
Rolladensteuerung

Projektdokumentation

Für die Prüfung zum

Bachelor of Science

Des Studiengangs **Informatik**
Studienrichtung **Angewandte Informatik**

An der

Dualen Hochschule Baden-Württemberg Karlsruhe

Von

Leon Sauer (1543553)
Christopher Klammt (7849985)
Florian Jochem (3568778)

Abgabedatum	29. Juni 2018
Kurs	TINF16B2
Vorlesung Dozent	Systemnahe Programmierung Prof. Dr. Ralph Lausen

Erklärung

(gemäß §5(3) der „Studien- und Prüfungsordnung DHBW Technik“ vom 29.09.2015)

Wir versichern hiermit, dass unsere Projektdokumentation mit dem Thema „Rollladensteuerung“ selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt wurden.

Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Ort, Datum

Unterschrift

Sperrvermerk

Der Inhalt dieser Dokumentation darf weder als Ganzes noch in Auszügen Personen außerhalb des Prüfungsprozesses und des Evaluationsverfahrens zugänglich gemacht werden, sofern keine anders lautende Genehmigung der Ausbildungsstätte vorliegt.

Vorwort

Diese Projektdokumentation ist im Rahmen der Vorlesung **Systemnahe Programmierung** erarbeitet worden.

Die in der Dokumentation behandelten Themen, Konzepte sowie Implementierungen basieren auf den in der Vorlesung behandelten Inhalten.

Soweit möglich werden Bilder und Skizzen direkt in den Text eingebunden, um dem Leser so ein besseres Verständnis zu ermöglichen. Weitere Quelltextauszüge, Bilder und Skizzen finden sich im Anhang.

Inhaltsverzeichnis

Inhaltsverzeichnis	4
Abbildungsverzeichnis	5
Abkürzungsverzeichnis	5
Einleitung	6
Motivation	6
Aufgabenstellung	6
Grundlagen	7
Assembler	7
Der 8051 Mikrocomputer	7
Entwicklungsumgebung MCU-8051 IDE	8
Konzept	9
Analyse	9
Programmentwurf	9
Implementierung	11
Zusammenfassung	12
Anhang	13

Abbildungsverzeichnis

Abb. 1:	MCU-8051 IDE — Übersicht	8
Abb. 2:	MCU-8051 IDE — Multiplex LED	8

Abkürzungsverzeichnis

Einleitung

Motivation

Ziel des Projekts ist es eine Rollladensteuerung zu entwickeln. Hierbei soll es möglich sein, zunächst mit Hilfe von 2 „Buttons“ die Rolllade hoch- oder runterfahren lassen zu können. Wünschenswert ist hier ein Pausieren zwischen Hoch- und Herunterfahren.

Erweitert werden soll die Anwendung um eine Automatik, sodass die Rolllade zu einer festgesetzten Uhrzeit automatisch hoch- und zu einer anderen Uhrzeit runterfährt.

Falls genug Zeit verbleibt soll zudem die Möglichkeit implementiert werden, die Uhrzeit, zu der die Rolllade sich automatisch bewegt, selber setzen zu können.

Aufgabenstellung

Aus der vorangegangenen Darstellung der Motivation ergeben sich drei große Aufgabenstellungen, welche wie folgt umrissen werden können.

1. Rollladensteuerung mit zwei Schaltern
 - ❖ 2 Schalter darstellen
 - ❖ Display für Anzeige, ob Rolllade offen oder geschlossen ist sowie Uhrzeit
 - ❖ Inputhandling
2. Erweiterung bestehender Anwendung um Automatik
 - ❖ Automatik durch Timer oder Zeitauslesung
3. Manipulation Automatik durch Endanwender
 - ❖ Werte für Automatik-Timer zugänglich machen und dynamisch gestalten

Diese Aufgaben sind entsprechend der Nummerierung anzugehen, um so eine möglichst angenehme Umsetzung des Projekts gewährleisten zu können.

Zur Umsetzung wird die Entwicklungsumgebung **MCU-8051 IDE** (s.S.8) verwendet.

Grundlagen

Assembler

Unter einem Assembler versteht man im Allgemeinen ein Computerprogramm, das Assemblersprache in Maschinensprache umwandelt.

Grundsätzliche Aufgaben eines Compilers belaufen sich also u.a. auf die Verwaltung von Adressen für Befehle oder Daten, die Verwaltung von Konstanten, die Umsetzung von Befehlsmnemonics sowie Datenmnemonics in deren binäre Repräsentation, das Ignorieren von Kommentaren, das Inkludieren weiterer Dateien, die Definition und Anwendung von Makros als auch das Übersetzen von Maschinencode, Objektdateien bzw. grundlegenden Übersetzungs-Listen.

