



Segmentanzeigen-Rechner

Bei der Entwicklung von Mikrocontroller-Schaltungen mit Ausgabe durch Segmentanzeigen stellt die Berechnung der Bitmuster für die Ansteuerung für den Entwickler eine mühselige und zeitintensive Arbeit dar. Der hier vorgestellte Segmentanzeigen-Rechner erleichtert diese Arbeit. Er dient als Hilfsmittel zur Festlegung von benutzerdefinierten Zeichen und zur automatischen Berechnung der zugehörigen Bitmuster auch bei individuell zu wählenden Anschlussreihenfolgen zwischen Mikrocontroller und Anzeige.

Effektiver programmieren

Bei der Verwendung von Segmentanzeigen in Mikrocontroller-Schaltungen, seien es 7-, 14- oder 16-Segment-Anzeigen, tritt immer wieder das Problem auf, mit welchem Datenbyte die gewünschte Zahl oder das gewünschte Zeichen auf der Anzeige erzeugt wird. Die Berechnung der Ansteuerungsbitmuster für die unterschiedlichen Zeichen ist im Regelfall eine mühselige und zeitintensive Arbeit. Der „ELV Segmentanzeigen-Rechner“ schafft hier Abhilfe. Er soll als Hilfsmittel zur Festlegung von benutzerdefinierten Zeichen für Segmentanzeigen dienen. Der Rechner, ein kleines Windows-Programm, das Sie kostenlos aus unserem Internet-Angebot herunterladen können, gibt zu den eingestellten Segmenten das zugehörige Binär- oder Hex-Muster aus. Weiterhin lässt sich der verwendete Segmentanzeigen-Typ über ein Konfigurationsmenü auswählen und die Anschlussreihenfolge an den jeweiligen Mikrocontroller festlegen, um das Layout einer Platine zu vereinfachen. Nach dem Anklicken der Segmente der gewünschten Zahl oder des gewünschten

Zeichens wird der zugehörige Binär- oder Hex-Code berechnet und angezeigt.

Segmentanzeigen

Segmentanzeigen werden in unterschiedlichen Konfigurationen angeboten. Es gibt 7-Segment-, 14-Segment- oder 16-Segment-Anzeigen, die Bezeichnung schließt gleich auch die Anzahl der Anzeigesegmente ein. Der Dezimalpunkt (DP) ist nicht in jeder Segmentanzeige implementiert und wird deshalb aus der Berechnung ausgeschlos-

sen. Eine schematische Darstellung der unterschiedlichen Segmentanzeigen-Typen ist in Abbildung 1 zu sehen. Links in dieser Abbildung ist eine 7-Segment-Anzeige dargestellt. Eine 7-Segment-Anzeige wird hauptsächlich zur Anzeige von Zahlen verwendet, darauf werden wir noch zurückkommen.

In der Mitte der Abbildung 1 ist eine 14-Segment-Anzeige zu sehen. Mit einer 14-Segment-Anzeige lassen sich die Zahlen 0 bis 9 und die gesamten Großbuchstaben abbilden. Zusätzlich sind noch Kleinbuch-

