

Technische Dokumentation

für die Elektronikbaugruppe

Universelle LED-Steuerung ULS

Version 1.0

Dateiname: BA20-ULS.ZIP Ausgabe 01.03.2018

Inhaltsverzeichnis

	Seite
1 Beschreibung der Baugruppe -----	3
1.1 Allgemeines -----	3
1.2 Schaltungsbeschreibung -----	5
1.3 Aufbau und Inbetriebnahme -----	6
2 Software und Programmierung -----	7
3 Ausführung -----	9
3.1 Bauvorschläge für die LED-Kappen -----	9
3.2 Beschreibung der Anschlüsse -----	10
4 Allgemeines -----	12
4.1 Literaturverzeichnis -----	12
4.2 Dateienverzeichnis -----	13
4.3 Elektronische Bauelemente -----	13
4.4 Hinweise zu den Nutzungsrechten -----	13
4.5 Technische Hinweise -----	15
4.5.1 Ausdrucken der Dateien -----	15
4.5.2 Gehäuse und Frontplattenbeschriftungen -----	15
4.5.3 Bauelementeauswahl -----	15
4.5.4 SMD-Bestückung -----	15

1 Beschreibung der Baugruppe

1.1 Allgemeines

Der Eine oder Andere kann sich sicherlich noch an die Zeit erinnern, als die ersten LED verfügbar waren. Seitdem haben sich diese Bauelemente ein breites Anwendungsfeld erschlossen, nicht zuletzt wegen der Entwicklung von weiß- und blaustrahlenden Typen. Mittlerweile sind LEDs in einer großen Vielfalt verfügbar und damit an vielen Stellen einsetzbar.

Für einfache Anwendungen genügt meist ein Vorwiderstand, dessen Wert sich nach den Angaben des jeweiligen Datenblattes der LED richtet. Interessanter sind Ansteuerungen, mit denen es möglich ist, das gesamte Farb- und Helligkeitsspektrum abzudecken. Beschäftigt man sich intensiver mit dieser Problematik, stellt man schnell fest, dass es vorteilhaft wäre, mehrere LEDs einzusetzen. Dies bedeutet jedoch, dass jede einzelne LED einen eigenen Steuerkanal benötigt, bei den RGB-LEDs sogar drei.

Wenn man die LEDs voneinander völlig unabhängig steuern möchte, kommt weder eine Matrixsteuerung noch eine Lösung mit Schieberegistern in die nähere Auswahl.

Die Trägheit des menschlichen Auges macht es möglich, LEDs auf der Basis der Pulsweitenmodulation (PWM) anzusteuern. Für das Auge entsteht so der Eindruck einer mittleren Helligkeit, die durch den LED-Strom und der Einschaltzeit innerhalb einer Periodendauer bestimmt wird.

Es wurde daher bei der Entwicklung der universellen Ansteuerbaugruppe darauf Wert gelegt, mit einer geringen Anzahl von Bauelementen auszukommen und gleichwohl etliche Einsatzgebiete abdecken zu können.

Ansteuermodule für LEDs gibt es in den unterschiedlichsten Ausführungen. Oft hat man jedoch keine Möglichkeit, eigene Ideen umzusetzen, da es aufwendig ist, Steuerausgänge in ausreichender Anzahl bereitzustellen.

Mit dem in diesem Artikel vorgestellte Modul soll eine Schaltung vorgestellt werden, mit der sich viele Wünsche kostengünstig und mit einem relativ geringen Aufwand umzusetzen lassen.

Für Projekte mit vielen LEDs wird neben anderen der Aufbau mit Schieberegistern favorisiert. Die Übertragungszeit des Anzeigemusters nimmt mit steigender LED-Anzahl zu und die Helligkeitssteuerung kann mit einem vertretbaren Aufwand maximal für einzelne Gruppen realisiert werden. Das ist leider hinderlich, wenn man das volle Farbspektrum moderner RGB-LEDs anwenden möchte.

Hier sollen kurz die Vorteile der Baugruppe ULS aufgezeigt werden:

- 4 Ports (optional) zu je 8 Kanälen frei programmierbar
- kontinuierliche Farb- und Helligkeitssteuerung
- Ausgleichsmöglichkeit bei Unterschieden der LED-Kennlinien, insbesondere im Bereich kleiner Ströme
- Anpassung an unterschiedliche Betriebsspannungen

Der Kreativität sind kaum Grenzen gesetzt, da die Implementierung eines Bussystems die Übermittlung der aktuellen Betriebsart an ein identisch aufgebautes Slave-Modul übertragen werden kann. Dabei wurde auf spezielle I²C-Bausteine verzichtet. Dies hat den Vorteil, dass im Slave-Modul dieselben Muster für die dort angeschlossenen LEDs erzeugt werden können oder die Anzeigen unabhängig von der Master-Baugruppe generiert werden.

Die Busstruktur würde es auch zulassen, komplette Bitmuster an die Slave-Bausteine zu übertragen. Dies kann man machen, es muss jedoch an die erforderlichen Übertragungszeiten gedacht werden. Eventuell führt es zu Störungen beim PWM-Signal, das interruptgesteuert erzeugt wird. Verwendet man Pausen in der Anzeige für die Datenübertragung, ist aber dies kein Problem.

Generierung und Anzeige der Bitmuster sind so vielfältig, dass es nicht möglich ist, alles in einem Beispielprogramm unterzubringen. Es wird noch umfangreicher, wenn man Zweifarb- oder RGB-LEDs einsetzt.

Da Mikrocontroller mittlerweile preiswert erhältlich sind, entstand eine darauf basierende Schaltung. Da man über eine Softwareanpassung in der Lage ist, jederzeit die Funktionalität anzupassen oder zu erweitern, ist dies eine bessere Wahl, als ein „Schaltkreisgrab“ aufzubauen. Der ATmega16 der Firma Microchip (früher Atmel) verfügt über 4 Ports mit jeweils 8 frei konfigurierbaren Kanälen. Damit sind 32 Kanäle verfügbar (z. B. für 32 weiße bzw. 10 RGB-LEDs). Die Timeranzahl ist leider zu gering, um alle LEDs mit den intern verfügbaren PWM-Signalen anzusteuern. Daher wurde innerhalb der Software (Interruptsteuerung) eine Routine realisiert, in der jeder Kanal gezielt angesprochen werden kann.

