Dokumentation PIC Simulator Rechnerarchitektur

Aaron Moser

Hochschule Offenburg

Angewandte Informatik

Aufteilung:

Allgemeines:

• Grundsätzliche Arbeitsweise eines Simulators.

• Vor- und Nachteile einer Simulation.

• Programmoberfläche und deren Handhabung.

Realisation:

• Beschreibung des Grundkonzepts

• Beschreibung der Gliederung

• Programmstruktur, Ablaufdiagramme, Variablen usw.

• Welche Programmiersprache wurde gewählt? Warum? (falls Wahlfreiheit besteht)

• tiefer gehende Beschreibung der Funktionen an Hand ausgewählter Beispiele

( BTFSx, CALL, MOVF, RRF, SUBWF, DECFSZ, XORLW ). Diese und ggf.

weitere Befehle anhand von kurzen Programmsequenzen und Ablaufdiagrammen

erläutern.

• Realisierung der Flags und deren Wirkungsmechanismen.

• Wie wurden Interrupts implementiert? (Auszug aus dem Listing)

Diagramme und Beschreibung der Interruptfunktion.

• Wie wurde die Funktion des TRIS-Registers realisiert? (Latchfunktion?)

• Hardwareansteuerung (falls implementiert)

• Diagramm und Beschreibung mittels State-Machine z.B. für EEPROM

Zusammenfassung:

• · Wie weit konnten die Funktionen des Bausteins per Software nachgebildet wer-

den?

• · Fazit, persönliche Erfahrung und Erkenntnis. Was passierte während der Entwick-

lung des Projektes? Welche Probleme tauchten auf und wie wurden Sie gelöst. Ver-

meiden Sie dabei negative Formulierungen. Was würde ich anders machen, wenn

ich das Projekt nochmals realisieren müsste? (Umfang des Fazits ca. ¾ bis 1 Seite

oder 10 % des Gesamtumfangs)

Divide and Conquer bei Aufgabeneinteilung(MVC Architektur)

Welche Klassen hat mein Projekt? Aufgaben?

Wie habe ich das Projekt gemanaged (GitHub Repository)

Welche Programme verwendet? (VSC) warum?

Kernel Thread oder Benutzer Thread?