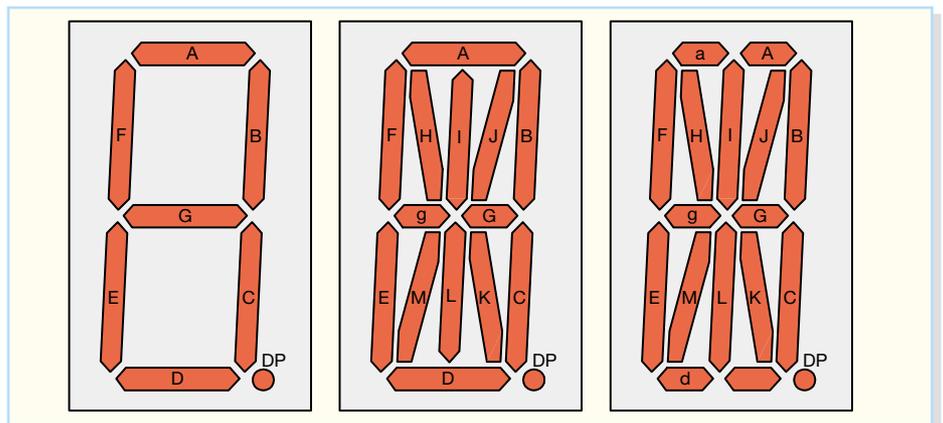


Bild 1: Die hier behandelten und gängigen Segmentanzeigen-Typen

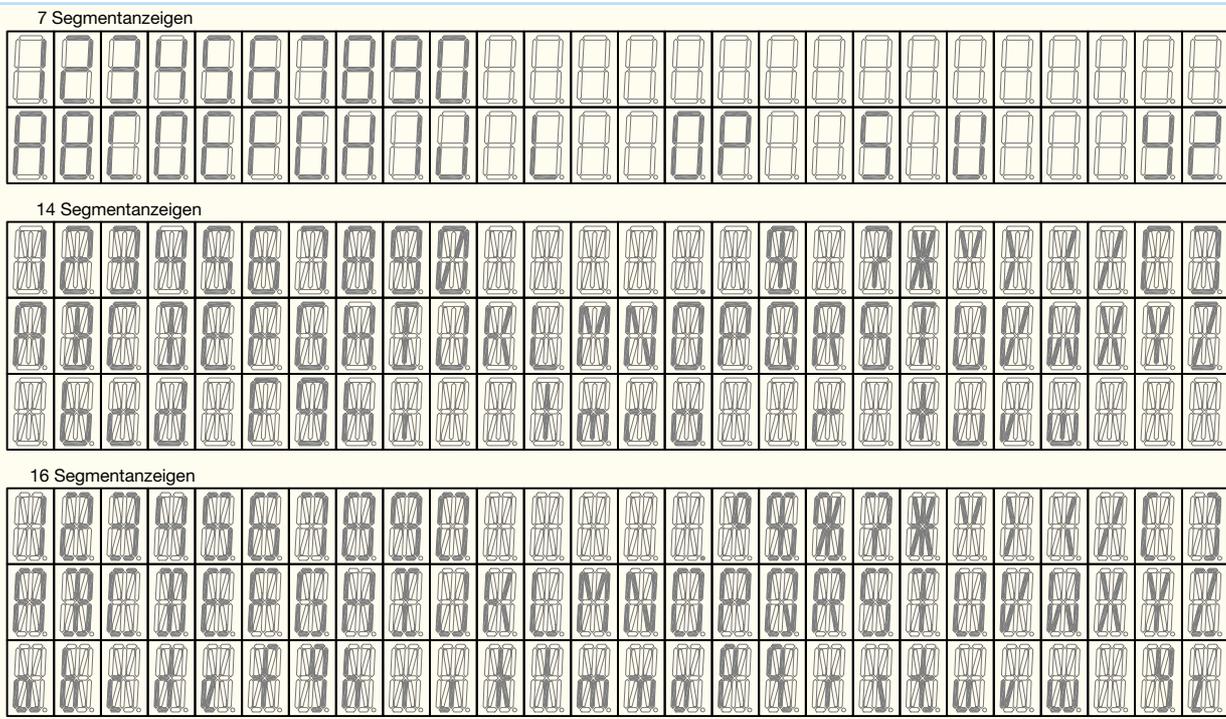


Bild 2: Die Zeichensätze der drei Segmentanzeigen-Typen

staben (beispielsweise das „b“ oder das „w“) und noch einige Sonderzeichen (beispielsweise eine Klammer „[]“) anzeigbar.

Rechts in Abbildung 1 ist eine 16-Segment-Anzeige dargestellt. Sie ähnelt der 14-Segment-Anzeige, mit dem Unterschied, dass die Segmente „A“ und „D“ nochmals geteilt sind. Dies hat den Vorteil, dass mit einer 16-Segment-Anzeige neben den Zahlen 0 bis 9 und den Großbuchstaben noch fast alle Kleinbuchstaben angezeigt werden können. Weiterhin sind noch mehrere Sonderzeichen realisierbar. In Abbildung 2 sind die mit der jeweiligen Anzeige darstellbaren

Anschlussleitung nach außen geführt ist. Zur Einsparung von Anschlusspins besitzen die Segmentanzeigen meist eine gemeinsame Katode oder eine gemeinsame Anode. Das bedeutet, dass die einzelnen Segmente einer Segmentanzeige mit gemeinsamer Katode mit positivem Potential, beispielsweise von einem Mikrocontroller erzeugt, anzusteuern sind. Diese Ansteuerungsart wird als statische Ansteuerung bezeichnet. Der Anschluss für das GND-Potential der Segmente wurde zusammengefasst und auf einen Anschlusspin herausgeführt. In der Abbildung 3 ist eine

muster für die unterschiedlichen Zeichen ist immer wieder eine mühselige und zeitintensive Arbeit. Der „ELV Segmentanzeigen-Rechner“ schafft hier als Hilfsmittel zur Zeichen- und Belegungsfestlegung und automatischen Bitmuster-Berechnung Abhilfe.