Die eingesetzten Treiberschaltkreise ermöglichen einen LED-Strom von bis zu 500 mA bei einer maximalen Betriebsspannung von 50 V. Damit ist ferner eine Reihen- und/oder Parallelschaltung von LEDs problemlos möglich.

Dies ist eine Möglichkeit für den Aufbau eines Steuermoduls mit der entwickelten Leiterplatte. Sie hat den Nachteil, dass weitestgehend auf Bedienelemente verzichtet werden muss, um zum Beispiel manuell auf Farbe oder Helligkeit Einfluss zu nehmen.

Bei der Konzeptentwicklung wurde diese Problematik nicht übersehen und folgende Lösung favorisiert. Mit der Integration eines Bussystems (TWI) ist jederzeit eine Erweiterung möglich.

Im Schaltplan fallen möglicherweise die Widerstände R5/R6 sofort auf. Sie sind Bestandteil der Implementierung des Bussystems. An dieser Stelle soll nicht näher auf das Funktionsprinzip eingegangen werden, da umfangreiche Informationen in [5] und [6] verfügbar sind.

Das Aufbaukonzept der Baugruppe ermöglicht den Einsatz der Leiterplatte als Master- oder Slave-Baugruppe (MSBG) in einem Bussystem. Die diesem Artikel beiliegenden Beispielprogramme für die MSBG reißen nur kurz die Möglichkeiten an, die sich mit diesem Steuerungskonzept realisieren lassen. Weiterhin steht eine einsatzfähige Firmware für eine LED-Kette zur Verfügung, die gerade in der Vorweihnachtszeit vielfältige Lichteffekte ermöglicht. Beim Prototyp wurden je Kanal 4 LED parallel angeschlossen, um eine größere Anzahl an Leuchtpunkten zu erhalten.

Soll eine MSBG aufgebaut werden, kann auf die Bestückung des Treiberschaltkreises (IC40) mit den strombegrenzenden Widerständen verzichtet werden. Der Port C des MCs ist damit frei für die Bedien- und Anzeigeelemente sowie den erforderlichen zwei Kanälen für die Buskopplung. Auf der Slave-Baugruppe kann über zwei dieser Kanäle per Jumper eine Adressierung vorgenommen werden, wenn nicht für jede Slave-Baugruppe ein separates Programm erstellt werden soll, bei dem die Adressierung in der Software erfolgt.

Auf der Masterbaugruppe stehen für die Ansteuerung der LED jetzt zwar nur noch 24 Kanäle zur Verfügung (24 weiße bzw. 8 RGB-LEDs ansteuerbar), aber die Information der eingestellten Betriebsart kann über den realisierten TWI-Bus auf die Slave-Baugruppe übertragen werden.

Da jede Baugruppe im System (Master oder Slave) separat das PWM-Signal für die angeschlossenen LEDs generiert, wird die Rechenkraft jedes einzelnen MCs optimal genutzt und man ist sehr flexibel bei der Programmierung der zu erzielenden Effekte. Daher sind zum Beispiel Lauflichter nicht auf die eingangs erwähnten 32 LEDs begrenzt und der Aufbau farbiger Laufbänder möglich.

Der Aufbau des Bussystems ermöglicht ferner die Einbindung einer LCD-Anzeige. Die vor Kurzem im Funkamateur vorgestellte Anpassung für ein LC-Display [13] soll hier als Beispiel dienen.

Auf der Leiterplatte sind Bohrungen für Abstandsbolzen vorhanden, die einen kompakten Aufbau von Master- und Slave-Baugruppen ermöglicht.

Sollte es beim Aufbau von Bussystemen, bei denen Master- und Slave-Baugruppen weiter voneinander getrennt betrieben werden sollen, zu Fehlern bei der Datenübertragung kommen, gibt es mehrere Möglichkeiten, diese zu beseitigen. Neben dem Einsatz spezieller I²C-Bustreibern oder der Reduzierung des Bustaktes findet man im Internet viele Hinweise.

An dieser Stelle sollen noch einige Hinweise zu weiteren Eigenschaften der LEDs gegeben werden, die man im ersten Augenblick eventuell übersieht, aber dadurch später der gewünschte Lichteffekt nicht erreicht werden kann. Gemeint sind die Helligkeit, der Abstrahlwinkel und die Farberzeugung.

Bei einem kleinen Abstrahlwinkel bekommt man mehr oder weniger punktuelle Effekte, die bei RGB-LEDs eventuell zu übereinanderliegenden Farbringen führt. Bei anderen Projekten ist möglicherweise eine diffuse Abstrahlung vorzuziehen. Außerdem sollte das Datenblatt auf Hinweise geprüft werden, die zuweilen bei RGB-LEDs angeben, dass produktionsbedingt nicht das komplette Farbspektrum erzielt werden kann, was bei LEDs mit 10 mm Durchmesser auftritt.

Weiterhin soll auf eine Besonderheit bei Zweifarb-LEDs hingewiesen werden, die nur über zwei Anschlüsse verfügen. Mit diesen können durch die ULS je nach Zuordnung der Anode und Katode nur die eine oder andere Farbe erzeugt werden.

