Hörerlebnisse. Nicht nur Fledermäuse und andere Tiere senden Ultraschallsignale aus, auch Gegenstände des täglichen Lebens können Ultraschall erzeugen. Ein klassisches Beispiel ist die Ultraschall-Fernbedienung, die bis in die 1989er-Jahre Verwendung zur Bedienung von Fernsehgeräten fand. Mit dem FMD1 können auch Ultraschall-Tiervertreiber getestet und durch technische Geräte (z. B. Schaltnetzteile) erzeugte Ultraschalltöne hörbar gemacht werden. Als Mikrofon kommt hier ein modernes MEMS-Mikrofon zum Einsatz, das sehr rauscharm ist und in einem weiten, linearen Frequenzbereich funktioniert.



MEMS-Mikrofon

MEMS-Mikrofone vereinen sehr kleine mechanische Strukturen mit elektronischen Komponenten. Das MEMS-1 deckt einen sehr weiten Frequenzbereich bis hinein in den Ultraschallbereich ab und ist als fertig einsetzbarer Baustein ausgeführt.



E-Paper-Display-Modul

Electronic Paper Displays sind für viele Anwendungen, insbesondere wenn es auf geringen Stromverbrauch sowie kontrastreiche Anzeige und die damit verbundene gute Ablesbarkeit ankommt, genau die richtige Lösung. Das neue E-Paper-Display-Modul stellt sowohl ein Lernobjekt für die Programmierung dieser seriell angesteuerten Displays als auch einen Ansatz für eigene, stromsparende Anzeigeapplikationen dar.



Homematic IP Temperatur- und Luftfeuchtigkeitssensor mit Display

Dieser kompakte, batteriebetriebene Sensor erfasst Temperatur- und Feuchtigkeitswerte in Räumen. Mithilfe der Homematic IP App bzw. im Zusammenspiel mit der Homematic Zentrale CCU2 lässt sich das Gerät als Raumthermostat einsetzen. Der Sensor selbst besitzt keine Bedienelemente, wodurch eine missbräuchliche Beeinflussung verhindert werden kann.



HomeMatic Funk-Controller für Dual-White-LEDs

Mit dem Controller können Dual-White-LEDs komfortabel per Homematic in der Lichtfarbe und Helligkeit gesteuert werden. Der PWM-Dimmer bietet zwei Open-Drain-Ausgangskanäle, die entweder als zwei separate, normale PWM-Dimmerkanäle oder als kombinierter Ausgang für einen Dual-White-Stripe verwendet werden können. Der Controller kann 12- und 24-V-Stripes steuern. Die Spannungsversorgung erfolgt üblicherweise über das vorhandene LED-Netzteil.

Mikrocontroller-Schnittstellen, Teil 5

Im fünften Teil werden die verschiedenen Aspekte der USB-Schnittstelle erläutert: Grundlagen und Einbindung in eigene Mikrocontroller-Applikationen.

Homematic Know-how, Teil 22

Das LED-Special – Dimmen mit Homematic. Wir zeigen, welcher Homematic Dimmer für die unterschiedlichen LEDs einzusetzen ist.

Raspberry Pi, Teil 5

Die Raspberry-Pi-Kamera im Robotik-Einsatz – eine Eigenbau-Pan-/Tilt-Kamera mit Zusatzfunktionen.

Arduino verstehen und anwenden, Teil 23

Nach dem Einstieg in den Umgang mit dem I²C-Bus steht der Einsatz von Portexpandern im Mittelpunkt, die eine einfache Porterweiterung ermöglichen.

Homematic Scriptprogrammierung, Teil 5

In diesem Teil geht es hauptsächlich um die Kommunikation mit Geräten mit API-Unterstützung.

Homematic Usertreffen – Neues und Interessantes

Ein Bericht von den äußerst interessanten Homematic Usertreffen in Kassel und Österreich.

Sat-Finder DUR-line SF4000 BT

Satellitenantenne einstellen leicht gemacht – der DUR-line-Sat-Finder ermöglicht das Ausrichten quasi auf Knopfdruck und mithilfe einer App.

FH Emden fördert weiblichen Fachkräftenachwuchs

Frauen sind in technischen Berufen noch immer in der Unterzahl. Die Hochschule Emden/Leer unterstützt den weiblichen Fachkräftenachwuchs, indem sie nicht nur Studierende fördert, sondern auch Projekte generiert, die Frauen und Mädchen bereits in jungen Jahren Lust auf „MINT“-Disziplinen machen.

Kreis-LED-Wecker K LW1

- ▶ Vielseitige Weck- und Schaltuhr
- ▶ Anzeige dimmbar
- ▶ Homematic Anbindung möglich

Mehr als nur der selbst gebaute Hingucker mit kombinierter Digital- und Kreisanzeige – der K LW1 verfügt über 2 Weckzeiten, eine genaue und stromausfallsichere Echtzeituhr, Funkuhr-Option und die Möglichkeit, die Uhr sogar als Homematic Schaltuhr einsetzen zu können.