In der Abbildung 4 ist das Dialogfeld des Segmentanzeigen-Rechners dargestellt.

Im Feld „Anzeige“ ist die ausgewählte Segmentanzeige, in diesem Fall eine 7-Segment-Anzeige, zu sehen. Im Menüfeld „Auswahl“ lassen sich die weiteren möglichen Segmentanzeigen-Typen (14- oder 16-Segment) auswählen.

Die Abbildung 5 zeigt eine mögliche Ansteuerung einer 7-Segment-Anzeige durch einen Mikrocontroller. Die Segmentanzeige ist durchgängig an den Port 1 eines Mikrocontrollers angeschlossen. Die Anschlussreihenfolge beginnt mit Segment „A“ an Port 1.0 und endet mit dem Dezimalpunkt an Port 1.7.

Zur Auswahl der einzelnen Zahlen oder Zeichen sind im Feld „Anzeige“ des Seg-

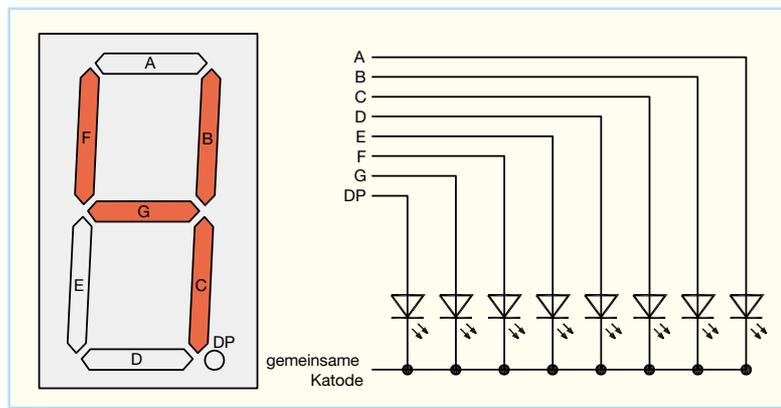


Bild 3: Der Aufbau einer 7-Segment-Anzeige

Zeichen aufgeführt. Eine Lücke bedeutet, dass das jeweilige Zeichen gar nicht oder nur unzureichend mit dem gewählten Segmentanzeigen-Typ dargestellt werden kann.

Aufbau einer 7-Segment-Anzeige

Im Folgenden ist der innere Aufbau einer Segmentanzeige am Beispiel einer 7-Segment-Anzeige dargestellt. Diese besteht intern aus mehreren Leuchtdioden, deren Leuchtfläche auf die jeweilige Segmentgröße angepasst ist. Die Segmentanzeigen besitzen (außer bei wenigen Mehrfachanzeigen) keinen Ansteuerungscontroller, so dass für jedes Segment eine

Segmentanzeige mit gemeinsamer Katode dargestellt. Zur Ansteuerung der abgebildeten Ziffer 4 sind die Segmente „B“, „C“, „F“ und „G“ mit einer positiven Spannung anzusteuern. Der innere Aufbau einer 14- oder 16-Segment-Anzeige gestaltet sich ähnlich. Hier werden jeweils 14 oder 16 Leuchtdioden für die unterschiedlichen Segmente verwendet. Des Weiteren sind hier noch weitere Ansteuerungsarten (dynamische Ansteuerung) möglich, worauf wir an dieser Stelle jedoch nicht eingehen wollen.

