

LED-Display mit MAX7219

Gunther Zielosko

1. Einführung

Wer das 4-zeilige alphanumerische LC-Display des Plug-and-Play-Labs kennt, wird mit dessen Möglichkeiten mehr als zufrieden sein. Allerdings gibt es auch Anwendungen des BASIC-Tiger®, die andere Varianten der Meßwertanzeige benötigen. Sei es, daß

- Zahlen bei schlechten Beleuchtungsverhältnissen dargestellt werden sollen,
- große Ziffern gebraucht werden,
- bei eigenständige Lösungen das Plug-and-Play-Lab nicht zur Verfügung steht,
- einfach Geld gespart werden soll und der Kauf eines weiteren LC-Displays nicht lohnt,
- andere Darstellungsformen gewünscht werden, wie z.B. Leuchtbalken oder "british flag",

oder andere Gründe, die für den Einsatz von LED's in verschiedenen Formen sprechen.

Typische Vertreter der LED-Displays sind die sogenannten 7-Segment-Anzeigen, die mit 7 "Leuchtbalken" (Segmenten) alle Ziffern darstellen können. Rechnet man das fast immer vorhandene Komma (DP) mit dazu, sind es 8 Segmente pro Stelle (Digit). Bild 1 zeigt ein LED-Display in Aktion, Bild 2 die gebräuchliche Bezeichnung der einzelnen Segmente.



Bild 1: typische 7-Segment-LED-Anzeige

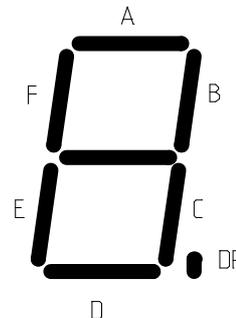


Bild 2: Segmentzuordnung

Bisher gab es bei der Ansteuerung von LED-Displays mit Mikrorechnern, zu denen auch der BASIC-Tiger® gehört, immer eine Reihe von Problemen, von denen einige im folgenden betrachtet werden sollen. Fangen wir mit den ganz gewöhnlichen 7-Segment-LED-Displays (siehe oben) mit mehreren Stellen an.

Zuerst ist es nicht ganz einfach, Ziffern bzw. Zahlen in Tiger-Basic in die bei diesen LED-Displays übliche 7-Segmentform zu bringen, da Tiger-Basic diese Darstellungsform nicht unterstützt (eigentlich schade!).

Auf der Hardwareseite sieht es auch nicht besser aus, in der Regel werden viele Anschlüsse (Ports) gebraucht, um die einzelnen Segmente erst einmal logisch anzusteuern. Wenn man

dann noch Leistung braucht (damit die LED's auch hell leuchten), geht es nicht ohne Verstärker für jedes Segment ab. Etwas trickreicher funktioniert es mit dem sogenannten Multiplex-Verfahren, hier werden die einzelnen Displaystellen (Digits) nacheinander aktiviert und der jeweils zum Digit dazugehörige 7-Segment-Code an die Einzelelektroden (Segmente) gegeben. Während man im "Direktbetrieb" für z.B. 8 Digits 64 Leitungen anzusteuern hätte (8 Digits x 7 Segmente + DP), sind es im Multiplexbetrieb nur noch 8 (Segmente) plus 8 (Digits), also 16. Allerdings ist der Ansteueraufwand höher, denn hier ist die Steuereinheit mit der Aufteilung der einzelnen Signale stark beschäftigt. Zudem muß dies wegen der Flimmerfreiheit noch sehr schnell gehen. Zusätzlich ist auch die Hardware stärker gefordert, denn es steht für jedes Digit nur noch ein Achtel der Gesamtzeit zur Verfügung, was bei gleicher Helligkeit rein rechnerisch den achtfachen Strom bei den Segmenten gegenüber der Einzelansteuerung erfordert.

Eine schwierige Aufgabe, wenn es nicht den MAX7219 von MAXIM gäbe, der alle diese Aufgaben allein bewältigt. Ihm reichen drei Portleitungen, über die er alle Informationen seriell vom BASIC-Tiger® bekommt. Die Ansteuerung von LED-Displays aller Art, also 7-Segment-Displays, aber auch exotische Varianten wie Bargraf- oder "british flag"-Displays werden problemlos bedient. Wie der MAX7219 mit dem BASIC-Tiger® funktioniert, wollen wir in diesem Applikationsbericht behandeln.