Der Kreis-LED-Wecker K LW1 ist ein Wecker-Bausatz, der mittels einer in der Mitte angeordneten Reihe von 5 LED-Matrix-Modulen und eines rundum geführten LED-Kreises die Uhrzeit gut und auch weithin ablesbar anzeigen kann. 2 einzeln aktivierbare Weckzeiten, eine akustisch-optische Wecksignalisierung, eine dimmbare und sich der Raumhelligkeit automatisch anpassende Anzeige und die Optionen als Funk-Schaltuhr machen die Uhr enorm vielseitig.

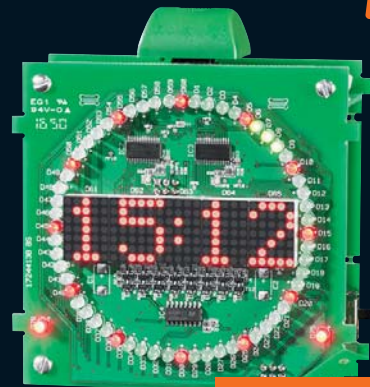
- Anzeigekombination aus mehrfarbigem LED-Kreis und LED-Matrixanzeige
- Aufrüstoption für DCF77-Funkempfänger. Zusätzlich zum DCF-Empfangsmodul CN-09 16 10 ist die externe DCF-Antenne CN-14 28 83 erforderlich.
- 8-Kanal-Sendemodul HM-MOD-EM-8, für Einsatz als Schaltuhr

USB-Netzteil, Eco-friendly, 5,0 W/1 A

CN-09 61 24 € 7,50

Bausatz Kunststoffgehäuse für Kreis-LED-Wecker K LW1

CN-15 13 95 € 9,95



ELV

NEU

MONTAGE
VIDEO

Komplettbausatz
CN-15 02 61

€ 49,95

Abm. (B x H x T):
106 x 105 x 55 mm
(ohne Gehäuse),
106 x 110 x 58 mm
(mit Gehäuse)

DCF-LED-Uhr DCF-7SEG

- ▶ Farblich individualisierbar
- ▶ Integrierte Helligkeitsregelung
- ▶ Mit Zeit- und Datumsanzeige

Ein Elektronik-Klassiker im neuen Outfit – diese LED-Uhr mit weißen 7-Segment-Anzeigen kann mit Farbfolien zu einer Uhr mit beliebiger Anzeigefarbe konfiguriert werden und wird damit zum Hingucker in jedem Raum. In der Grundausstattung eine hochgenaue Quarzuhr mit RTC-Steuerung kann die Uhr zur DCF-Uhr aufgerüstet werden.

weit brillanteres und „farbigeres“ Anzeigebild mit einer völlig homogenen Farb- und Helligkeitsverteilung.

Abm. (B x H x T): 160 x 120 x 35 mm

Der Clou an dieser Uhr sind die weißen 7-Segment-Anzeigen mit einer Ziffernhöhe von 20/15 mm für ein gleichmäßiges Leuchtbild. Schon in Weiß ergibt sich eine eindrucksvolle Anzeige, mit den optionalen Farbfolien kann man nahezu beliebige Farbeffekte erzielen, und es entsteht ein gegenüber farbigen LED-Anzeigen

Gleich mitbestellen:

Farbfilterfolien-Set, 6 Farben

CN-09 86 74 € 3,95

USB-Netzteil inkl. USB-Leitung, 1,5 m

CN-09 61 24 € 7,50



Anzeigefarben änderbar

DCF-LED-Uhr DCF-7SEG

Komplettbausatz (ohne DCF-Modul)

CN-09 83 69 € 64,95

LED-Uhr KLU2001

- ▶ Zeitanzeige mit mehr als 130 LEDs
- ▶ Kombination aus analoger und digitaler Anzeige
- ▶ Mit Zeit- und Datumsanzeige

Diese moderne Designuhr setzt die Tradition der klassisch-kultigen ELV-LED-Uhren fort.

Die Zeitanzeige von Stunde, Minute und Sekunde erfolgt mit 2 LED-Kreisen mit mehr als 130 roten Leuchtdioden. Im Zentrum wird gleichzeitig das Datum digital angezeigt. Im Grundausbau wird die Uhr quartzgesteuert, jedoch ist ein optionales DCF77-Modul für den Empfang des Zeitzeichensenders DCF77 einfach nachrüstbar.

ne zeigerähnliche Darstellung erzielt, und insbesondere die Stundenanzeige erfolgt sehr genau und fließend wie bei einer normalen Zeigeruhr. Das „Zifferblatt“ wird durch ständig leuchtende LEDs auf dem äußersten Kreis gebildet, die im 5-min-Raster angeordnet sind und so die vertraute Orientierung auf einem analogen Zifferblatt erleichtern.