Anwendungsvorschläge

- Weihnachtsbeleuchtung für Baum und Dekoration
- Farbvariationen für Beleuchtungszwecke, z. B. Hintergrundbeleuchtung im Kunstbereich, Kugel- und Faserlampen mit Farbwechsel, beleuchtete Silhouetten, Flackerlicht
- Anzeigen für Werbung und Beleuchtungseffekte durch LED-Cluster (z. B. Glassteinbeleuchtung)
- PC-Modding, Dimmer und LED-Würfelgitter (LED-Cube)

- Mehrfarb-LED als Temperaturanzeige mit Farbübergang
- Windrose (Anzeige der Stellung von Antennenrotoren)
- Erweiterung zu einem System, welches von einem PC gesteuert werden kann
- mit Zusatzbaugruppe AC 230 V Verbraucher steuerbar

1.2 Schaltungsbeschreibung

Die prinzipielle Funktion der Steuerung ist im Blockschaltbild dargestellt.

Die Baugruppe ist dafür ausgelegt, mit zwei Betriebsspannungen versorgt werden zu können. Eine stabilisierte Spannung (DC +5 V) versorgt dabei den MC und ist in jedem Fall erforderlich. Die Anschlussklemme X10 ermöglicht außerdem den separaten Anschluss einer zweiten Spannungsquelle. Per Jumper (JMP3) auf dem Steckverbinder X5 kann gewählt werden, mit welcher dieser beiden Spannungen die LEDs betrieben werden sollen. Als vorteilhaft hat sich ein kleines Schaltnetzteil erwiesen, das neben einer DC +5 V- auch eine DC +12 V-Spannung bereitstellt (stabilisiert).

Mit dem Quarz (Q1) wird die Taktfrequenz für den MC festgelegt. Durch die RC-Kombination (R1, C1) wird ein „Reset“ (Einschaltimpuls) erzeugt und über den Steckverbinder X4 ist der Anschluss eines externen Reset-Tasters möglich.

Die Ausgänge des MCs lassen bis zu 20 mA zu. Bei kleinen Projekten kann auf die Treiberschaltkreise, die an jeden Port angeschlossen sind, verzichtet werden.

Ansonsten ist der Einsatz der Treiberschaltkreise zu empfehlen. Sie reduzieren den Leistungsumsatz im MC und der hochohmige Anschluss an den Kanälen, die für die Programmierung erforderlich sind, ermöglicht ein Softwareupdate für den MC ohne Entfernung des MCs aus der Schaltung. Außerdem ist man durch die Treiber sehr flexibel, was die Betriebsspannung für die LED betrifft (laut Datenblatt bis zu DC +50 V).

Die LEDs - über die Ports A, B und D angesteuert - werden über den Steckverbinder X30 angeschlossen; Port C ggf. über X31.

Die Pins der beiden Steckverbinder X20 und X21 sind parallel geschaltet. Bei der Master-Baugruppe ermöglicht dies den Anschluss weiterer Slave-Baugruppen und den gleichzeitigen Betrieb eines Programmiergerätes. Welcher MC dieses Verbundes programmiert werden soll, entscheidet der Jumper (JMP2) auf dem entsprechenden Steckverbinder X3. Als problemlos einsetzbar erwies sich wieder einmal mehr der USBasp-Brenner, auf den bereits näher in einer der letzten Ausgaben des Funkamateurs eingegangen wurde. Natürlich kann auch extern mit jedem anderen Programmiergerät, was für ATMEL-MCs geeignet ist, programmiert werden.

Zu achten ist auf die korrekte Einstellung der Fuse- und Lookbits.

Stellt das Programmiergerät eine Versorgungsspannung für die angeschlossene Baugruppe zur Verfügung, kann durch eine Lötbrücke (BR2) auf der Bestückungsseite der Leiterplatte dauerhaft eine Verbindung realisiert werden.

Sollte es zu Problemen beim Programmieren kommen, könnten die an den entsprechenden Pins parallel angeschlossenen LEDs die Ursache sein, wenn man auf die Treiberschaltkreise verzichtet hat; ggf. müssen diese vorher abgetrennt werden.

Sind im speziellen Anwendungsfall die maximal 500 mA, mit denen die Kanäle der Treiberschaltkreise betrieben werden können, zu gering, der Gesamtstrom für die Leiterplatte zu hoch oder sollen AC-Verbraucher angeschlossen werden, muss eine separate Treiberbaugruppe eingesetzt werden. Ein Beispiel ist eine Variante mit Transistoren, die es in der Entwicklungsphase zuließ, diverse LED-Typen zu testen.

Will man die Möglichkeiten der PWM nutzen, muss darauf geachtet werden, dass die zusätzliche Treiberbaugruppe mit der gewählten PWM-Frequenz betrieben werden können.

In den Programmbeispielen werden die LEDs mit einer Frequenz von ca. 65 Hz angesteuert, um einen weitestgehend flackerfreien Farb- bzw. Helligkeitseindruck zu erzielen.

Jede Baugruppe ermöglicht den Einsatz von maximal drei Tasten, mit denen in das Programm eingegriffen werden kann. Diese können, wenn Mikrotaster verwendet werden, direkt auf der Leiterplatte angebracht sein. Außerdem sind diese Anschlüsse über den Steckverbinder X1 nach außen geführt.

Im Beispielprogramm wurde eine multifunktionale Auswertung der Tasten realisiert. Das bedeutet, dass das Programm erkennt, ob eine Taste kurz oder etwas länger betätigt wurde. Damit sind bei einer Eintastbedienung bis zu 6 verschiedene Funktionen möglich. Im Beispielprogramm wird bei einer kurzen Betätigung von

Taste 1 die nächste Betriebsart gewählt, mit einer längeren Betätigung stellt sich die erste im Programm festgelegte Betriebsart ein.

1.3 Aufbau und Inbetriebnahme

Um die Nachbausicherheit zu erhöhen und Interessenten an dieser Technik eine fehlerfreie Baugruppe zu präsentieren, wurden zunächst die beiden Module (Master und Slave) parallel auf einem Entwicklungsbord aufgebaut. Schaltungstechnisch relativ anspruchslos erfordert es nur eine einfach gehaltene, für beide Baugruppen einsetzbare Leiterplatte (Zwei-Ebenen-Platine).

Da diese mittlerweile in einer durchkontaktierten Version relativ preiswert erworben werden können, wurde sich für diese Ausführung entschieden.

