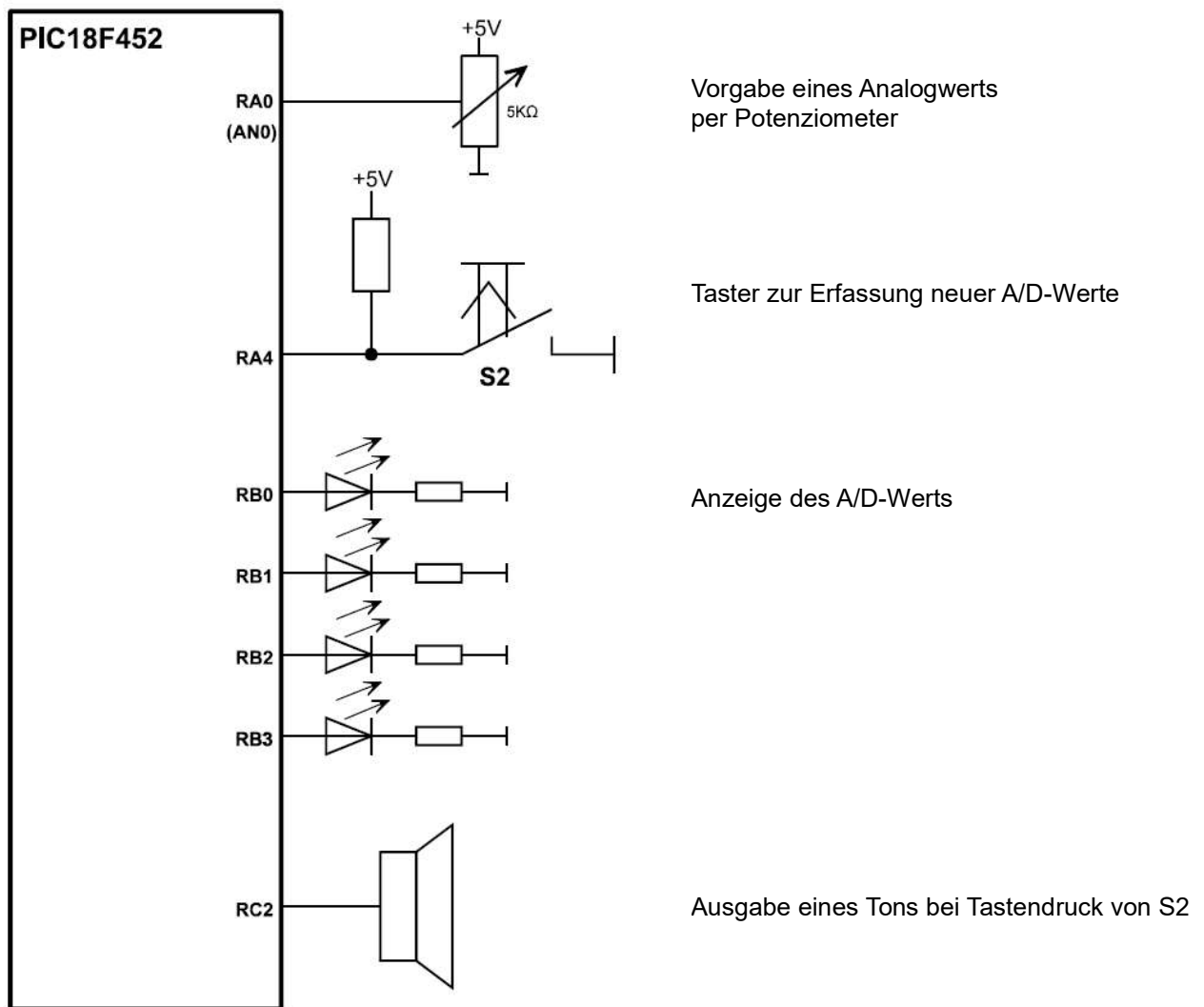


Erfassung eines Analogwerts und Ausgabe auf LEDs in PIC18-Assembler

Die folgende Schaltungsanordnung mit dem Mikrocontroller 18F452 ist gegeben:



Aufgabe 1 Programmwurf

Entwickeln Sie ein Programm in PIC18-Assembler, das folgende Aufgaben erfüllt:

- Abfrage des Tasters S2 am Port RA4 im Polling-Verfahren. Solange der Taster nicht gedrückt ist, soll keine Aktion erfolgen. Wird der Taster betätigt, soll der 10-Bit A/D-Umsetzer zum Lesen des Analogeingangs AN0 gestartet und die obersten 4 Bit des Wandlungsergebnisses an den LED's RB<3:0> angezeigt werden.
- Anschließend soll der Ausgang RC2 des Lautsprechers invertiert werden. Dies sorgt innerhalb der Hauptprogrammschleife dafür, dass eine Rechteckspannung ausgegeben wird. Damit deren Frequenz nicht zu hoch ist, soll eine einfache Zeitschleife in einem Unterprogramm zur Verlängerung der Periodendauer aufgerufen werden.
- Dieser ganze Vorgang soll als Endlosschleife wiederholt werden.

Für den A/D-Umsetzer sind folgende Einstellungen vorzunehmen:

A/D Conversion Clock: $F_{osc}/32$
 Voltage References: $V_{REF+} = V_{DD}$ $V_{REF-} = V_{SS}$

Verwenden Sie das vorgegebene Projektverzeichnis „Laborversuche\PIC1ad“ und die darin enthaltene Quelltextvorlage „PIC1ad.asm“. Achten Sie darauf, alles (auch die Projektdateien) in diesem **einen Verzeichnis** zu speichern.

Aufgabe 2 Debuggen per Simulation (zu Hause, Vorbereitung!)

Testen Sie das Programm mit Debugger > Select Tool > MPLAB SIM.

Verwenden Sie ein Watch-Fenster, um die Inhalte von Registern und Speichervariablen zu verfolgen.

Binden Sie einen vorgegebenen Stimulus für die simulierten Eingangssignale RA4 und ADRES ein mit
Debugger > Stimulus > Open Workbook „Stimulus ADRESL pic1ad.sbs“

Um in der Simulation dieselbe Taktfrequenz wie auf der Hardware (Quarz mit 4MHz) zu haben, tragen Sie in
>Debugger >Settings
[Osc / Trace] folgende Frequenz ein:
Processor Frequency: 20 MHz (Standardvorgabe) ⇒ **4 MHz** ⇒ dies ist **Fosc** in der Simulation

Öffnen Sie die Textdatei „Stimulus ADRESL pic1ad.txt“ in einem Editor, um die vorgegebenen AD-Werte zu sehen.
Jede Zeile dieser Datei enthält einen Hex-Wert der beim Abschluss der AD-Umsetzung in das AD-Ergebnisregisters ADRESL (AD Result Low) eingetragen wird - bei Überlauf auch in ADRESH. Wenn das AD-Ergebnisformat „Linksbündig“ gewählt wurde, erscheint der Wert bei der Umsetzung entsprechend nach links geschoben.

Verfolgen Sie bei schrittweisem Debuggen [F7] bzw. durch geeignetes Setzen von Haltepunkten und >Run [F9] die Inhalte der Register. Wenn Sie im Logic Analyzer die für Sie interessanten Portanschlüsse eintragen, können Sie den Zeitverlauf ihrer Pegel beobachten.

Aufgabe 3 Debuggen mit Hardware (im Labor)

Testen in gleicher Weise im Debug-Betrieb das Programm auf der Hardware.
Ihr Zielsystem ist über den Debugger ICD3 über USB mit dem PC verbunden.

Aufgabe 4 Programmlauf auf der Hardware ohne PC

Um Ihr Programm ohne Anbindung zum PC autark auf dem Zielsystem als Embedded-Anwendung laufen lassen zu können, wird der In-Circuit-Debugger ICD3 in einem anderen Modus betrieben: Statt als *Debugger* wird er als *Programmer* ausgewählt. Ihr Programm wird dann ohne Debug-Erweiterung in den Flash-ROM Ihres Mikrocontrollers übertragen, wodurch es durch Reset des Systems (auch Power-On-Reset) gestartet wird.

Führen Sie dies durch mit Programmer > Select Programmer > MPLAB ICD3
und erneutes LOAD (Flashen).

Sobald Sie die Nabelschnur zum In-Circuit-Debugger trennen, wird Ihr Zielsystem rückgesetzt und Ihr Programm startet. Dasselbe geschieht, wenn Sie die Versorgungsspannung neu einschalten oder die Reset-Taste drücken.