2. Der MAX7219

2.1. Die Zusammenschaltung mit dem BASIC-Tiger®

Das 24-polige DIL-IC beinhaltet sämtliche Treiber für die LED's, erledigt das komplette Timing für die multiplexe Ausgabe und speichert die seriell eingeschriebenen Informationen solange, bis neue kommen. Dabei ist die Änderung einzelner Stellen (Digits) möglich. Weiterhin kann die Helligkeit der Anzeige mit zwei Methoden verändert werden, per Widerstandseinstellung an ISET (z.B. mit Fotowiderstand) oder per Programmierung. Ebenfalls komfortabel können verschiedene Anzeigemodi über Register eingestellt werden, dies ist besonders dann wichtig, wenn keine 7-Segment-Anzeigen angesteuert werden sollen (z.B. Bargraf-Anzeigen usw.). Ein Datenblatt des MAX7219 ist unter:

<http://209.1.238.250/arpdf/1339.pdf>

aus dem Internet abzurufen.

Die folgende Abbildung zeigt die Pinbelegung des MAX7219 und die einfache Zusammenschaltung mit dem BASIC-Tiger[®]:

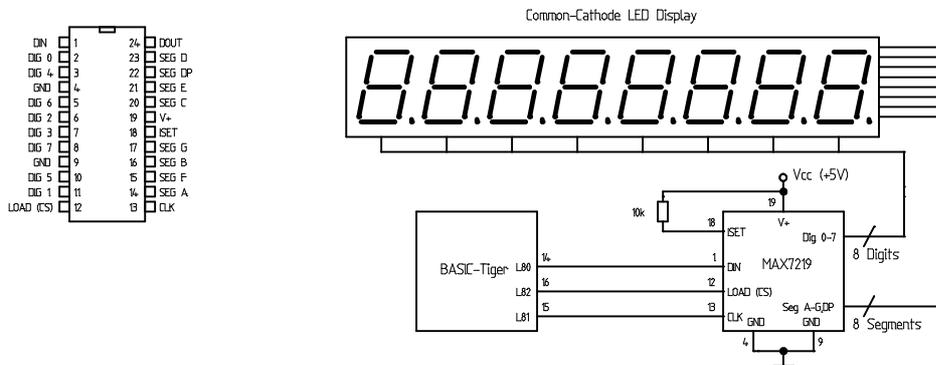


Bild 3: Zusammenschaltung BASIC-Tiger[®] mit MAX7219 (Digit 0 des Displays ist rechts!).

2.2. Die Programmierung des MAX7219

Wie wir gesehen haben, ist der Hardwareaufwand für das LED-Display sehr gering. Der Programmieraufwand dagegen ist wegen der vielen Möglichkeiten etwas höher. Zunächst werden wir das Prinzip der Programmierung des MAX7219 behandeln und später konkrete Beispiele mit dem BASIC-Tiger[®] ausprobieren.

Jeder "Datensatz" für den MAX7219 besteht aus 16 Datenbits, die wie folgt angeordnet sind:

- D 0 - D 7 darzustellende Zeichen oder Steuerinformationen
- D 8 - D 11 Zieladresse der Daten, entweder die Speicheradresse für das Digit oder die Adresse des jeweiligen Steuerregisters
- D 12 - D 15 nicht benutzt (z.B. immer 0 setzen)

Die Datenübertragung beginnt immer mit dem Bit 15 (MSB), das ist bei der Bereitstellung der Daten im BASIC-Tiger[®] zu beachten.

Als nächstes wollen wir uns mit den Zieladressen (D 8 - D 11) beschäftigen, die ja entweder das Digit oder bestimmte Steuerfunktionen festlegen. Die folgende Tabelle 1 zeigt die komplette Registerbelegung:

Hexadezimal	Adresse				Funktion	Name
	D11	D10	D9	D8		
0h	0	0	0	0	Keine Funktion	No-Op
1h	0	0	0	1	Direktadressierung Digit 0	Digit 0
2h	0	0	1	0	Direktadressierung Digit 1	Digit 1
3h	0	0	1	1	Direktadressierung Digit 2	Digit 2
4h	0	1	0	0	Direktadressierung Digit 3	Digit 3
5h	0	1	0	1	Direktadressierung Digit 4	Digit 4
6h	0	1	1	0	Direktadressierung Digit 5	Digit 5
7h	0	1	1	1	Direktadressierung Digit 6	Digit 6
8h	1	0	0	0	Direktadressierung Digit 7	Digit 7
9h	1		0	1	Dekodier-Mode	Decode Mode
Ah	1	0	1	0	Helligkeit	Intensity
Bh	1	0	1	1	Stellen-Auswahl	Scan Limit
Ch	1	1	0	0	Display abschalten	Shutdown
Dh	1	1	0	1		
Eh	1	1	1	0		
Fh	1	1	1	1	Display-Test	Display Test

Tabelle 1 Datenbits 8 - 11, die Register-Adressen des MAX7219

In diese Register können nun Daten geschrieben werden, die je nach Zieladresse verschiedene Wirkungen haben. Fangen wir mit den reinen Steuerdaten an.

