Praktikum 8./9. November 2010

Jörn Loviscach

Versionsstand: 29. Oktober 2010, 00:44

In diesem Praktikum soll ein Fotowiderstand durch einen Modellbau-Servo automatisch ins optimale Licht gedreht werden.

Aufgabe zu Hause als Vorbereitung: Wie müssen die .c-Datei und die -h-Datei für die zweite Aufgabe (siehe nächste Seite) aussehen? Schreiben Sie die auf oder programmieren Sie die ohne Servo.

Erzeugen Sie ein neues C-Projekt in der IAR Embedded Workbench (Einstellungen wie bisher, also Device: MSP430G2231, Compiler: keine Optimierung, Driver: FET Debugger). Kopieren Sie die Dateien servo01.h und servo01.c neben die Datei main.c des Projekts. Fügen Sie in der Projektverwaltung der IAR Embedded Workbench die Datei servo01.c per rechtem Mausklick hinzu (Add > Add Files). Ergänzen Sie am Anfang der Datei main.c die Zeile `#include "servo01.h"`. Nun stehen Ihnen vier Funktionen zur Verfügung:

- `initialize` muss ganz zu Beginn aufgerufen werden. Es nimmt einige Grundeinstellungen vor. Die erste Zeile in `main` sollte deshalb `initialize();` lauten. Das bisherige `WDTCTL = WDTPW + WDTHOLD;` ist nicht mehr nötig.
- `readAnalog` hat keine Parameter und liefert einen Wert von 0 bis 1023, welcher linear von der Spannung an Pin 1.5 abhängt.
- `setServoTo` stellt einen Servo ein. Dessen Steuereingang muss an Pin 1.2 angeschlossen sein. Als Argument wird die gewünschte Position des Servos als Zahl von 0 bis 100 übergeben. Zahlen außerhalb des Bereichs 0 bis 100 machen nichts kaputt, zumindest nicht sofort. Der Servo dreht aber nicht beliebig weit, sondern erreicht irgendwann seine mechanische Grenze.
- `waitMilliseconds` hält das Programm für die als Argument angegeben Zahl an Millisekunden an.

Schließen Sie den 1 kΩ-Widerstand, den Fotowiderstand und den Servo so an:

- Widerstand und Fotowiderstand bilden einen Spannungsteiler zwischen V_{CC} (positive Versorgungsspannung, hier etwa 3,6 V) und GND (Ground = Masse).

Das eine Bein des Fotowiderstands wird mit V_{CC} verbunden; das eine Bein des Widerstands mit GND. Die Mitte des Spannungsteilers wird mit Pin 1.5 verbunden.

- Der Servo wird mit Spannung versorgt (rot: V_{CC}, braun: Masse) und mit seiner Steuerleitung (orange) an Pin 1.2 angeschlossen. (Diese Farben sind je nach Hersteller leicht verschieden. Die dunkelste ist Masse, die hellste die Steuerleitung.)
- Befestigen Sie den Fotowiderstand am Hebel des Servos.

Programmieraufgaben:

1. Schreiben Sie in main.c eine Funktion mit dem Prototypen `int setServoAndRead(int percentage)`, die den Servo auf die in Prozent angegebene Position stellt, dann 300 Millisekunden wartet, um dem Servo Zeit zum Bewegen zu geben, und zum Schluss die Spannung misst und als Ergebnis zurückliefert. Schreiben Sie eine `main`-Funktion, die einige Aufrufe von `setServoAndRead` testet.
2. Lagern Sie die Funktion `setServoAndRead` in eine eigene .c- und eine eigene .h-Datei aus. Verwenden Sie diese .h-Datei in main.c. Benutzen Sie Include-Guards in der .h-Datei.
3. Ergänzen Sie diese .c- und diese .h-Datei um eine Funktion mit dem Prototypen `int findBest(int start, int end)`, die den Servo von der Position `start` bis zur Position `end` schwenkt und dann die Position mit der größten Lichtintensität zurückliefert. Diese Funktion verwendet die Funktion `setServoAndRead`.
4. Erweitern Sie das Programm so, dass es einmal von 0 bis 100 schwenkt, um die aktuell beste Position zu suchen, und dann laufend in kleinen Bereichen um die bisher beste Position schwenkt, um Änderungen des Lichts zu berücksichtigen. Versuchen Sie, das so zu programmieren, dass der Servo nie auf Positionen unter 0 oder über 100 gestellt wird.
5. Je nach verbleibender Zeit erweitern Sie das Programm zum Beispiel so:
   - Die Leuchtdioden auf dem Board zeigen an, in welche Richtung der Servo gerade fährt.
   - Das Programm führt regelmäßig wieder einen vollen Schwenk durch, um keine Änderungen zu verpassen, die weit weg von der aktuellen Position passieren.