Widerstände zur Strombegrenzung

LEDs sind Halbleiterbauelemente, die für den Betrieb einen Widerstand benötigen, dessen Wert nach bekannten Formeln berechnet wird oder mit einem Tool ermittelt werden kann (z. B. online auf [12]). Aus der Normreihe wird ein entsprechender Wert unter Beachtung der zu erwartenden Verlustleistung ausgesucht. Der Platz auf der Leiterplatte ist für Metallschichtwiderstände mit bis zu 600 mW Verlustleistung ausreichend. Ansonsten können die Widerstände an jeder Stelle der LED-Zuleitung eingebaut werden. Auf diese Art und Weise ist auch eine nachträgliche Erhöhung des Widerstandswertes möglich, wenn man eine Stromversorgung mit einer höheren Spannung einsetzen möchte, um z. B. mehrere LEDs in Reihe zu betreiben.

Sollte eine Verkleinerung der Widerstandswerte erforderlich sein, sind eventuell alle Widerstände auf der Leiterplatte kurz zu schließen und die passenden Bauelemente extern einzufügen.

Unter Umständen kann es bei längeren Leitungen trotz des Einsatzes geschirmter Leitungen zu Störungen anderer Verbraucher kommen. Weitere Entstörmaßnahmen wären der Einsatz von Ferritperlen oder/und der Einbau von Kondensatoren direkt an den LEDs.

2 Software und Programmierung

Das Herzstück der vorgestellten Schaltung ist der MC ATmega16, von dem die kompletten Datenblätter über die Homepage des Herstellers bezogen werden können [10].

Möchte man den Funktionsumfang erweitern und der Programmspeicher reicht nicht mehr aus, kann er problemlos durch den pinkompatiblen ATmega32 ausgetauscht werden.

Für die Erstellung der MC-Software ist Bascom - eine komplette Basic-Entwicklungsumgebung für die verschiedensten AVR Controller - zum Einsatz gekommen. Aber auch jede andere Software, mit der sich Code für diese Prozessoren erstellen lässt, ist einsetzbar.

Die Beispielprogramme sind voll funktionsfähig; sie beinhalten neben dem Quellcode für die LED-Kette auch die compilierten Hex- bzw. Binär-Dateien. Nach der Einstellung von Fuse- und Lookbits und der Programmierung wird die zyklische Programmabarbeitung durch ein „Lebensbit“ angezeigt, wenn die Zweifarb-LED (VD4) an Port C angeschlossen wurde.

Der Quelltext ist sehr umfangreich dokumentiert. Dennoch gibt es sicherlich, solange wie man sich einarbeitet, allerhand Fragen. Man sollte sich nicht scheuen, in diesem Fall eine kurze Anfrage an den Autor zu stellen.

Die Programmspeicher der MC sind mit zu je ca. 50% belegt. Das lässt noch eine große Anzahl zusätzlicher Routinen zu, wenn auf dem vorgeschlagenen Konzept aufgebaut werden soll. Kleinere Anwendungen sind schon mit weit weniger Pogrammcode realisierbar.

An dieser Stelle soll kurz auf die Programmstruktur eingegangen werden.

Nach den allgemeinen Einstellungen (Definitionsdatei für den Prozessortyp, Taktfrequenz, usw.) folgt der Deklarationsteil, in dem zunächst eine Bibliothek (nur bei der Masterbaugruppe) für den TWI-Bus eingebunden wird. Nach der Auflistung der Variablen folgt die Konfiguration für die Ports, der Timer 0 und 2 (jeweils 8 bit Timer) und den Routinen für die Interruptbehandlung. Um eine bessere Übersicht zu behalten, werden die einzelnen Kanäle der Ports mit „Alias“-Namen versehen.

Es folgt die Initialisierung, die einmal beim Start des MCs durchlaufen wird (z. B. Festlegung der Konstanten und Slave-Adressen).

Das Hauptprogramm wird zyklisch innerhalb einer Do-Loop-Schleife durchlaufen. Wird die Betriebsart geändert, erfolgt durch eine Subroutine die Informationsübertragung (TWI-Bus) an die Slave-Baugruppe. Danach wird das Datenfeld, in dem der Zeitpunkt für das Ausschalten jedes einzelnen LED-Kanals eingetragen ist, entsprechend dem zu erzielenden Lichteffect beschrieben. Dies kann je nach Art des Effektes per Zählschleife, Zufall oder durch das Einlesen eines Datenmusters, das am Programmende definiert ist, erfolgen.

Bei größeren Verschachtelungstiefen innerhalb von Schleifen ist deren maximale Anzahl vom eingesetzten Compiler abhängig.

Mit zusätzlichen Pausen ist die „Durchlaufgeschwindigkeit“ des jeweiligen Lichteffectes beeinflussbar.

Damit zeitnah auf eine Tastenbetätigung reagiert werden kann, sind die Wartezeiten mittels Subroutine programmiert, in der zyklisch eine eventuelle Eingabe abgefragt wird.

Etwas Umfangreicher wird die Programmierung, wenn statt weißer RGB-LEDs verwendet werden. In diesem Fall muss getestet werden, ob es programmtechnisch optimaler ist, die drei Farbanteile durch nebeneinander liegende Kanäle bereitzustellen, oder dies portweise zu organisieren.

Die Interruptroutine von Timer 0 wird alle 60 μ s aufgerufen. Diese kurze Zeit ist erforderlich, um mit der PWM-Steuerung bei 256 möglichen Zuständen der Kanäle einen gleichmäßigen Helligkeitseindruck flackerfrei zu erzeugen.

Zum Zeitpunkt $t=0$ werden alle Kanäle aktiviert, das heißt, es erfolgt das Einschalten aller LEDs. Danach wird zyklisch abgefragt, ob der im LED-Datenfeld für die Helligkeit hinterlegte Wert dem PWM-Zählwert entspricht. Ist diese Bedingung erfüllt, wird der jeweilige Kanal deaktiviert (die jeweilige LED wird ausgeschaltet).

