

Versuch 2: Sieben-Segment-Anzeige

Zielstellung:

- Kennenlernen von 8-Bit-Mikrocontrollern,
- Arbeit mit einem Mikrocontrollerentwicklungssystem,
- Eingabe und Editieren von Quellprogrammen in der Programmiersprache C,
- Compilieren und Linken von Anwenderprogrammen,
- Download von Anwenderprogrammen in die Zielhardware,
- Starten und der praktische Test von Anwenderprogrammen,
- Durchführung einer indirekten Adressierung mithilfe von Zeigeroperationen

Versuchsvorbereitung

Für die Versuchsdurchführung ist es hilfreich, eine knapp gehaltene Befehlsreferenz für die Programmiersprache C, das Datenblatt des eingesetzten Controllers Dallas 80C320 und eine ASCII Tabelle zur Verfügung zu haben.

Aus der Vorlesung und Übung sollte der Umgang mit der Entwicklungsumgebung Keil μ Vision bekannt sein.

Bereiten Sie sich auf die Beantwortung von Fragen aus der separat verfügbaren Datei „Beispielfragen Praktikum Mikrotechnik.pdf“ vor.

Ermitteln Sie die Größe des Vorwiderstandes R_V entsprechend der nachfolgenden Abb. 2, wenn die Sieben-Segment-Anzeige von einem 5VOLT Low-Power-Schottky-TTL Baustein gespeist wird, die Flussspannung U_F der Diode 1,2 V beträgt und ein Diodenstrom von $I_D = 5 \text{ mA}$ einzuprägen ist!

Informieren Sie sich über den Aufbau von 7-Segment-Anzeigen, insbesondere über deren Anschlussbelegungen (gemeinsame Kathode, gemeinsame Anode)!

Die aktuelle Anleitung ist unter: [Fakultät EI, Labore E, Mikrotechnik](#) zu finden!

Versuchsaufbau

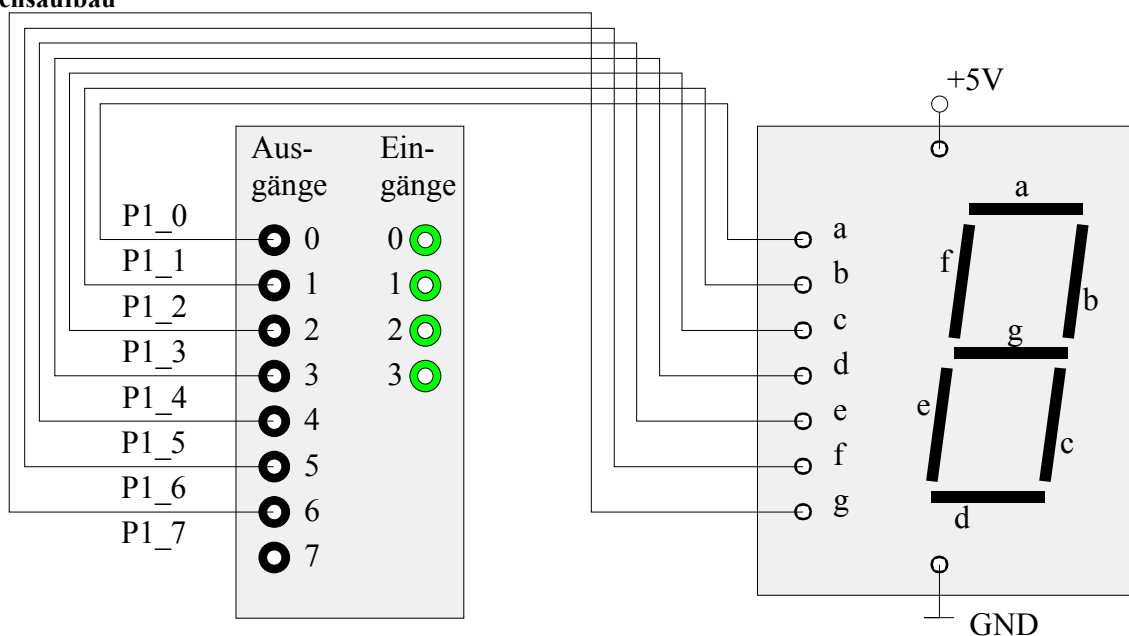
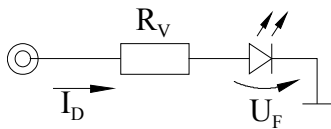