ELV Segmentanzeigen-Rechner

Die Berechnung der Ansteuerungsbit-



Bild 4: Die Bedienoberfläche des ELV-Segmentanzeigen-Rechners

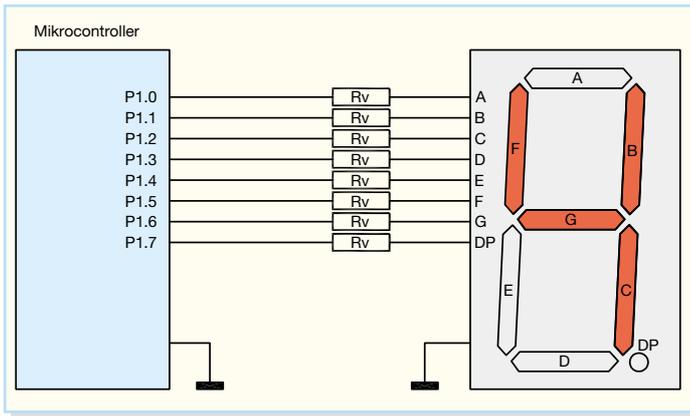


Bild 5: Beispiel 1 für die Ansteuerung einer 7-Segment-Anzeige mit einem Mikrocontroller

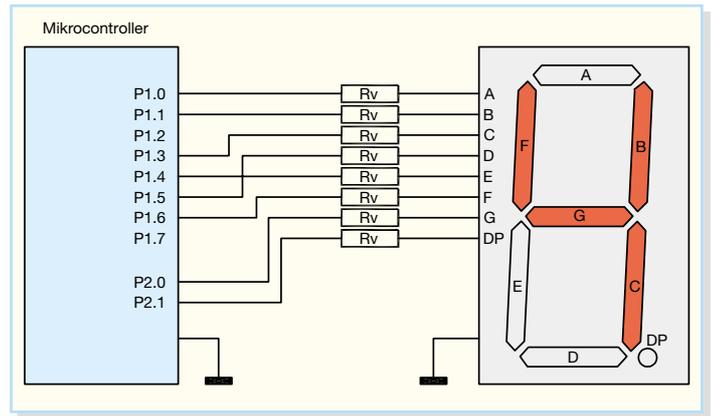


Bild 7: Beispiel 2 für die Ansteuerung einer 7-Segment-Anzeige mit einem Mikrocontroller

mentanzeigen-Rechners die einzelnen Segmente durch Anklicken einzustellen. Im „Anzeigen-Fenster“ der Abbildung 4 ist eine „4“ dargestellt. Das Ausgabefeld „Segmentfolge“ (unterhalb der Anzeige) zeigt die aktuell eingestellte Segmentfolge an.

```

//+++++
// Zifferncodetabelle für 7-Segment-Anzeigen
//
//      01
// 20 02
//      40
// 10 04
//      08 80
SEGMENTE      .DB      03Fh      ;0 '0'
               .DB      006h      ;1 '1'
               .DB      0dBh      ;2 '2'
               .DB      04Fh      ;3 '3'
               .DB      066h      ;4 '4'
               .DB      0EDh      ;5 '5'
               .DB      0FDh      ;6 '6'
               .DB      007h      ;7 '7'
               .DB      07Fh      ;8 '8'
               .DB      0EFh      ;9 '9'

```

Bild 6: Die fertige Zeichensatzdefinition in einem Assemblerlisting

Hier ist die Segmentfolge durchgängig ausgewählt, d. h. das niederwertigste Bit ist das Segment „A“ und das höchstwertige Bit ist der Dezimalpunkt („P“). Im Ausgabefenster „binär“ wird die zugehörige Binärfolge angezeigt. Diese ist abhängig von der eingestellten Segmentfolge. Wird bei-

spielsweise das Segment „A“ aktiviert, so wird an der Stelle des Segmentes „A“ (abhängig von der eingestellten Segmentfolge) eine „1“ geschrieben. In der Abbildung 4 sind die Segmente „B“, „C“, „F“ und „G“ aktiviert. Der Binär-Code lautet somit 01100110. Aus diesem Binär-Code wird der zugehörige Hex-Code generiert und im Ausgabefeld „hex“ angezeigt. Er lautet hier 66. Diese Zeichenfolgen bzw. die Hexfolge wird benötigt, um die Zeichensatz-Tabelle für die Ansteuerung einer Segmentanzeige zu füllen. In der Abbildung 6 ist ein Ausschnitt einer solchen Definition einer Assembler-Programmoberfläche dargestellt. In der abgebildeten Definitionstabelle sind die Hex-Kombinationen für die Zahlen 0 bis 9 angegeben. Bei der Verwendung eines Zeichens innerhalb der zu

erstellenden Software ist die jeweilige Hex-Kombination aus dem Feld „Segmente“ auszuwählen und auf den Datenbus der Segmentanzeige zu legen. In der Zeile 4 ist die in der Abbildung 4 errechnete Hex-Kombination (066h) wiederzufinden.