Der 8051 Mikrocomputer

Der 8051 Mikrocontroller ist der Produktfamilie *MCS-51* von Intel zuzuordnen und ist trotz seiner Ersterscheinung in den 1980er Jahren bis heute einer der beliebtesten Mikrocontroller. Hierbei handelt es sich um einen 8-bit Controller in drei unterschiedlichen Größen: **Short**, **Standard**, **Extended**.

Während *Short* und *Standard* häufig als DIP zur Verfügung stehen und problemlos „Drop-In Kompatibilität“ unterstützen, basieren die *Extended*-Modelle auf einem anderen Formfaktor und liefern keine Unterstützung für „Drop-In“. Obwohl alle drei Modelle individuelle, spezielle Features liefern, teilen sie viele Eigenschaften und können zudem in 8051 Assembly „beschrieben“ werden.

Wesentliche Merkmale, die dem 8051 zu Popularität verholfen haben, sind:

- 128 nutzerdefinierte „Software Flags“
- 8-bit Datenbus, 16-bit Adressbus
- 16-bit Timer
- 3 interne sowie 2 externe Interrupts
- 4 8-bit Ports
- 128 bytes RAM

Mikrocontroller der Familie MCS-51 nehmen aktuellen Schätzungen zu Folge 50% des Einsatzes von Mikrocontrollern in Embedded-Systems ein und sind in einer breiten Palette von Anwendungsgebieten vorzufinden, so u.a. in der Kontroll-, Automobil- und Roboterindustrie.

Entwicklungsumgebung MCU-8051 IDE

Wie bereits in dem Kapitel *Einleitung* angemerkt, wird zur Umsetzung des Projekts die MCU-8051 IDE verwendet.