Leider werden vom menschlichen Auge linear ansteigende Helligkeitswerte nicht als solche wahrgenommen. Daher ist es erforderlich, den angestrebten Helligkeitswert zu quadrieren, auf den Wertebereich von 0 bis 255 zu normieren und in das LED-Datenfeld einzutragen. Das Beispielprogramm realisiert auf diese Art und Weise ca. 60 mögliche Helligkeitswerte. Möchte man direkt auf einen Wert zugreifen, kann die Tabelle

aller Wertepaare, die sich am Anfang des Programmkomentars befindet, verwendet werden. Kommt man mit einer geringeren Anzahl aus, kann im Initialisierungsbereich des Programms der Wert von K_hst (Konstante Helligkeitsstufen) reduziert werden.

Mit Timer 2 wird die Steuerung der Anzeige „Lebensbit“ und die Wartezeit für einen automatischen Betriebsartenwechsel realisiert, um bei der Verwendung von Master- und Slave-Baugruppen weitestgehend eine synchrone Anzeige zu erhalten.

Hat man sich entschlossen, in allen Baugruppen dasselbe Bitmuster ablaufen zu lassen und wird zu jedem Zeitpunkt völlige Synchronität verlangt, sind in der Software entsprechende Verfahren einzufügen und zu optimieren (z. B. zusätzliche Wartezeiten, um Unterschiede bei den Zykluszeiten zwischen Master- und Slave-Baugruppen anzupassen). Im Beispielprogramm wird periodisch die Betriebsart neu initialisiert. Damit wird der Start für die Anzeigeroutine zum (fast) identischen Zeitpunkt ausgelöst.

Weitere Subroutinen ermöglichen die schnelle Einstellung einer Grundhelligkeit, das Auslesen von LED-Mustern aus dem Datenbereich, das Umrechnen des PWM-Wertes, die Reaktion auf eine Tasteneingabe und den Datentransfer über den TWI-Bus. Bei dem letzten genannten Programmteil soll erwähnt werden, dass bei entsprechender Konfiguration auch mehrere Datenbytes hintereinander an eine Slave-Baugruppe gesendet werden können. Dies wäre eine weitere Variante, um LED-Muster an die Slave-Baugruppen zu übertragen. Die Programme beider Baugruppen müsste für diese Funktionsweise anders konzipiert werden.

3 Ausführung

3.1 Bauvorschläge für die LED-Kappen

Leider war die bisherige Suche nach einer industriell gefertigten LED-Abdeckung, die neben einer ansprechenden Form auch in der Lage ist, das Licht diffus abzustrahlen, nicht befriedigend.

Eine Alternative sind industriell gefertigte Kerzen - dazu näheres am Ende dieses Abschnittes.

Über diverse Anbieter kann für ca. 0,60 € ein kleiner „Kerzentropfen“ (z. B. Conrad Electronic, Best.-Nr.: 181581-62) bezogen werden.

Von Anbietern, die den 3D-Druck anbieten, kamen ablehnende Antworten in Bezug auf die Materialeigenschaften oder der Mindestbestellwert liegt zu hoch, wobei dies die eleganteste Lösung gewesen wäre.

Vor diesem Hintergrund wurde recherchiert, welche Alternativen zum Einsatz kommen könnten. In die enge Auswahl kamen zwei Materialien. Das Bild zeigt die Ergebnisse, die nach etwas Bearbeitung zum gewünschten Ziel führten.

Bei dem ersten Material handelt es sich um Plexiglas (Muster 1). Nach dem Anschleifen gibt das Formelement ein diffuses Licht ab.

Ein besseres Ergebnis kann mit Heißkleber (milchig) erzielt werden. Er lässt sich thermisch gut bearbeiten und nimmt nach dem Erkalten seine ursprüngliche Färbung wieder an.

Eine Möglichkeit für ein Leuchtmittel mit einer 5 mm LED zeigen die Muster 3 bis 5.

Als Halbzeug dient ein, in einem Wasserbad auf ca. 40- 50 °C erwärmtes Stück Heißkleber mit einem Durchmesser von 11 mm. Durch die Erwärmung lässt sich der Heißkleber besser schneiden. Benötigt wird ein Stück von 30 mm Länge für den Schaft und eine 4 mm dicke Scheibe. Für weiße LED, die heller strahlen, kann der Schaft auch länger gewählt werden.

In die Scheibe wird ein Loch mit einem Durchmesser von 6 mm gestanzt. Wird eine Klammer o. ä. an die als LED-Führung dienende Aderendhülse (16 mm²) gelötet, ist der entstandene Ring für die Montage einseitig zu schlitzen. Nachdem die Seite mit dem konischen Kragen der Aderendhülse erwärmt wurde (Heißluftpistole), kann sie auf dem Schaft fixiert werden (nur leicht eindrücken!). Nach dem Erkalten wird der Ring aufgesteckt und mit einem ca. 10 mm langen Schrumpfschlauch etwa 1 mm überstehend überzogen (Schaftbedeckung 5 mm). Bei einer sich anschließenden vorsichtigen Erwärmung wird die Aderendhülse eingeklebt. Erfolgt die Erwärmung zu stark, wird Heißkleber durch die Schrumpfung herausgedrückt.

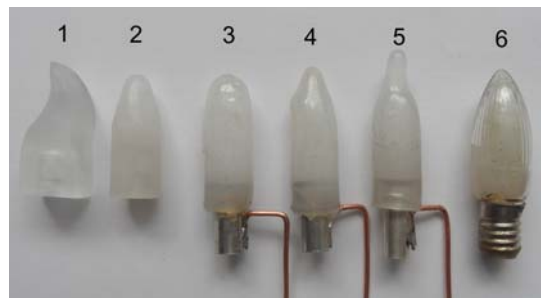
Nach dem Erkalten kann die andere Schaftseite behutsam erwärmt und eine Kerzenform moduliert werden. Nachdem die konfektionierte LED in die Aderendhülse eingeführt wurde, wird mit einem Schrumpfschlauch, der ferner als Zugentlastung dient, alles wasserdicht verschlossen.

