

Angewandte Informatik

Semester 4

Dokumentation

Simulator eines Mikrocontrollers PIC16F84

Name: Aaron Moser

Matrikelnummer: 187648

E-Mail: aaronmoser@t-online.de

Prüfungsdatum: 05.05.2022

Betreuer: Dipl.-Ing. Stefan Lehmann

Inhaltsverzeichnis

[Einleitung 16](#_Toc103120973)

[Mikrocontroller PIC16F84 16](#_Toc103120974)

[Graphische Benutzeroberfläche des Simulators 17](#_Toc103120975)

[Hauptmenü 17](#_Toc103120976)

[LST Anzeigefeld 18](#_Toc103120977)

[SFR und W Registertabelle 18](#_Toc103120978)

[STATUS, OPTION und INTCON Registertabelle 19](#_Toc103120979)

[RAM (Datenspeicher) Tabelle 19](#_Toc103120980)

[PORT A und PORT B Feld 19](#_Toc103120981)

[STACK Tabelle 20](#_Toc103120982)

[Laufzeitzähler Feld und Quarzfrequenz Drop-Down 20](#_Toc103120983)

[Simulatorsteuerungsfeld 21](#_Toc103120984)

[Implementierung 22](#_Toc103120985)

[Ablaufdiagramme 23](#_Toc103120986)

[main – Methode 23](#_Toc103120987)

[Klassendiagramme 25](#_Toc103120988)

[Klasse MyControlModel 25](#_Toc103120989)

[Klasse MyControlView 26](#_Toc103120990)

[Klasse Alu 27](#_Toc103120991)

[Klasse Eeprom 27](#_Toc103120992)

[Klasse EepromThread 28](#_Toc103120993)

[Klasse InstructionDecoder 28](#_Toc103120994)

[Klasse Pic 29](#_Toc103120995)

[29](#_Toc103120996)

[Klasse ProgramMemory 30](#_Toc103120997)

[Klasse Ram 31](#_Toc103120998)

[32](#_Toc103120999)

[33](#_Toc103121000)

[Klasse Stack 34](#_Toc103121001)

[Klasse Time 34](#_Toc103121002)

[Klasse ReadProgramFile 35](#_Toc103121003)

[Klasse MyModel 35](#_Toc103121004)

[Klasse MyModelData 36](#_Toc103121005)

[Klasse ProgramStepInformation 36](#_Toc103121006)

[Klasse Main 36](#_Toc103121007)

[Klasse GUIAbout 36](#_Toc103121008)

[Klasse GUIHelp 37](#_Toc103121009)

[Klasse GUIMainFrame 37](#_Toc103121010)

[37](#_Toc103121011)

[Klasse GUIMCMenu 37](#_Toc103121012)

[37](#_Toc103121013)

[Klasse GUIMenuBar 38](#_Toc103121014)

[Klasse GUIPorts 39](#_Toc103121015)

[39](#_Toc103121016)

[Klasse GUIProgramMemory 40](#_Toc103121017)

[Klasse GUIRamTable 40](#_Toc103121018)

[Klasse GUIRegister 41](#_Toc103121019)

[Klasse GUIRegistersDetailed 42](#_Toc103121020)

[Klasse GUIStack 42](#_Toc103121021)

[Klasse GUITestFileTable 43](#_Toc103121022)

[Klasse GUITime 44](#_Toc103121023)

[Klasse IMyView 44](#_Toc103121024)

[Klasse MyColors 44](#_Toc103121025)

[Klasse MyView 45](#_Toc103121026)

[Abbildung 1 (Q1) PIC16F84 5](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120930)

[Abbildung 2 Übersicht Benutzeroberfläche 6](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120931)

[Abbildung 3 LST 7](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120932)

[Abbildung 4 SFR und W 7](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120933)

[Abbildung 5 STATUS, OPTION und INTCON 8](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120934)

[Abbildung 6 RAM 8](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120935)

[Abbildung 7 Port A und Port B 8](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120936)

