

Bau und Programmierung eines Roboters für ROBOCUP JUNIOR RESCUE Teil 2.1: Programmieren

Jetzt geht's los mit dem Programmieren. Alles was du dafür brauchst gibt es auf der Seite, die in der Beschreibung hier unten verlinkt ist.

MPLABX starten,

Projekt öffnen, Lernprogramme U1, Slave 1, Klick, Doppelklick, Klick, Klick, Doppelklick

Oder so... Klick, Klick, Doppelklick

Die erste Lerneinheit: wir steuern das LCD an. Dies machen wir mit dem Slave vom PIC16F1519, deshalb der Name. Der Slave ist der kleinere Chip auf dem Board und ist ein PIC16F876A.

Es werden eingebunden die beiden Standardlibraries standard io und standard lib und natürlich der Compiler

Wir schauen uns als erstes den Header für die Initialisierung an, STRG[CTRL] und Klick

Wir müssen den Mikrocontroller so einstellen wie wir ihn brauchen

Die Interrupts an oder ausstellen

Die Pins am Chip als Ausgang oder Eingang definieren

Die Analogpins auswählen

Den Prescaler vom TMR0 aussuchen

Den I2C Bus einstellen

TMR0 spinnt manchmal, hiermit gehts immer

General Interrupt machen wir mal aus

Zurück zum Hauptprogramm

Jetzt kommt der Header für das LCD dran, STRG und Klick

Für die Funktion delay müssen wir dem Mikrocontroller sagen, mit welcher Taktfrequenz wir arbeiten

Um den Header universell zu gestalten stellen wir hier die Steuerungspins ein und den benutzen Port

Dann legen wir fest, an welchen Pins die Daten übertragen werden, wir benutzen nur vier

Dafür müssen wir auch die Funktionen obere und untere 4 bit anpassen

Für unser Board ist natürlich schon alles richtig eingestellt

Hier eine Auswahl von Funktionen, um mit dem LCD zu spielen

Wir brauchen nur den Befehl für die Cursorposition, die Adressen für die erste Zeile sind 0 bis 39 und 64 bis 103

Von diesen 40 Speicherstellen sind nur jeweils 16 sichtbar

Näher eingehen möchte ich auf die Funktionen Befehl an LCD, Daten an LCD und LCD String

Befehl an LCD

Um eine bessere Übersicht zu bekommen teile ich das Fenster

Hier festhalten und runterziehen

Die Funktion befindet sich hier

Um die Cursorposition festzulegen, an die das nächste Zeichen gesendet wird, müssen wir eine 128 plus einen Wert zwischen 0 und 103 senden, normalerweise begnügen wir uns mit den Werten 0 bis 15, für die erste Zeile und 64 bis 79 für die zweite

Diesen Wert übergeben wir an die Funktion Befehl an LCD

Zuerst wird solange gewartet, bis das LCD empfangsbereit ist, an der Stelle kann man auch einfach eine Millisekunde warten, aber wer hat schon soviel Zeit, dann wird dem LCD mitgeteilt, dass jetzt ein Befehl kommt und dann werden zuerst die oberen 4 Bit, dann die unteren gesendet

Fertig ist die Laube

Daten an LCD

Die Funktion ist im Prinzip genauso aufgebaut, wie Befehl an LCD

Wir müssen dem LCD nur jetzt mitteilen, dass ein Datum kommt, Datum ist der Singular von Daten

Die Daten, die wir senden, interpretiert das LCD als ASCII Code

Wie du sofort erkennen kannst musst du, um eine 1 auf das Display zu schreiben, eine 49 senden, um ein großes A zu schreiben eine 65, und so weiter

Na, der Rest der Funktion ist klar, ne?

Zurück zum Hauptprogramm

Erstmal stehen hier noch Einstellungen des Mikrocontrollers, darum kümmern wir uns evtl. später mal

Hier nochmal Frequenz angeben, ne

Wir haben an dem Pin PORTA 5 eine LED, die blinkt, wenn der Chip korrekt läuft

Dann kommen die Deklaration der Variablen und der benutzten Funktionen

Im Hauptprogramm rufen wir die Funktionen zur Initialisierung des Chips und des LCDs auf
Flags löschen
Interrupts ein oder ausschalten

Jetzt gehts endlich los:

Das Beispielprogramm benutzt zwei weitere Funktionen, die auf den eben erklärten basieren.

