

Mikrocontroller: Übung 2

Thema: 1. Schritte mit Assembler und C

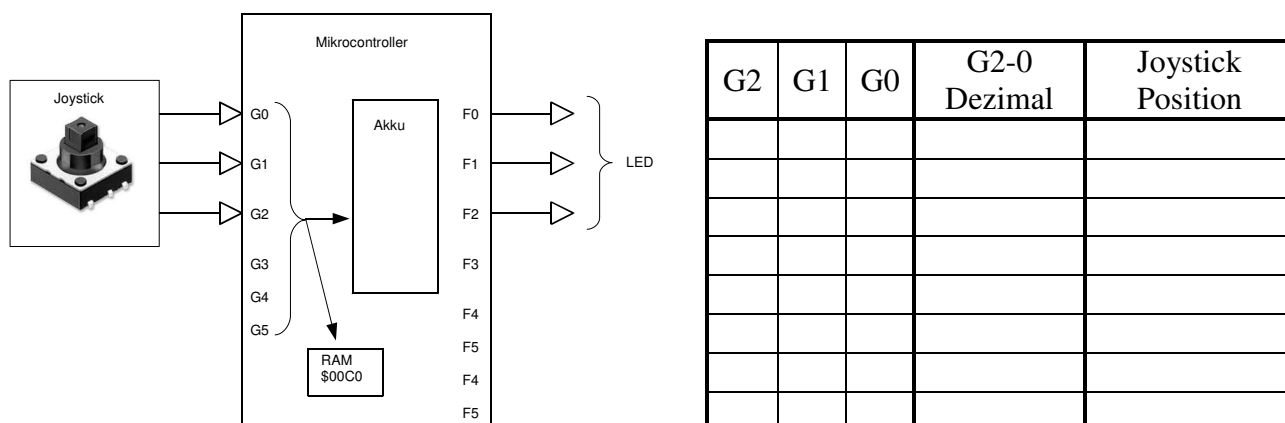
Arbeitsmodalität: Einzelarbeit

Lernziele dieser Übung

Sie setzen zum ersten Mal die Entwicklungsumgebung im MC-Labor ein und erkennen die Unterschiede zwischen der Programmierung in Maschinencode, Assembler und C. An Hand eines einfachen Beispiels verstehen Sie Aufbau und Funktionsweise der Ports unseres Mikrocontrollers.

Beschreibung der Grundaufgabe

Am Port G liege durch den Joystick 3-bit Wert an. Dieser Wert ist direkt auf Port G und in das interne RAM auf Adresse \$00C0 zu kopieren. Jede Änderung des 3-bit Wertes soll nachgeführt werden. Füllen Sie Anschliessend die Wahrheitstabelle aus.



1. Assembler-Lösung

- 1.1 Ändern Sie ein auf dem Template „Ueb02_PortIO_ASM“ basierendes Assembler-Projekt so ab, dass es die gewünschte Funktion realisiert. Verwenden Sie dabei vorerst die symbolischen Namen der internen Steuerregister. Assemblieren Sie das File und kontrollieren Sie die Funktion auf dem MC CAR.
- 1.2 Ersetzen Sie im Programm jetzt die vordefinierten symbolischen Namen durch absolute Adressen (siehe „MC9S08JM60A.inc“ u. „HSLU_HCS08_Programing_Guide.pdf“) und prüfen Sie erneut die Funktion auf MC CAR.
- 1.3 Überprüfen Sie jetzt die Funktion ihres Programmes im CW Debugger, in dem sie die Befehle **go**, **stepi**, **stop**, **mem**, **reg**, **bp** in der "Debugger Shell" verwenden.
- 1.4 Im Memory Fenster des Debuggers kontrollieren Sie die transferierten Werte ("Refresh while Running" wählen). Ab welcher Adresse beginnt Ihr Anwenderprogramm?

2. C-Lösung

Führen Sie Aufgabe 1 in der Hochsprache C aus (Template „Ueb02_PortIO_C“). Überprüfen Sie im Debugger Disassembly-Fenster wie die C Anweisungen durch den Compiler übersetzt wurden. Was fällt Ihnen auf?