[Abbildung 8 Stack 9](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120937)

[Abbildung 9 Laufzeitzähler 9](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120938)

[Abbildung 10 Simulatorsteuerung 10](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120939)

[Abbildung 11 Ablauf main 12](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120940)

[Abbildung 12 MyControlModel 14](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120941)

[Abbildung 13 MyControlView 15](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120942)

[Abbildung 14 Alu 16](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120943)

[Abbildung 15 Eeprom 16](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120944)

[Abbildung 16 EepromThread 17](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120945)

[Abbildung 17 InstructionDecoder 17](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120946)

[Abbildung 18 ProgramMemory 19](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120947)

[Abbildung 19 Pic 19](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120948)

[Abbildung 20 Ram 22](#_Toc103120949)

[Abbildung 21 Time 23](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120950)

[Abbildung 22 Stack 23](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120951)

[Abbildung 23 ReadProgramFile 24](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120952)

[Abbildung 24 MyModel 24](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120953)

[Abbildung 25 MyModelData 25](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120954)

[Abbildung 26 Main 25](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120955)

[Abbildung 27 ProgramStepInformation 25](file:///C:\Users\Windows%2010\Desktop\Studium\Repos_Computerarchitecture\Rechnerarchitektur_Simulator_PIC\documentation\WORD\Dokumentation_Final.docx#_Toc103120956)

[Abbildung 28 GUIAbout 25](#_Toc103120957)

[Abbildung 29 GUIHelp 26](#_Toc103120958)

[Abbildung 30 GUIMainFrame 26](#_Toc103120959)

[Abbildung 31 GUIMCMenu 26](#_Toc103120960)

[Abbildung 32 GUIMenuBar 27](#_Toc103120961)

[Abbildung 33 GUIPorts 29](#_Toc103120962)

[Abbildung 34 GUIProgramMemory 29](#_Toc103120963)

[Abbildung 35 GUIRamTable 29](#_Toc103120964)

[Abbildung 36 GUIRegister 30](#_Toc103120965)

[Abbildung 37 GUIRegistersDetailed 31](#_Toc103120966)

[Abbildung 38 GUIStack 31](#_Toc103120967)

[Abbildung 39 GUITestFileTable 32](#_Toc103120968)

[Abbildung 40 GUITime 33](#_Toc103120969)

[Abbildung 41 IMyView 33](#_Toc103120970)

[Abbildung 42 MyColors 33](#_Toc103120971)

[Abbildung 43 MyView 34](#_Toc103120972)

# Einleitung

Die Aufgabe der Projektarbeit im Fach Rechnerarchitektur ist es, den Simulator eines Mikrocontrollers zu entwickeln, der dessen Funktionen so gut wie möglich nachbildet.

Ein Simulator ist ein Modell eines real existierenden Objekts, welches nachgebildet werden soll. Er ist nicht als exaktes Abbild zu verstehen, sondern kann je nach Anforderung und Aufgabe vereinfacht werden.

Dementsprechend bildet der hier dokumentierte Simulator des Mikrocontrollers PIC16F84 diesen nur zu Teilen ab. Manche Teile wurden zur Vereinfachung nicht implementiert. Reale Eigenschaften wie verändertes Verhalten bei erhöhten Bauteiltemperaturen oder andere physikalische Eigenschaften werden nicht nachgebildet.

Der Simulator besteht aus einer graphischen Benutzeroberfläche (GUI), welche die internen und externen Zustände des Mikrocontrollers wie z.B. Ein- und Ausgangspins oder den Datenspeicher darstellt und einem im Hintergrund laufenden Programm, welches die interne logische Funktion des Mikrocontrollers nachbildet.