Durch die Taste „Ändern“ im Feld „Seg-

mentfolge“ lässt sich die aktuell eingestellte Segmentfolge ändern. Ist es beispielsweise aus layouttechnischen Gründen nicht möglich, die Segmentanzeige fortlaufend an einen Mikrocontroller anzuschließen (Abbildung 7 zeigt ein solches Beispiel), so ist die Segmentfolge abzuändern. Diese Änderung wird durch Klicken der Taste „Ändern“ im Feld „Segmentfolge“ gestartet. Nun sind die Segmente gemäß ihrer Segmentfolge anzuklicken. Die Festlegung der Segmentfolge ist für das erwähnte Ansteuerungsbeispiel 2 in Abbildung 8 dargestellt. Begonnen wird die Festlegung mit dem niederwertigsten Bit. Im Ansteuerungsbeispiel 2 (Abbildung 7) ist an Port 1.0 das niederwertigste Bit, das Segment „A“, angeschlossen. Ein nicht verwendeter Portpin, wie etwa der Port 1.2 im Beispiel, wird mit dem „Minus“ (-) in die Segmentfolge aufgenommen. Dieser Portpin wird bei der späteren Berechnung nicht berücksichtigt und bleibt auf „0“. Nach Eingabe der gesamten neuen Segmentfolge ist die Taste „Beendet“ zu betätigen. Die geänderte Segmentfolge wird gespeichert und im Ausgabefeld „Segmentfolge“ ausgegeben. In der Abbildung 9 ist die geänderte Segmentfolge zu sehen. Die eingestellte „4“ ergibt nun eine andere Hex-Kombination (014Ah). Dabei ist zu beachten, dass zur Anzeige einer „4“ auf der Segmentanzeige an den Port 1 eine 4Ah und an den Port 2 ein 01h anzulegen ist. Die nicht belegten Pins (Port 1.3 und Port 1.7) wurden aus der Berechnung ausgeblendet und auf „0“ geschaltet.

Damit steht mit dem Segmentanzeigen-Rechner ein einfach und flexibel zu handhabendes Tool zur Verfügung, das dem Entwickler die Arbeit deutlich erleichtern kann. **ELV**

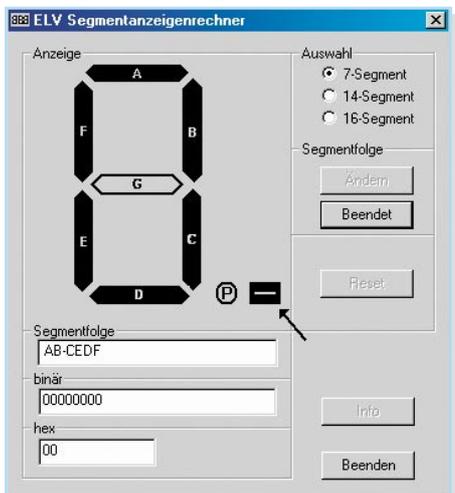


Bild 8: Die Festlegung der Segmentfolge gemäß Ansteuerungsbeispiel 2

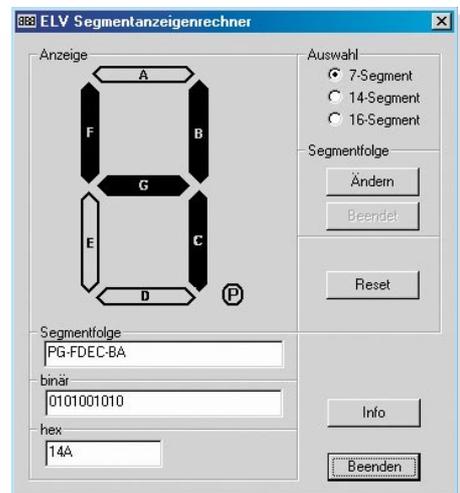


Bild 9: Das Berechnungsergebnis der geänderten Segmentfolge

Kostenloser Download des Segmentanzeigen-Rechners unter: www.elvjournals.de