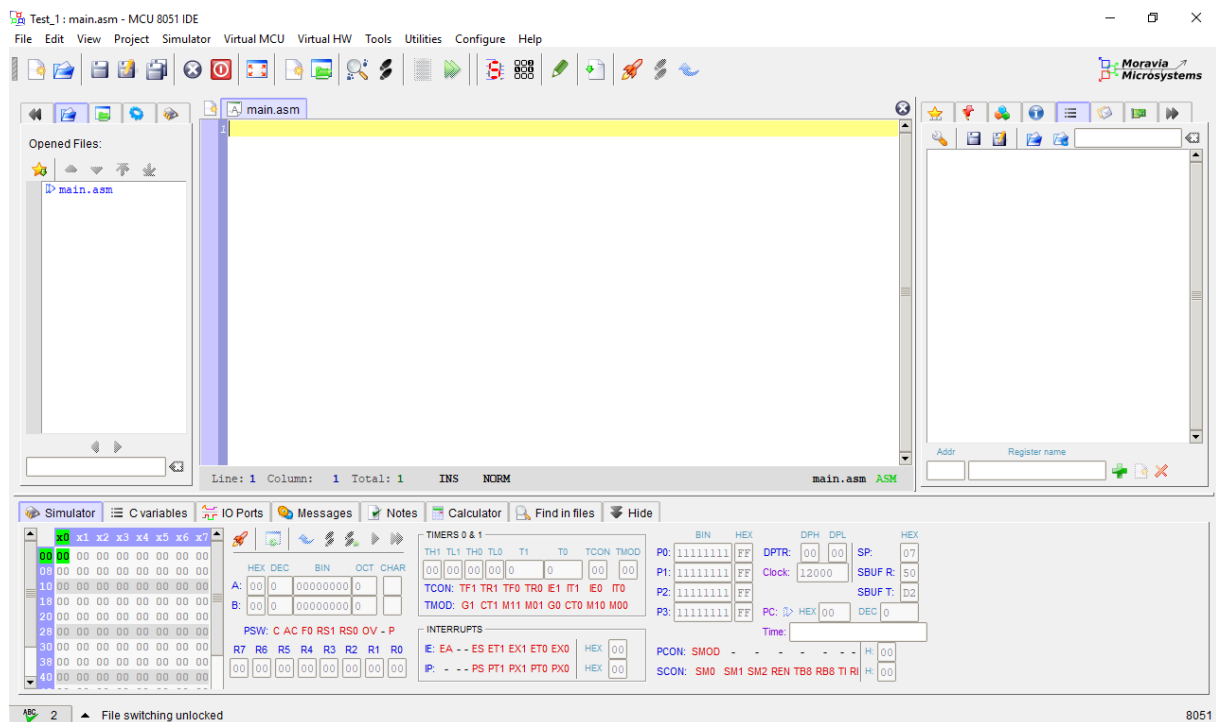


Abb.1: MCU-8051 IDE — Übersicht

Neben der Möglichkeit speziell für den 8051 Mikrocontroller Programme zu schreiben, liefert die Entwicklungsumgebung zusätzlich die Möglichkeit Hardware zu simulieren und stellt dem Entwickler so u.a. verschiedene Displays zur Verfügung.