# Mikrocontroller PIC16F84

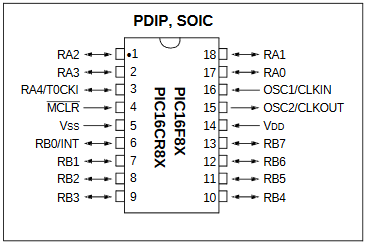
Der Mikrocontroller PIC16F84 wurde im Jahr 1998 von *Microchip Technology Inc.* entwickelt. Er gehört zur Produktfamilie der PIC16CXX. Die interne Architektur entspricht der Harvard-Architektur, bei der Daten- und Programmspeicher voneinander getrennt sind, dadurch entfällt das in der von-Neumann-Architektur typische Nadelöhr. Grundsätzlich umfässt die Größe des Befehlssatzes 35 Befehle, wobei beim Simulator wenige dieser Befehle nicht implementiert sind (CLRWDT, SLEEP, TRIS, OPTION). Der Programmspeicher kann mit einem Programm geladen werden, welches eine Reihe an Befehlen beinhaltet.

Abbildung (Q1) PIC16F84

# Graphische Benutzeroberfläche des Simulators

Abbildung Übersicht Benutzeroberfläche

Die graphische Benutzeroberfläche des Simulators besteht aus einem Hauptfenster mit einem Hauptmenü, mehreren Tabellen, 4 Schaltflächen und einem Drop-Down-Menü.

## Hauptmenü

Das Hauptmenü besteht aus 4 Punkten (Datei, Ansicht, Microcontroller und Hilfe), die wiederum in kleinere Punkte unterteilt sind.

Der Menüpunkt „Datei“ wird in die Punkte „Testdatei laden“, „Programmzustand laden“, „Programmzustand speichern“ und „Simulation beenden“ aufgeteilt. Wobei nur die Punkte „Testdatei laden“ und „Simulation beenden“ zurzeit implementiert sind, da die anderen Punkte für die Bewertung nicht relevant sind.

„Ansicht“ besteht aus dem Punkt „Thema“ über den zwischen den Anzeigemodi „Dunkler Modus“ und „Heller Modus“ gewählt werden kann.

Zusätzlich zu den vier Schaltflächen im Hauptfenster sind unter dem Hauptmenüpunkt „Microcontroller“ vier Punkte zum Steuern der Simulation (Programm starten, Programm pausieren, Programm zurücksetzen, Schritt für Schritt). Außerdem gibt es einen Unterpunkt „Bearbeitungsintervall“, über den die Geschwindigkeit der Programmausführung geändert werden kann um die Möglichkeit zu bieten, die Programmausführung besser nachzuvollziehen. Zur Auswahl stehen: „Sofort“, „1 Sekunde“ und „2 Sekunden“.

Der Punkt „Hilfe“ enthält die Unterpunkte „Sprache“, „Anleitung“ und „Ueber“. Wobei der Punkt „Anleitung“ nicht komplett implementiert ist. Unter „Sprache“ gibt es die Möglichkeit zwischen „Deutsch“ und „Englisch“ zu wählen.

## LST Anzeigefeld

Um den Simulator ein Programm simulieren zu lassen, muss eine nach bestimmten Regeln formatierte Datei mit .lst-Endung geladen werden. Um dies zu tun, klickt man im Hauptfenster des Simulators auf den Menüpunkt „Datei“ und anschließend auf „Testdatei laden“. Nun öffnet sich ein Dateiauswahlfenster, in dem man dann eine geeignete Datei auswählt.

Abbildung LST

Die Textdatei wird Zeile für Zeile in einer Tabellenform in das entsprechende Teilfenster(s. rote Markierung) geladen. Die Tabelle besteht aus drei Spalten, zum einen die Nummerierung, welche ab 1 aufsteigend erfolgt. In der zweiten Spalte befinden sich die entsprechenden Breakpoints, welche als Kästchen zum Anhaken implementiert sind. Falls sich eine Zeile um einen Kommentar handelt, so wird der zugehörige Breakpoint deaktiviert. Die dritte Spalte entspricht eins zu eins den Inhalten der geladenen .lst-Datei. Der als nächstes auszuführende Befehl wird durch eine türkise Färbung markiert.