Mit Heißkleber ist man relativ einfach in der Lage, kleine Formen, die in den unterschiedlichsten Farben erstrahlen sollen, preisgünstig herzustellen.

Wer es eleganter möchte, kann als weitere Möglichkeit eine Spitzen- bzw. Riffelkerze (mindestens Sockelgröße E10) behutsam „entkernen“ und sie danach mit Heißkleber auffüllen. Danach wird die LED mit ihrer Anschlussleitung eingeklebt und fertig ist das Lämpchen (Muster 6).

Für eine preiswerte Anschlussleitung kann flaches, 4-poliges Telefonkabel eingesetzt werden. Mit den passenden Modularsteckern und -buchsen ist damit ein platzsparendes Stecksystem aufbaubar. Für Einsätze im Außenbereich ist der Temperaturbereich zu beachten, ggf. ist auf rundes AWG-Kabel oder ähnliche Typen auszuweichen, die beständiger gegen UV-Licht sind.

Erst bei sehr großen Längen oder LEDs mit höherem Strombedarf als 20 mA sollte der Spannungsabfall auf der Leitung bei der Dimensionierung des jeweiligen Vorwiderstandes beachtet werden.





Eine andere Möglichkeit ist der Umbau von Kerzen, die im Zusammenhang mit kabellosen Weihnachtsbaumbeleuchtungen angeboten werden.

Bei ebay werden Sets als B-Ware und damit recht preiswert angeboten. Sie bieten im Raum für die Batterie die Möglichkeit, nicht nur die LED-Vorwiderstände unterzubringen, sondern auch Leiterplatten mit einer Ansteuerelektronik.

Neugierig geworden? In der Bauanleitung zu adressierbaren LEDs (LAD) gibt es mehr Informationen.

3.2 Beschreibung der Anschlüsse

Folgende Übersicht der Steckverbinder und der Klemme X10 erleichtert den Anschluss der Module:

Bedienung (parallel zu den Tastern auf der LP)

X1. 1	GND	Masse
X1. 2	T1	Taste 1
X1. 3	T2	Taste 2
X1. 4	T3	Taste 3

Adressierung (Jumper: Slave-Adressierung)

X2. 1	GND	Masse
X2. 2	ADR0	Adresse 0 (Taste 2)
X2. 3	GND	Masse
X2. 4	ADR1	Adresse 1 (Taste 3)

Reset bei der Programmierung (Jumper: MC-Auswahl)

X3. 1	PRG_/R	Signal vom Programmiergerät
X3. 2	/Reset	Reset-Eingang am MC

Reset durch Taster

X4. 1	GND	Masse
X4. 2	/Reset	Reset-Eingang am MC

Auswahl der LED-Betriebsspannung (Jumper: Wahl der Einspeisung)

X5. 1	ES1	Einspeisung 1 (+5V)
X5. 2	+Uext LED-Spannung	
X5. 3	ES2	Einspeisung 2 (+nV)
X5. 4	+Uext LED-Spannung	

Anzeige LED (zweifarbzig)

X6. 1	GND	Masse
X6. 2	Anode1	LED grün
X6. 3	Anode2	LED rot

Einspeisung

X10. 1	+5V	Anschluss 1
X10. 2	+Uext Anschluss 2	
X10. 3	GND	Masse

Programmier- und Busanschluss (für X21 gilt dieselbe Belegung)

X20. 1	MOSI	Programmier-Steuerleitung
X20. 2	+5V	DC Betriebsspannung (optional Brücke 2)
X20. 3	-	-
X20. 4	SCL	Busleitung (Takt)
X20. 5	/Reset	Programmier-Steuerleitung
X20. 6	SDA	Busleitung (Daten)
X20. 7	SCK	Programmier-Steuerleitung
X20. 8	GND	Masse
X20. 9	MISO	Programmier-Steuerleitung
X20.10	GND	Masse

Anschluss LED 1-24		(weiß)	(farbig)
X30. 1	L01_2 Katode	LED 1	bzw. LED 1 rot
X30. 2	L02_2 Katode	LED 2	bzw. LED 1 grün
X30. 3	L03_2 Katode	LED 3	bzw. LED 1 blau
X30. 4	L04_2 Katode	LED 4	bzw. LED 2 rot
X30. 5	L05_2 Katode	LED 5	bzw. LED 2 grün
X30. 6	L06_2 Katode	LED 6	bzw. LED 2 blau
X30. 7	L07_2 Katode	LED 7	bzw. LED 3 rot
X30. 8	L08_2 Katode	LED 8	bzw. LED 3 grün
X30. 9	L09_2 Katode	LED 9	bzw. LED 3 blau
X30.10	L10_2 Katode	LED 10	bzw. LED 4 rot
X30.11	L11_2 Katode	LED 11	bzw. LED 4 grün
X30.12	L12_2 Katode	LED 12	bzw. LED 4 blau
X30.13	L13_2 Katode	LED 13	bzw. LED 5 rot
X30.14	L14_2 Katode	LED 14	bzw. LED 5 grün
X30.15	L15_2 Katode	LED 15	bzw. LED 5 blau
X30.16	L16_2 Katode	LED 16	bzw. LED 6 rot
X30.17	L17_2 Katode	LED 17	bzw. LED 6 grün
X30.18	L18_2 Katode	LED 18	bzw. LED 6 blau
X30.19	L19_2 Katode	LED 19	bzw. LED 7 rot
X30.20	L20_2 Katode	LED 20	bzw. LED 7 grün
X30.21	L21_2 Katode	LED 21	bzw. LED 7 blau
X30.22	L22_2 Katode	LED 22	bzw. LED 8 rot
X30.23	L23_2 Katode	LED 23	bzw. LED 8 grün
X30.24	L24_2 Katode	LED 24	bzw. LED 8 blau
X30.25	+Uext LED-Spannung		
X30.26	GND	Masse	
X30.27	+Uext LED-Spannung		
X30.28	GND	Masse	
X30.29	+Uext LED-Spannung		
X30.30	GND	Masse	
X30.31	+Uext LED-Spannung		
X30.32	GND	Masse	
X30.33	+Uext LED-Spannung		
X30.34	GND	Masse	
X30.35	+Uext LED-Spannung		
X30.36	GND	Masse	
X30.37	+Uext LED-Spannung		
X30.38	GND	Masse	
X30.39	+Uext LED-Spannung		
X30.40	GND	Masse	