Damit wird die Uhr zur hochgenauen Funkuhr. Durch die besondere Anordnung der Leuchtdioden wird ei-

Eine halb transparente, rote Abdeckscheibe erhöht den Kontrast der LEDs und gibt den Blick auf die prozessorgesteuerte Technik der Uhr frei. Wahlweise als Wand- oder Tischuhr betreibbar.



Abm. (B x H x T):
194 x 194 x 35 mm

Komplettbausatz LED-Uhr KLU2001

CN-04 67 26 € 59,95

Steckernetzteil Eco-Friendly 12 V

CN-09 51 05 € 8,95

Gleich mitbestellen: DCF-Empfangsmodul DCF-2

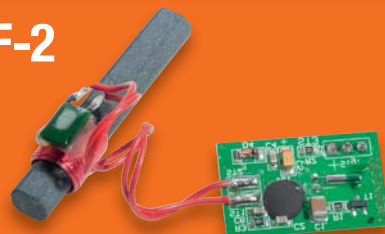
Hochwertiger DCF-Empfänger inkl. DCF-Antenne, ausgeführt in Miniaturlösung und mit innovativer Technik.

Das speziell für den Empfang des DCF-2-Senders im Langwellenbereich bei 77,5 kHz konzipierte Empfangsmodul

liefert an seinem Open-Collector-Ausgang bereits das demodulierte DCF-Signal.

- Ermöglicht die Umrüstung z. B. oben stehender Uhren zu Funkuhren. So entfällt auch das Anpassen auf Sommer-/Winterzeit

DCF-Empfangsmodul DCF-2
CN-09 16 10 € 9,95



Statt flackernder Leuchtstoffröhre – LED-Feuchtraum-Wannenleuchten, IP65

ELV

- ▶ Integrierte LED-Technik, dadurch besonders helles, gerichtetes Licht
- ▶ Sofort helles Licht auch bei niedrigen Umgebungstemperaturen
- ▶ Einfache Montage, Durchverdrahtung möglich



VIDEO
ONLINE



Die LED-Feuchtraum-Wannenleuchten stehen in puncto Lichtstärke und Lichtqualität Leuchten mit Leuchtstofflampenbestückung in nichts mehr nach – im Gegenteil, sie haben zahlreiche Vorteile.

Sie sind sparsamer, erheblich langlebiger, starten bei jedem Klima sofort und flackerfrei und arbeiten flimmerfrei.

Damit sind sie der energiesparende Ersatz für die herkömmliche Leuchtstoffröhre, sowohl bei Neuinstallation als auch beim Ersatz.

- Für den Einbau in Innen-, Außen- und Feuchtraumbereichen (staub- und wasserdicht, IP65)
- Neutralweißes Arbeitslicht mit 4000 K
- Hohe Betriebslebensdauer: bis zu 30.000 h Betrieb
- Homogenes und blendfreies Licht

Preis-Hit
ab

€ 17,95

Lieferung inkl. Montagematerial



Großer Anschlussraum, Durchverdrahtung möglich – Kabelverschraubung bereits enthalten



Einfache Montage mit mitgelieferten Halteclips, Schrauben und Dübeln

Kundenbewertung

Best.-Nr. CN-12 40 05



kaifischer85****:

„Leuchte erfüllt alle Erwartungen. Wurde als Ersatz für eine alte Leuchtstoffröhre bestellt. Endlich nicht mehr 30 s warten, bis die Lampe gestartet ist. Die Leuchte ist leicht anzuschließen. (...)“

Länge	Leistungs- aufnahme	Vgl. Leistung (Leuchtstofflampe)	Lichtstrom	Abm. (B x H x T)	Gewichteter Eigenverbrauch	Best.-Nr.	Preis
590 mm	18 W	bis 36 W	1500 lm	590 x 86 x 66 mm	18 kWh/1000 h	CN-12 40 05	€ 17,95
1180 mm	24 W	bis 58 W	2000 lm	1180 x 86 x 66 mm	24 kWh/1000 h	CN-12 40 06	€ 24,95
1180 mm	36 W	bis 58 W*	3000 lm	1180 x 86 x 66 mm	36 kWh/1000 h	CN-12 40 07	€ 34,95

*Für den Ersatz von 58-W-Leuchtstofflampen empfehlen wir die Best.-Nr. CN-12 40 06. Mit Best.-Nr. CN-12 40 07 ist der Lichteindruck heller als bei einer 58-W-Leuchtstofflampe. Diese Leuchten enthalten fest integrierte LEDs. Energieeffizienzklasse A++ bis A