## SFR und W Registertabelle

SFR (Special Function Register) ist ein Register mit Variablen, welche von der CPU (Central Processing Unit) und Peripheriegeräten genutzt werden um die Operationen des Mikroprozessors zu steuern.

Abbildung SFR und W

Das W-Register wird genutzt um Zwischenergebnisse zu speichern und ist flüchtig. Alle Werte werden in der Tabelle als Dezimalwerte angezeigt.

## STATUS, OPTION und INTCON Registertabelle

Die drei Teilregister des SF-Registers STATUS, OPTION und INTCON werden anhand ihrer einzelnen Bits beurteilt, weshalb sie in der Benutzeroberfläche explizit dargestellt werden, um die einzelnen Bits besser ablesen zu können. Eine 0 entspricht einem logischen False (clear) und eine 1 dem logischen True (set).

Abbildung STATUS, OPTION und INTCON

## RAM (Datenspeicher) Tabelle

Das RAM ist der Datenspeicher für Werte, die eine längere Zeit gespeichert werden sollen und später verwendet werden. Er besteht aus zwei Bänken, die allerdings in der GUI visuell nicht leicht voneinander unterschieden werden können. Jede Bank umfässt eine Speicherkapazität von 128 Byte. Wobei bei einem Byte der Wertebereich auf 0 – 255 begrenzt ist.

Abbildung RAM

## PORT A und PORT B Feld

Port A und B bestehen aus einem Register für die Ein- und Ausgänge und einem Register, welches bestimmt, ob die Ein-/Ausgänge als Ein- oder Ausgang definiert sind. Pin 4 von Port A wird auch „externer Zählereingang“ genannt, denn er kann dazu genutzt werden den Timer0 zu inkrementieren, falls das T0CS-Bit Pin 4 zugewiesen ist. Außerdem ist Pin 0 von Port B der „externe Interrupteingang“, der abhängig vom INTEDG-Bit bei einer steigenden oder einer fallenden Flanke einen Interrupt auslöst. Zusätzlich wird ein Interrupt ausgelöst, wenn an einem der vier Pins 4 bis 7 von Port B eine Zustandsänderung stattfindet. Allerdings müssen noch die Bits GIE und INTE oder RBIE gesetzt sein.

Abbildung 7 Port A und Port B

## STACK Tabelle

Abbildung Stack

Der Stack besteht aus acht Einträgen, die hier initial -1 sind, da bei dieser Implementierung bei einer Operation, bei der ein Eintrag vom Stack geholt wird geprüft wird, ob der Stack leer ist und wenn ja, wird eine Exception geworfen. Ansonsten wird der Eintrag, auf den der Stackpointer zeigt, aus dem Stack geholt und in den Programmzähler geschrieben.

## Laufzeitzähler Feld und Quarzfrequenz Drop-Down

Abbildung Laufzeitzähler

Der Laufzeitzähler ist ein Aufwärtszähler im Millisekundenbereich, der ab 0 beginnt zu zählen. Der Wert, der hinzuaddiert wird, ist abhängig von der eingestellten Quarzfrequenz und der Zykluszeit eines Befehls. Die Möglichkeiten sind: 32kHz => 0.125ms, 100kHz => 0.04ms, 500kHz => 0.008ms, 1MHz => 0.004ms, 2MHz => 0.002ms, 4MHz => 0.001ms, 8MHz => 0.0005ms, 12MHz => 0.0003ms, 16MHz => 0.00025ms und 20MHz => 0.0002ms.

## Simulatorsteuerungsfeld

Abbildung Simulatorsteuerung

Das Simulatorsteuerungsfeld besteht aus 4 Schaltflächen. „Start“ startet die Programmausführung, wobei der Simulator Schritt für Schritt durch den Programmspeicher läuft, der die Befehle der vorher geladenen Testdatei enthält. Falls kein Breakpoint gesetzt ist, läuft das Programm weiter, bis es terminiert oder in einer Endlosschleife.