Abb.1: Verdrahtung der Ausgabelösungen mit der Sieben-Segment-Anzeige



Die Ausgangsbits des Mikrocontrollersystems P1_0 ... P1_7 mit den Anschlüssen 0, 1, 2, 3, 4, 5 und 6 sind entsprechend der nachfolgenden Abb. 1 mit den Leuchtbalken a, b, c, d, e, f und g der Sieben-Segment-Anzeige zu verbinden, das Ausgabebit P1_7 des Mikrocontrollers bleibt unbeschaltet.

Abb2: Ersatzschaltung eines Leuchtbalkens der Sieben-Segment-Anzeige

Die Ersatzschaltung des an die Ausgabeleitungen angeschlossenen Verbrauchers eines Leuchtbalkens kann mithilfe einer Leuchtdiode und eines Vorwiderstandes dargestellt werden und ist in Abb. 2 dargestellt. Bei der verwendeten Sieben-Segment-Anzeige existiert eine gemeinsame Kathode.

Aufgabenstellung:

Es ist ein Programm zu entwickeln, welches mithilfe der getchar()-Funktion eine Zahl von 0 bis 9 von der Tastatur einliest, eine Umwandlung in den Sieben-Segment-Code durchführt und anschließend eine Ausgabe auf der Sieben-Segment-Anzeige vornimmt. Die mit der PC-Tastatur eingegebene Zahl ist zusätzlich auf dem Monitor anzuzeigen!

Details zur Umsetzung (jeder Punkt erfordert zur Umsetzung einige wenige Sourcecodezeilen):

1. Innerhalb einer Endlosschleife
2. soll mit der Funktion getchar() ein Zeichen aus der seriellen Schnittstelle RS232 gelesen werden.
3. Der ASCII-Wert des Zeichens ist in einer Variablen vom Typ char zu speichern.
4. Nur Werte zwischen 0 bis 9 sollen akzeptiert, d.h. weiter verarbeitet werden.
5. In einer Unterfunktion char ascii_to_7segment(char in); ist die Transformation des ASCII-Codes der eingelesenen Zahl (möglich sind Werte von 0 bis 9) in den Sieben-Segment-Code durchzuführen.
6. Der Rückgabewert soll direkt am Port P1 ausgegeben werden können, P1_7 soll immer low sein.
7. Die Umrechnungsvorschrift ist dabei effektiv zu gestalten und stellt den Kern des zu entwickelnden C-Programms dar.
8. Die eingelesene Zahl ist zusätzlich mithilfe des printf-Befehls auf dem Bildschirm anzuzeigen.
9. Zeichen außerhalb des Bereichs 0 bis 9 sollen im Display mit '-', (Segment g) und einem Hinweistext mittels printf(); signalisiert werden.
10. Lagern Sie die erstellte Funktion in eine Datei „sieben_segement.c“ aus und erstellen Sie die nötige Header-Datei „sieben-segment.h“.
11. Binden Sie die erstellten Dateien in das Projekt ein.

Hinweise zur Erstellung des Programms:

- Die Software zur Bearbeitung ist Keil µVision. Die Software beinhaltet einen Editor, Compiler, Linker, Simulator und Debugger.
- Auf dem Laborrechner evtl. vorhandene Vorlagen für die Projekte sind nicht direkt zu nutzen. Erstellen Sie sich in jedem Fall eine Kopie der Daten! Eine Sicherung auf das eigene Homelaufwerk (Novell-Login), USB-Stick o.ä. ist in jedem Fall durchzuführen.

Listing des Roh-Programms versuch2.c

```
#include <reg320.h>           /* include 8051 header file      */
#include <stdio.h>             /* Standard I/O functions    */
#include "init.h"

void main (void)
{
    init();

    while (1) /* loop forever */
    {
        /******
        ***** hier den eigenen Programmcode einfügen *****
        *****/
    }
}
```