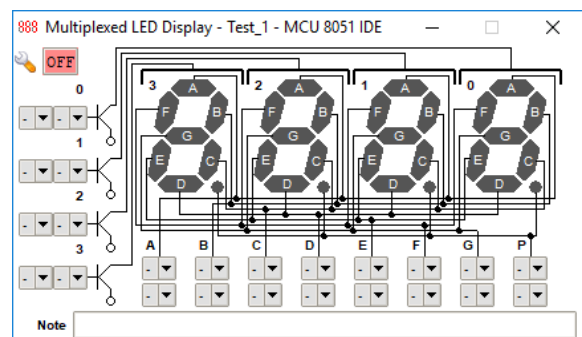


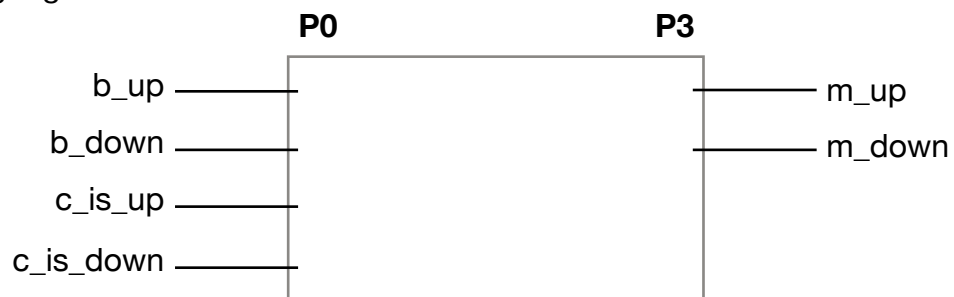
Abb.2: MCU-8051 IDE — Multiplex LED

Konzept

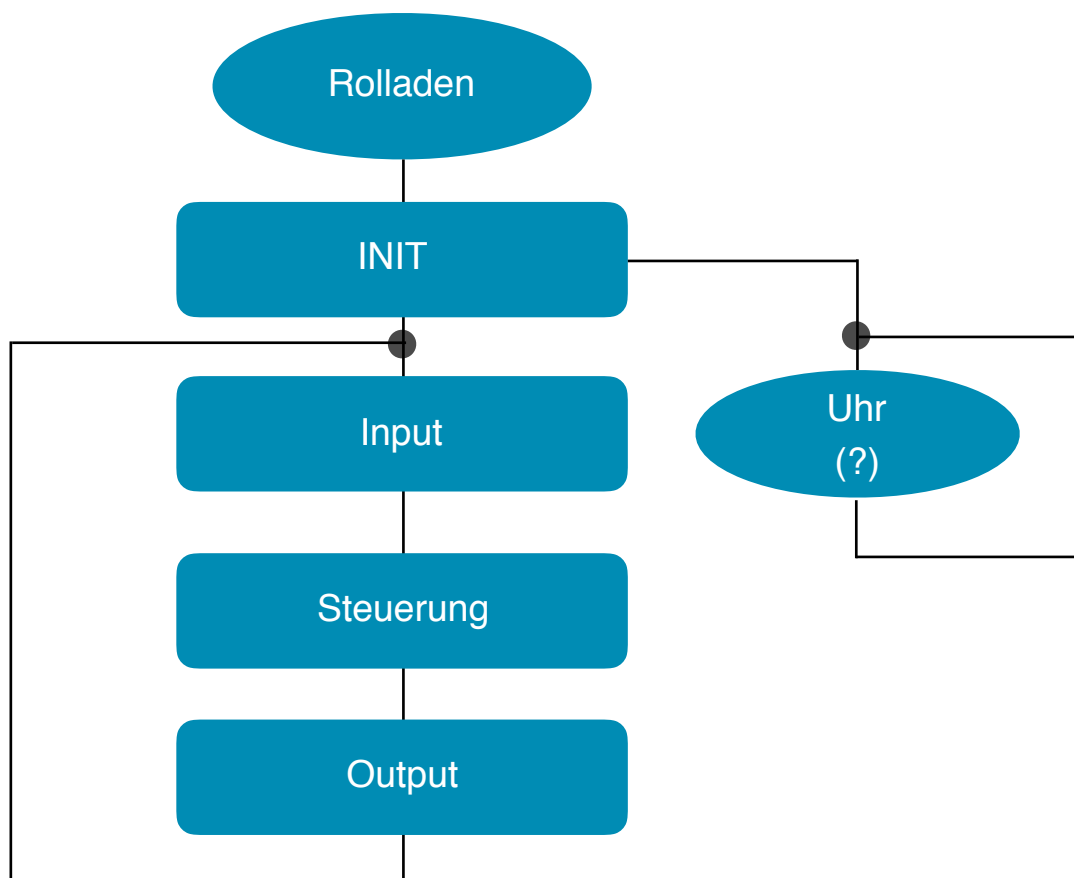
Analyse

Programmentwurf

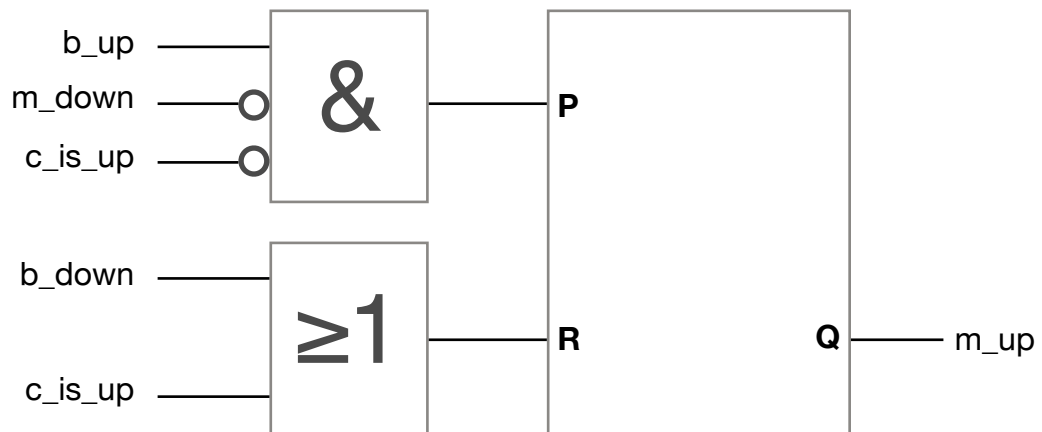
❖ Pinbelegung



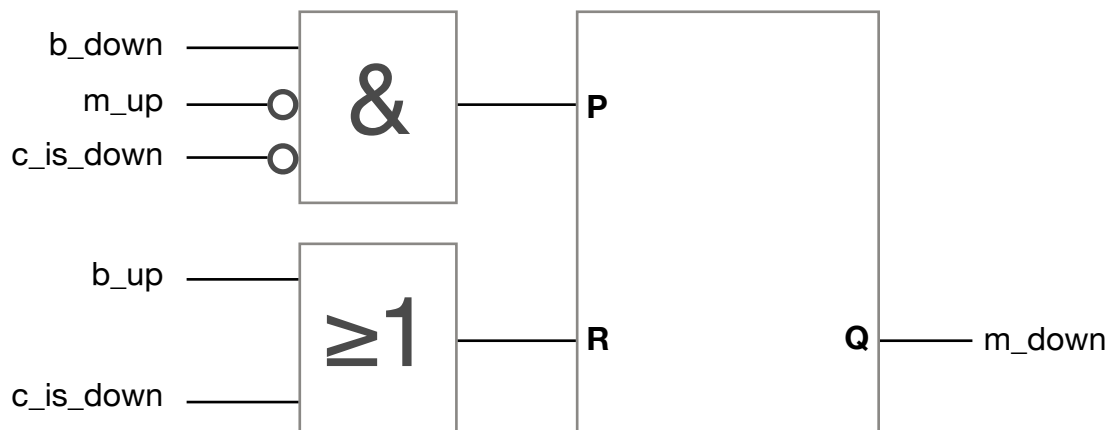
❖ Programmablauf



❖ Button Interrupt (1)



❖ Button Interrupt (2)



Implementierung

Zusammenfassung

Anhang