# Implementierung

Bei der Implementierung des Projekts wurde versucht die Programmstruktur nach der MVC (Model, View, Controller) – Architektur zu realisieren Dies ist zu einem gewissen Grad gelungen, wobei jedoch die Schnittstellen drei ConcurrentLinkedQueues sind. Über diese werden Änderungen an der GUI an das Backend gesendet und vom Backend Informationen an das Frontend.

# Ablaufdiagramme

## main – Methode

Abbildung Ablauf main

## MyModel run - Methode

Abbildung

# Klassendiagramme

## Klasse MyControlModel

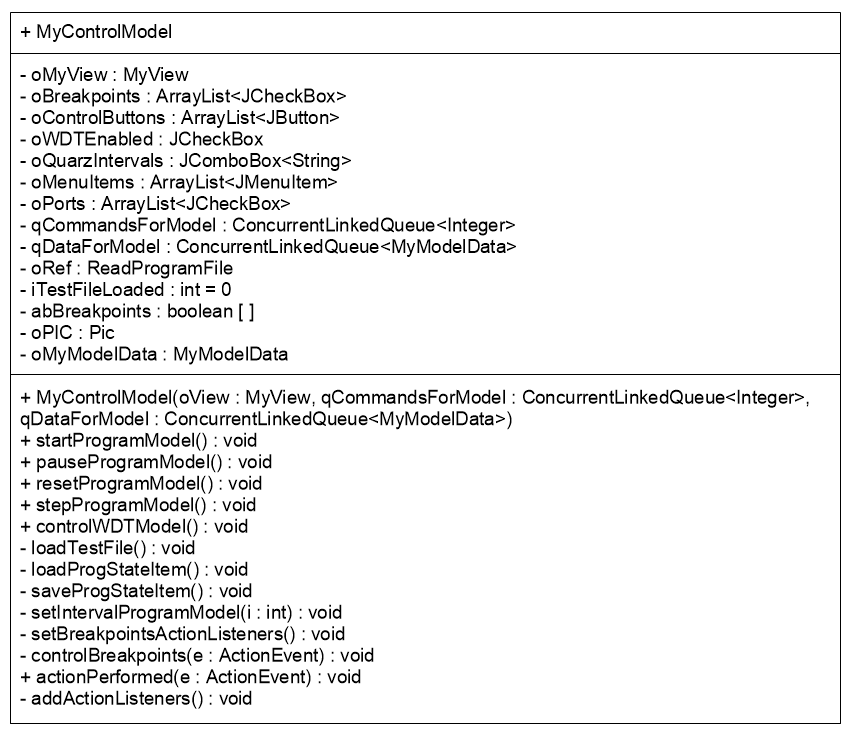


Abbildung MyControlModel

## Klasse MyControlView

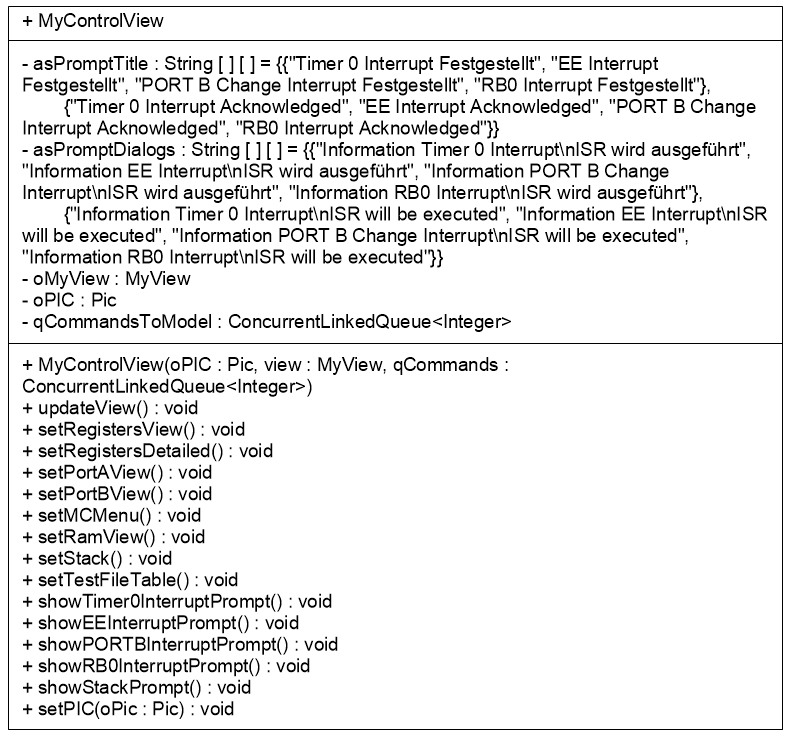


Abbildung MyControlView

## Klasse Alu

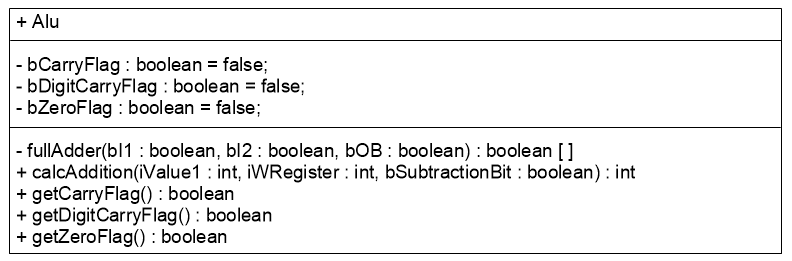


Abbildung Alu