LCD_String überträgt eine Zeichenkette ans LCD, beginnend mit der angegebenen Adresse.

Dann wird das Display gelöscht und der Cursor auf die nullte Position gesetzt

jetzt betreten wir die Endlosschleife, die geht von da bis da

Die Funktion Befehl an LCD setzt den Cursor an die Spalte 4, also die fünfte Spalte, dann übertragen wir den Wert des Zählers Z

Die Funktion Zahl_an_LCD überträgt eine dreistellige Zahl an die angegebene Tabulatorstelle.

Das Display wird dabei in Viererblöcke aufgeteilt. Es können also insgesamt acht dreistellige Zahlen angezeigt werden.

Da wir keine negativen Werte messen, habe ich auf die Möglichkeit ein Minuszeichen anzuzeigen verzichtet.

Jetzt wird es spannend

Jetzt wollen wir das natürlich ausprobieren

Zuerst bespreche ich die Handhabung des PIC Kit 3

Zuerst stecken wir den Programmer mit dem USB-Anschluss an unseren Computer. Dann leuchten die LEDs ein bisschen rum.

Das andere Ende:

Auf dem Programmiergerät ist ein Pfeil, dieser Pin wird über das mitgelieferte Kabel mit dem Pin 1 des Slave auf unserem Board verbunden, bei mir ist das das blaue Kabel, bei euch ist das wahrscheinlich eine andere Farbe

Hää? Da sind aber sechs Anschlüsse am Programmer. Und am Board nur fünf. Genau. Der sechste Pin ist fürs Low-Voltage-Programming, brauchen wir nicht.

Jetzt müssen wir unserer IDE sagen, dass wir mit dem PIC Kit3 Programmieren wollen. Dazu gehen wir auf Projects, Rechtsklick auf unser Projekt, ganz unten Properties. Hier wird der Programmer angezeigt.

Anklicken und Apply. Fertig. Jetzt erscheinen hier oben drei Symbole farbig statt wie bisher grau

Jetzt Roboter anschalten. Achtung Roboter aufbocken, es kann sein, dass sich die Räder drehen.

Wenn wir hier drauf klicken erscheint das Output-Fenster und beim ersten Mal dauert es ganz schön lange, bis das Programm endlich aufgespielt ist. Achtung: Auf gar keinen Fall den Vorgang abbrechen, führt dazu, dass sich der Programmer dauerhaft aufhängt. Kann man nur mit Hilfe eines anderen Programmers beheben!!!

Ich persönlich finde den Brenner von Sprut besser, als den PIC Kit3. Auch wenn er nicht komplett alle Chips programmieren kann. Die gängigen PICs kann er allemal. Der Nachteil ist, dass du noch mehr löten musst.

Oder du bestellst ihn bei mir, zusammen mit der Hauptplatine und dem Motortreiber. Lieferzeit nicht unter 8 Wochen. Ich rate dir aber die Platinen selbst zu löten, denn mit den professionell hergestellten Platinen, die ich dir zum Selbstkostenpreis überlasse, ist das Löten wirklich nicht schwer. Das machen bei mir schon die Fünftklässler. Den PIC den du dafür brauchst kommt vorprogrammiert zusammen mit den Platinen.

Auf der Seite von Sprut findest du alle notwendigen Infos.

Jetzt zum Programmieren

Du lädst dir dieses Programm von der Homepage runter.

Für Windows 10 brauchst du auch noch dieses Programm Brenner in den Computer stecken, und Lib usb win 32 installieren

Zuerst müssen wir das Programm kompilieren. Klick,

Dann den Brenner mit dem USB-Kabel an den Computer anschliessen

Brennerprogramm starten

Den Brenner über das Programmierkabel mit dem PIC verbinden. Pin 1, Pin1. Roboter ist aus.

Dann klickst du auf Identify PIC. Es sollte jetzt der 16F876A angezeigt werden, danach wählst du das Hex-File aus, dass du auf den PIC schreiben willst. Der Pfad wird hier im Output-Fenster angezeigt. Dann auf

write Hex-File und warten bis das grüne O.K. flashed

Jetzt erst den Roboter einschalten.

Die LED sollte jetzt blinken und das LCD zeigt jetzt dieses an

Ja, was denn jetzt?

Das kriegst du schon selbst raus