Register No-Op

Dieses Register braucht man eigentlich nur, wenn mehrere MAX7219 kaskadiert werden und davon nur einzelne Bausteine umprogrammiert werden sollen.

Register Decode-Mode

Hier können die einzelnen Darstellungsvarianten auf dem Display bzw. der Einzel-LED-Anordnung gewählt werden. Prinzipiell gibt es für jedes Digit die Möglichkeit, die 7-Segmentdarstellung oder eine nicht dekodierte Anzeige zu programmieren. Eine 0 im Datenbit heißt, daß im zugehörigen Digit keine Dekodierung erfolgt, eine 1 bedeutet, daß das zugehörige Digit im 7-Segment-Code angesteuert wird, Beispiele zeigt Tabelle 2:

Hex	Adresse								Funktion
	D7	D6	D5	D4	D3	D2	D1	D0	
00h	0	0	0	0	0	0	0	0	Keine Dekodierung* an allen Digits
01h	0	0	0	0	0	0	0	1	BCD-Code B** nur für Digit 0
0Fh	0	0	0	0	1	1	1	1	BCD-Code B** nur für die Digits 0-3
FFh	1	1	1	1	1	1	1	1	BCD-Code B** für alle Digits

Tabelle 2: Beispiele für Belegung des Decode-Mode-Register

* keine Dekodierung heißt bei log 1

D0 = Segment G leuchtet
D1 = Segment F leuchtet
D2 = Segment E leuchtet
D3 = Segment D leuchtet
D4 = Segment C leuchtet
D5 = Segment B leuchtet
D6 = Segment A leuchtet
D7 = Segment DP leuchtet

** BCD-Code B ist die normale 7-Segment-Darstellung der Ziffern 0-9, die darauf folgenden Hexadezimalwerte A - F werden als -, E, H, L, P und blank dargestellt. Das Bit 7 wird gesetzt, wenn der Dezimalpunkt (Komma) leuchten soll.

Register Intensity

Mit einem Eintrag in dieses Register wird die Display-Helligkeit eingestellt. Dazu werden nur die Datenbits D 0 bis D 3 benötigt, die Bits D 4 bis D 7 können auf 0 oder 1 liegen. Die minimale Helligkeit (praktisch dunkel) erhält man mit dem Wert 0h, die maximale mit Fh. Die Stufung erfolgt in 1/32 der Gesamtanzeigzeit des Digits, dabei ist 0h = 1/32, 1h = 3/32, 2h = 5/32 usw. bis Fh = 31/32.

Scan Limit Register

Auch dieses Register ist leicht zu programmieren. Hier werden nur die Bits D 0 bis D 2 benutzt. Bei dem Steuerwort 0h wird nur das niederwertigste Digit angesteuert (einstelliges Display), bei 1h die Digits 0 und 1 (zweistelliger Betrieb), bei 2h dreistellig usw. 7h bedeutet also, daß alle 8 Stellen angesteuert werden.

Werden nicht alle Stellen angesteuert, teilt der MAX7219 die zur Verfügung stehende Anzeigzeit auf weniger Digits auf, d.h. die Helligkeit (und damit der Strom pro Segment) nimmt zu. Das kann man hardwareseitig durch geeignete Wahl des Widerstandes an ISET oder softwareseitig durch das Intensity Register ausgleichen.

Shutdown Register

Dieses Register dient zum Abschalten des Displays, z.B. um Strom zu sparen oder das Display blinken zu lassen. Die beiden möglichen Zustände steuert Bit D 0, bei 0 ist das Display abgeschaltet, bei 1 arbeitet es normal.

Display Test Register

Mit Bit D 0 werden bei 1 alle Segmente aller Digits mit maximaler Helligkeit betrieben, d.h. alles leuchtet. Bei D 0 = 0 arbeitet das Display wieder normal.

Digit Register 0 - 7

Werden die Digit Register angesprochen, schreibt man die anzuzeigenden Daten (Bits D 0 - D 3) direkt in die Display-Speicher. Zu beachten ist noch, daß das Bit D 7 für das Komma zuständig ist. Will man an irgendeine Stelle das Komma setzen, wird dort in Bit 7 zusätzlich zur anzuzeigenden Ziffer eine "1" eingetragen. Wie schon erwähnt, können das die Ziffern 0-9 bzw. ein paar Sonderzeichen sein oder direkt adressiert einzelne Leuchtbalken (dann sind alle Bits ansprechbar). Da man jedes Digit einzeln adressieren kann, müssen auch nur die Daten übertragen werden, die sich geändert haben. Das ist vorteilhaft, wenn z.B. zwei Meßwerte von verschiedenen Unterprogrammen eingegeben werden, links Strom, rechts Spannung. Jedes Unterprogramm "bedient" dann seine Displayhälfte. Das erleichtert die Programmierung erheblich.