Anschluss LED 25-32 (optional)

X40. 1	L25_2 Katode	LED 25
X40. 2	L26_2 Katode	LED 26
X40. 3	L27_2 Katode	LED 27
X40. 4	L28_2 Katode	LED 28
X40. 5	L29_2 Katode	LED 29
X40. 6	L30_2 Katode	LED 30
X40. 7	L31_2 Katode	LED 31
X40. 8	L32_2 Katode	LED 32
X40. 9	+Uext LED-Spannung	
X40.10	GND	Masse

4 Allgemeines

4.1 Literaturverzeichnis

Theorie

- [1] Volles Spektrum: Farbgenerator mit RGB-Mischdioden
Helmut Israel
FA 2001/7; S.754
- [2] Farbexperimente mit LEDs (1, 2)
Dipl.-Ing. Andreas Köhler
FA 2003/5,6; S.478/585
- [3] Neue RGB-LEDs - alter Hut oder mehr Farben?
Dr.-Ing. Klaus Sander
FA 2003/12; S. 1232
- [4] RGB-LEDs durch HSV-Farbsystem einfacher ansteuerbar
Dr.-Ing. Klaus Sander
FA 2010/8; S. 1066

AVR, TWI

- [5] http://www.mikrocontroller.net/artic-les/AVR_TWI
- [6] <http://www.rn-wissen.de/index.php/TWI>

Software-Entwicklungsumgebung

- [7] BASCOM
<http://www.mcselec.com>
- [8] Fuse-Calc
<http://www.engbedded.com/fusecalc>

Programmierinterface

- [9] USBasp - USB programmer for Atmel AVR controllers
<http://www.fischl.de/usbasp>

Datenblätter

- [10] Datenblätter Mikrocontroller (Fa. Microchip)
<http://www.microchip.com/design-centers/8-bit>

Lieferant für LEDs

- [11] <http://www.leds24.com>
- [12] <http://www.led1.de/shop/widerstandsrechner/widerstandsrechner.html>

Baugruppe

- [13] Serielle Ansteuerung von LC-Displays via Zweidrahtbus
Ingolf Bauer
FA 2012/8; S. 826

4.2 Dateienverzeichnis

Für das Arbeiten mit der Dokumentation können der kostenfreie Adobe- oder Foxit-Reader verwendet werden, da alle Dateien im *.pdf Format veröffentlicht sind.

Für das Öffnen der Target-Datei (T3001-Format) kann die kostenfrei angebotene Version vom Leiterplattenhersteller Beta-LAYOUT (<https://de.beta-layout.com/leiterplatten/technik/downloads/>) eingesetzt werden.

Dateien der Dokumentation

01. ULS (001 Kurzinformation).pdf	Vorstellung der Elektronikbaugruppe
02. ULS (002 Dokumentation).pdf	Dokumentation (geöffnet)
03. ULS (003 Fuse-Lock-Bits).pdf	Einstellung für MC
04. ULS (100 Blockschaltbild).pdf	
05. ULS (101 Stromlaufplan).pdf	
06. ULS (V x).T3001	Leiterplatte (kpl.); x: aktuelle Version

Hinweis

Folgende Dateien können bei Bedarf aus der beiliegenden Target-Datei generiert werden:

- Blockschaltbild
- Stromlaufplan
- Bestückungsplan
- Löt- und Bestückungsseite der Leiterplatte

Programmbeispiel (Quelltext)

10. ULS_Lichterkerne-IHV (V x).bas	Quellcode (BASCOS); x: aktuelle Version
------------------------------------	---

Hinweis

Der Quellcode kann mit dem SDK eingelesen, kompiliert und in den MC übertragen werden. Die Angaben zu den Fuse- und Lock-Bits sind zu beachten.

4.3 Elektronische Bauelemente

Für Elektronikamateure, die in Ihrer Nähe keinen Fachhandel für elektronische Bauteile haben oder denen die Bestellung, zum Beispiel beim Elektronikversand Reichelt (<http://www.reichelt.de/>) für ein oder zwei Fehlteilen zu aufwendig sein sollte, können wir (nur innerhalb von Deutschland) eventuell Hilfe bei der Beschaffung anbieten.

Das Angebot an Leiterplatten ist abhängig von der Nachfrage und der jeweiligen Baugruppe. Die Leiterplatten sind ein- oder zweiseitig, gebohrt und beschichtet; zweiseitige seit mehreren Jahren durchkontaktiert.

4.4 Hinweise zu den Nutzungsrechten

Die Bauanleitungen (einschließlich eventueller Software) darf unter den folgenden Bedingungen frei kopiert oder weitergegeben werden:

- es darf kein Preis für die Dokumentation (*.zip) erhoben werden, außer einer angemessenen Kopiergebühr
- für einen kommerziellen Vertrieb der Baugruppen ist das Einverständnis des Autors erforderlich
- die Verwendung von Teilen der Dokumentation in eigenen Publikationen ist erlaubt, sofern ein eindeutiger Hinweis auf die Quelle erfolgt
- eine eventuell der Baugruppe beigelegte Software kann als Shareware konzipiert sein; dann ist sie zeitlich nicht limitiert, besitzt jedoch nur in der registrierten Version ihren vollen Funktionsumfang

Die Registrierung der Software ermöglicht Ihnen, Hinweise über die neueste (eventuell fehlerbehebene) Version zu erhalten. Wir haben uns bemüht, fehlerfrei zu programmieren. Die Erfahrung mit vielen Programmen zeigt aber, daß Software der „Version 1.0“ selten fehlerfrei funktioniert.

