Praktikum 22./29. November 2010

Jörn Loviscach

Versionsstand: 26. November 2010, 23:25

In diesem Praktikum soll die Reaktionszeit eines Menschen gemessen und angezeigt werden.

Aufgabe zu Hause als Vorbereitung: Wie müsste der Code für die zweite Aufgabe aussehen? Schreiben Sie den auf oder programmieren Sie den ohne Display.

Das C-Projekt basiert auf der kleinen Funktionsbibliothek, die aus display01.c und display01.h besteht. Die Funktionsnamen und -parameter sollten selbsterkärend sein.

Schließen Sie das Display (Link zur Anschlussbelegung) so an das LaunchPad: jeweils Versorgungsspannung und Masse verbinden, Pin 1.0 an RS (Register Select, hier ein Signal zur Unterscheidung zwischen Befehlen und Daten), Pin 1.1 an E (Enable, das Signal, eine Datenübertragung duchzuführen), Pins 1.4 bis 1.7 an die Datenleitungen D4 bis D7.

Lassen Sie den Aufbau sicherheitshalber kontrollieren, bevor Sie ihn mit dem PC verbinden.

Im Prinzip kann dieses Display zwei Zeilen mal acht Zeichen darstellen; dazu müsste man aber eine negative Spannung an seinen Pin 3 anlegen (V_0). Der Einfachheit halber betreiben wir das Display deshalb nur mit der oberen Zeile.

Programmieraufgaben:

1. Prüfen Sie den Aufbau mit einer `main`-Funktion, die einige Zeichen an verschiedene Positionen des Displays schreibt.

2. Schreiben Sie in `main.c` eine Funktion mit dem Prototypen `void write(unsigned int a)`, die die Zahl `a` rechtsbündig und ohne führende Nullen in der ersten Zeile des Displays ausgibt. Orientieren Sie sich dabei am Seminar 6. Der Fall, dass `a` gleich Null ist, soll korrekt behandelt werden.

3. Schreiben Sie in `main.c` eine Funktion mit dem Prototypen `int random(void)`, die Pseudo-Zufallszahlen generiert. Diese Funktion soll eine statische Variable `unsigned long a` (hier 32 Bit) haben, die

bei jedem Aufruf der Funktion durch `1103515245 * a + 12345` ersetzt wird (Link zu weiteren Möglichkeiten). Als Ergebnis – also als nächste Pseudo-Zufallszahl – gibt die Funktion die hinteren 15 Bits von `a` zurück. (Das stellt sicher, dass das Ergebnis in eine nichtnegative `int`-Zahl passt.) Schreiben Sie zum Ausprobieren dieser Funktion in `main` eine Endlosschleife, in der in jedem Durchlauf eine neue Zufallszahl erzeugt und eine halbe Sekunde lang anzeigt.

4. Schreiben Sie die Funktion `main` so, dass sie zehnmal wie im Video (Link) die Reaktionszeit misst und jeweils anzeigt. Die Wartezeit soll jeweils zufällig ein bis vier Sekunden betragen.

5. Bestimmen Sie in `main` die Summen der Reaktionszeiten und die Summe der Quadrate der Reaktionszeiten. (Vorsicht mit möglichen Überläufen!) Bestimmen Sie daraus den Mittelwert und die Varianz, alles in ganzen Millisekunden. Zeigen Sie nach Ende der Reaktionszeitmessungen den Mittelwert und die Varianz im Wechsel an.

6. Je nach verbleibender Zeit erweitern Sie das Programm zum Beispiel so:

    - Sorgen Sie dafür, dass trotz Rechnung mit Ganzzahlen kaufmännisch gerundet wird.
    - Bestimmen Sie die Standardabweichung, also die Wurzel aus der Varianz, ohne Gleitkommazahlen zu benutzen.