Übrigens, Sie können die einzelnen Stellen des Displays natürlich beliebig anordnen, da Sie ja per Software jedes Digit einzeln ansprechen können. Wenn die Digits allerdings entsprechend ihrer Wertigkeit auf dem Display erscheinen sollen, ist es sinnvoll, das Digit 0 des Displays rechts anzuordnen. Damit ist dann z.B. die Programmierung eines Zählers einfacher.

Das war's, damit sollten wir den MAX7219 mit dem BASIC-Tiger® „im Griff“ haben.

3. Programmierung des BASIC-Tiger®

3.1. Grundlagen

Es gilt nun, die drei Leitungen vom BASIC-Tiger® zum MAX7219 so anzusteuern, daß letzterer mit den Daten etwas anfangen kann.

Der Eingang LOAD (CS) des MAX7219 muß im Ruhezustand (d.h. es werden keine Daten übertragen) auf High liegen, der Takteingang CLK auf Low. Vor Beginn der Datenübertragung wird LOAD (CS) auf Low gebracht. Die Übertragung der Daten erfolgt dann mit dem SHIFT_OUT - Befehl des Tiger-BASIC, bei dem parallel zu den Datenbits synchron ein Takt übertragen wird. Wenn wir nun diesen Befehl so einsetzen, wie er für die Kopplung mit dem MAX7219 gebraucht wird, ist die Hauptarbeit getan. Dieser Befehl lautet vollständig:

SHIFT_OUT Log_iPortadr, Datenpin, Taktpin, Variable, Anzahl

Wir stellen uns die einzelnen Komponenten des Befehls einmal aus dem Hilfe-Text von Tiger-BASIC (Version 4) oder dem Handbuch zusammen und finden daraus:

Log_iPortadr ist eine Konstante vom Typ BYTE, WORD oder LONG und gibt die logische interne Portadresse an. *Wir wollen den Port 8 benutzen, hier wäre also eine 8 einzutragen.*

- **Datenpin** ist eine Konstante vom Typ BYTE, WORD oder LONG und bestimmt, an welchem Pin die Datenbits eingelesen werden. *Wir benutzen Pin L80 zum Datentransfer, also eine 0 (Bit 0 von Port 8) eintragen.*

- **Taktpin** ist eine Konstante vom Typ BYTE, WORD oder LONG und gibt an, welcher Pin als Taktpin verwendet wird. *In unserem Fall ist das Pin L81, also eine 1 (Bit 1 von Port 8).*
- **Variable** ist eine Konstante, Variable oder ein Ausdruck vom Datentyp BYTE, WORD, LONG oder STRING und wird ausgegeben. *Dies ist unser 16 Bit Datenwort, d.h. jede Ausgabe erfolgt gemäß der Tiger-BASIC-Regeln als WORD. Hier stehen sämtliche Steuerbefehle und die eigentlichen Nutzdaten drin.*
- **Anzahl** ist eine Variable, eine Konstante oder ein Ausdruck vom Typ BYTE, WORD oder LONG und gibt bei numerischen Variablen an, wieviele Bits geschrieben werden. Bei positiver Anzahl wird das niederwertigste Bit zuerst gesendet, bei negativer Anzahl das höherwertigste. *Wir wollen immer 16 Bit aussenden, d.h. hier wird generell eine 16 stehen. Mit dem Vorzeichen kann man bequem festlegen, mit welchem Bit die Übertragung beginnen soll. Da der MAX7219 gerne mit dem MSB beginnt, müssen wir -16 eintragen.*

Damit lautet unser Grundbefehl zum Aussenden von Daten immer:

SHIFT_OUT 8, 0, 1, DATEN*, -16

Daten* ist der jeweilige Inhalt der "Botschaft" an den MAX7219 und wird je nach Aufgabe gebildet.

Als letztes bleibt nur noch festzulegen, welchen Ruhepegel die einzelnen Steuerleitungen haben sollen. Wir legen fest:

L80 = DIN = Datenpin des MAX7219 = High = 1
L81 = CLK = Taktpin des MAX7219 = Low = 0
L82 = LOAD (CS) = Aktivierung des MAX7219 = High = 1

Der Rest ist einfach. Am Schluß dieses Artikels nun einige Beispielprogramme, die einige Möglichkeiten der Ansteuerung von LED-Displays zeigen:

- LED1.TIG ist ein Programm, das nacheinander auf die einzelnen Digits eines 8 -stelligen Displays die dazu passenden Ziffern von 1 - 8 schreibt.
- LED2.TIG steuert die einzelnen Segmente eines Digits der Reihe nach an.
- LED3.TIG zeigt am Beispiel der Helligkeitssteuerung die Möglichkeiten des MAX7219

Viel Spaß beim Ausprobieren weiterer Effekte!