Hardwarefehler sind auf Grund der Anfertigung von Mustergeräten weitestgehend ausgeschlossen; Maßabweichungen bei der Bestückung können durch verschiedene Bauformen der Bauelemente auftreten.

Sollten Sie glauben, einen Fehler gefunden zu haben, senden Sie uns bitte per Email folgendes zu:

- eine kurze, aber präzise Beschreibung für die Reproduzierbarkeit des Fehlers
- eine Beschreibung der eingesetzten Hard- und Software

Ohne diese Informationen ist eine Bearbeitung nicht möglich.

Einige Programme laufen unter Visual Basic (VB) der Versionen 4/5. Die jeweilige *.exe funktioniert daher nur unter Windows, wenn VB oder ein zusätzliches Softwarepaket mit den erforderlichen Dateien installiert ist.

Anwender, die darüber nicht verfügen, können sich eine Übersicht über die Funktionalität des Programms verschaffen. Das zusammengestellte Material beschreibt in Kurzform die Funktionalität der Software.

Da die entsprechenden Installationsroutinen für Anwender ohne VB recht groß sind (ca. 1 ... 5 MB), macht eine Verteilung über Online-Dienste kaum einen Sinn. In diesem Fall sollte per Email unter Angabe des Elektronikmoduls ein Informationsblatt angefordert werden.

In der Zukunft entstehende PC-Programme werden mit der Programmiersprache PROFAN entwickelt. Möglich sind auch Excel-Dateien mit Makrounterstützung.

Für Baugruppen, in denen Mikrocontroller (AVR der Fa. Microchip) eingesetzt werden, dient als SDK BASCOM (Basicdialekt). BASCOM ist bis zu einer Programmgröße von 4 kByte ohne Einschränkungen verwendbar und unserer Meinung nach recht schnell zu beherrschen.

4.5 Technische Hinweise

4.5.1 Ausdrucken der Dateien

Der Ausdruck der Dateien sollte keine Probleme bereiten.

Im Gegensatz zu den Frontplattenbeschriftungen, die im Maßstab 1:1 vorliegen, sind die Leiterplattenzeichnungen der ersten Module im Maßstab 2:1 erstellt, was bei der Fertigung beachtet werden muß.

Für die Leiterplattenentwicklung jüngerer Datums wurde die vom Leiterplattenhersteller Beta-LAYOUT (<https://de.beta-layout.com/leiterplatten/technik/downloads/>) angebotene kostenfreie TARGET-Version eingesetzt.

Bei der Erstellung des Layouts der Leiterplatten wurde auf eine einfache Leitungsführung geachtet. Allerdings sind nicht mehr alle Bauelemente in einem DIL-Gehäuse verfügbar, so dass häufiger die SMD-Technik zum Einsatz kommt.

Die Bestückung gestaltet sich dann zwar etwas komplizierte, es kann aber Platz gespart werden, was geringere Kosten bei der Leiterplattenherstellung bedeutet.

Es ist bei der Leiterplattenbestellung ebenfalls möglich, einen Bestückungsauftrag zu erteilen.

4.5.2 Gehäuse und Frontplattenbeschriftungen

Sofern bei einer Baugruppe ein größeres Gehäuse verwendet wurde, handelt es sich um ein Kunststoff-Halbschalen-Gehäuse, was zum Beispiel über den CONRAD-Elektronikversand bezogen werden kann.

Diese Gehäuseform hat den Vorteil, daß mit geringem Aufwand eine ansprechende Frontplatte gestaltet werden kann. Die den jeweiligen Baugruppen beigelegte Datei „* (Frontplatte).pdf“ beinhaltet jeweils einen Vorschlag. Sie können auf entsprechenden Papier (nicht zu dickes verwenden) ausgedruckt und unter Verwendung eines Klebestiftes auf die Frontplatte geklebt werden. Nach dem Trocknen kann noch eine selbstklebende Transparentfolie aufgeklebt werden. Mit einer Rasierklinge lassen sich jetzt überstehende Papier- und Folienreste schnell entfernen. Entsprechend den Konturen kann nun gekörnt und vorhandene Ausschnitte mit einer Laubsäge angefertigt werden. Nach Montage der Bedienelemente und der Verdrahtung mit der Leiterplatte kann die Montage des Gehäuses durchgeführt werden.

Stecker- oder Hutschienegehäuse können vom gleichen Anbieter, Reichelt-Elektronik oder direkt vom Hersteller (BOPLA) bezogen werden.

4.5.3 Bauelementeauswahl

Bauelemente ohne besonderen Hinweis sind als Vorschläge zu verstehen. An diesen Stellen können natürlich Typen verschiedener Hersteller zum Einsatz kommen, sofern sich ihre elektrischen Parameter gleichen.

Weitergehende Erläuterungen und Hinweise zum Aufbau elektronischer Schaltungen werden als bekannt vorausgesetzt bzw. können aus entsprechenden Literaturquellen bezogen werden.

4.5.4 SMD-Bestückung

Leider gibt es einige Bauelemente, die nicht in einem DIL-Gehäuse angeboten werden. Die Bestückung der Leiterplatte gestaltet sich daher komplizierter. Im Internet findet man zum Thema auch mehrere Anregungen und Tipps.

Zur Bestückung mit dem FTDI232RL wurde die LötKolbenspitze durch einen angespitzten 1,5 mm² Kupferdraht ersetzt. Die Leistung des LötKolbens mit 30 Watt ermöglicht ein gutes Fließen des Lötzinns. Eine Lupe, ausreichende Beleuchtung und eine ruhige Hand sind weitere Voraussetzungen für die Montage des ICs.

Anfragen, egal zu welcher Problematik, beantworten wir gerne per Email.

Viel Spaß beim Aufbau und Einsatz der Baugruppe wünscht Ihnen das

Amatronik Entwicklungsteam