## Klasse Eeprom

Abbildung Eeprom

## Klasse EepromThread

Abbildung EepromThread

## Klasse InstructionDecoder

Abbildung InstructionDecoder

## Klasse Pic

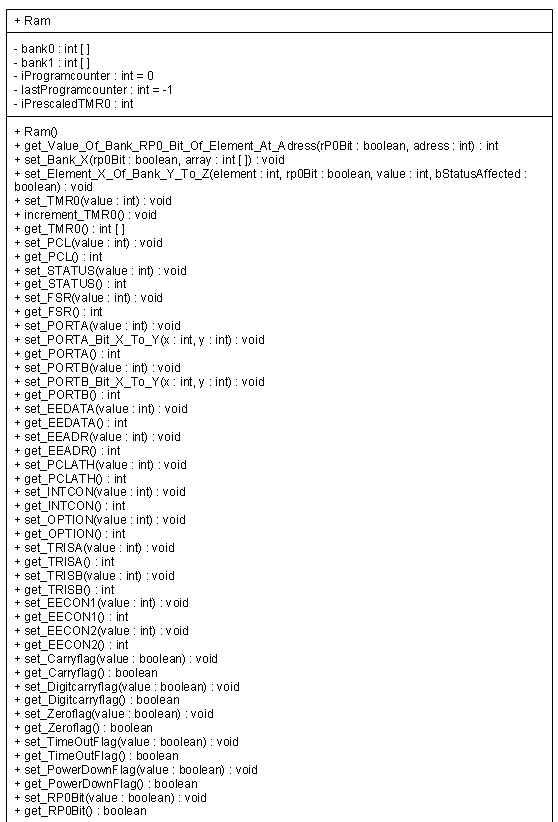
## 

## Klasse ProgramMemory

Abbildung ProgramMemory

Abbildung Pic

## Klasse Ram



## 

## 

Abbildung 21 Ram

## Klasse Stack

## Klasse Time

Abbildung Time

Abbildung Stack

## Klasse ReadProgramFile

Abbildung ReadProgramFile

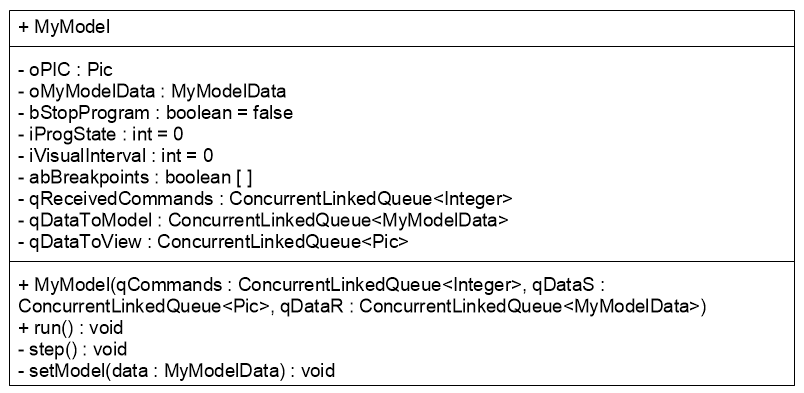
Klasse MyModel

Abbildung MyModel

## Klasse MyModelData

Abbildung MyModelData

## Klasse ProgramStepInformation

## Klasse Main

Abbildung Main

Abbildung ProgramStepInformation

## Klasse GUIAbout

Abbildung 29 GUIAbout

## Klasse GUIHelp

Abbildung 30 GUIHelp

## Klasse GUIMainFrame

## 

Abbildung GUIMainFrame

## Klasse GUIMCMenu

## 

Abbildung GUIMCMenu

## Klasse GUIMenuBar

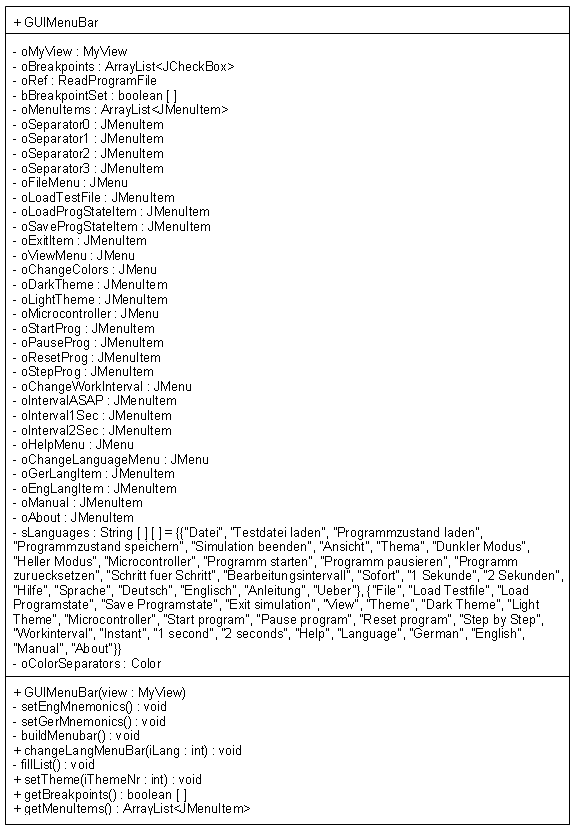


Abbildung 33 GUIMenuBar

## Klasse GUIPorts

## 

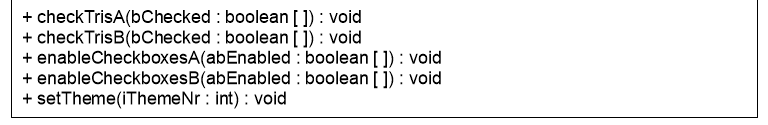


Abbildung 34 GUIPorts

## Klasse GUIProgramMemory

Abbildung 35 GUIProgramMemory

## Klasse GUIRamTable

Abbildung 36 GUIRamTable

## Klasse GUIRegister

Abbildung 37 GUIRegister

## Klasse GUIRegistersDetailed

Abbildung 38 GUIRegistersDetailed

## Klasse GUIStack

Abbildung 39 GUIStack

## Klasse GUITestFileTable

Abbildung 40 GUITestFileTable

## Klasse GUITime

Abbildung 41 GUITime

## Klasse IMyView

Abbildung 42 IMyView

## Klasse MyColors

Abbildung 43 MyColors

## Klasse MyView

Abbildung MyView

# Fazit