

PicoC-Compiler

Übersetzung einer Untermenge von C in den Befehlssatz der RETI-CPU

BACHELORARBEIT

Abgabedatum: 28th April 2022

Author:
Jürgen Mattheis

Gutachter:
Prof. Dr. Scholl

Betreuung:
M.Sc. Seufert

ERKLÄRUNG

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen/Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, bereits für eine andere Prüfung angefertigt wurde.

Danksagungen

Bevor der Inhalt dieser Schriftlichen Ausarbeitung der Bachelorarbeit anfängt, will ich einigen Personen noch meinen Dank aussprechen.

Wie mein Betreuer M.Sc. Tobias Seufert und wahrscheinlich auch mein Gutachter Prof. Dr. Scholl im Verlauf dieser Bachelorarbeit und des vorangegangenen Bachelorprojektes gemerkt haben, bin ich keine Person, die irgendwelche Dinge gerne so macht wie es üblich ist, ich schreibe meine Danksagung nicht auf eine bestimmte Weise nur weil sich irgendwann mal etabliert hat wie eine Danksagung üblicherweise aussieht. Ich halte nicht viel von künstlichen Floskeln, wie „mein aufrichtigster Dank“ oder „aus tiefstem Herzen“, sondern drücke Dinge nur so aus, wie ich sie auch wirklich meine.

Ich halte es eher so, dass wenn mir wirklich etwas am Dank gegenüber Personen liegt, ich mir wirklich den Aufwand mache einen Text zu schreiben in dem ich diesen zum Ausdruck bringe, im anderen Fall kann man sich bei mir auf die typischen Standardfloskeln einstellen.

An erster Stelle will ich zu meinem Betreuer M.Sc. Tobias Seufert kommen, der netterweise auch bereits die Betreuung meines Bachelorprojektes übernommen hatte. Wie auch während des Bachelorprojektes, haben wir uns auch bei den Meetings während der Bachelorarbeit hervorragend verstanden. Dabei ging die Freundlichkeit und Hingabe seitens Tobias weit über das hinaus, was man bereits als eine gute Betreuung bezeichnen würde.

Es gibt verschiedene Typen von Personen, es gibt Leute, die nur genauso viel tun, wie es die Anforderungen verlangen und nichts darüberhinaus tun, wenn es nicht einen eigenen Vorteil für sie hat und es gibt Personen, die sich für nichts zu Schade sind und dies aus einer Philanthropie oder Leidenschaft heraus tun auch wenn es für sie keine Vorteile hat. Tobias konnte ich während der langen Zeit, die er mein Bachelorprojekt und dann meine Bachelorarbeit betreut hat eindeutig als letzteren Typ Mensch einordnen.

Er war sich nie zu Schade für meine vielen Fragen während der Meetings, auch wenn ich meine Zeit ziemlich oft überzogen habe¹, er hat sich bei der Korrektur dieser Schriftlichen Ausarbeitung sogar die Mühe gemacht bei den einzelnen Problemstellen längere Textkommentare zu verfassen und das trotz dessen, dass meine Bachelorarbeit recht umfangreich zu lesen ist² und war sich nicht zu Schade die Rolle des Nachrichtenübersetzers zwischen mir und Prof. Dr. Scholl zu übernehmen. All dies war absolut keine Selbstverständlichkeit, vor allem wenn ich die Betreuung anderer Studenten, die ich kenne mit der vergleiche, die mir zu Teil wurde.

An den Kommentar zu meinem Betreuer Tobias will ich einen Kommentar zu meinem Gutachter Prof. Dr. Scholl anschließen. Ich hab während meines Bachelorprojekts und meiner Bachelorarbeit wahrscheinlich einen ziemlich sturren Eindruck dabei gemacht, bei der Weise, wie ich bestimmte Dinge umsetzen wollte und ich weiß es sehr zu schätzen, dass mir das durchgehen gelassen wurde. Es ist, wie ich die Universitätswelt als Student erlebe bei Arbeitsvorgaben keine Selbstverständlichkeit, dass dem Studenten überhaupt die Freiheit und das Vertrauen gegeben wird es auf seine eigenen Weise umzusetzen.

Vor allem da mein eigenes Vorgehen, vor allem Vorteile für mich hatte, da ich auf diese Weise am meisten über Compilerbau gelernt hab und eher Nachteile für Prof. Dr. Scholl, da mein eigenes Vorgehen entsprechend mehr Zeit brauchte und ich daher als Bachelorarbeit keinen dazu passenden RETI-Emulator mit Graphischer Anzeige implementieren konnte, da die restlichen Funktionalitäten des PicoC-Compilers noch implementiert

¹Wofür ich mich auch nochmal Entschuldigen will.

²Wobei er sich kein einziges Mal in geringster Weise entnervt darüber gezeigt hat.

werden mussten.

Glücklicherweise gibt es aber doch noch einen passenden RETI-Emulator, der den PicoC-Compiler über seine Kommandozeilenargumente aufruft, um ein PicoC-Programm visuell auf einer RETI-CPU ausgeführt zu sehen, für dessen Implementierung Michel Giehl sich zur Verfügung gestellt hat. Der RETI-Emulator von Michel Giehl ist unter [Link](#)³ zu finden. Letzten Endes steht den kommenden Studengenerationen glücklicherweise doch ein vollumfängliches Lerntool zur Verfügung, an dem sie die Funktionsweise einer RETI-CPU anhand der Ausführung eines kompilierten PicoC-Programmes beobachten könnten, wenn auch nicht auf dem ursprünglich geplanten Weg.

Eine Sache, die ich an meinem Gutachter Prof. Dr. Scholl absetits der Bachelorarbeit sehr schätze und mal ausgesprochen habe wollte, ist dass die Vorlesungen von Prof. Dr. Scholl in sich selbst abgeschlossen sind und es merklich ist, dass sich wirklich gründliche Gedanken darüber gemacht wurden wie der Inhalt aufzubereiten ist, damit es Verständlich beim Studenten ankommt und keine zum Verständniss notwendigen Informationen fehlen.

Mir hat die Implementierung des PicoC-Compilers tatsächlich ziemlich viel Spaß gemacht, da Compilerbau auch in mein perönliches Interessengebiet fällt, das Aufschreiben dieser Schriftlichen Ausarbeitung hat mir dagegen eher weniger Spaß gemacht. Wobei ich allerdings sagen muss, dass ich doch eine große Erleichterung verspüre das ganze Wissen über Compilerbau mal aufgeschrieben zu haben, dass ich mir aus verschiedenen Quellen angeeignet habe, damit ich mir keine Sorgen machen muss dieses doch wertvolle Wissen irgendwann wieder zu vergessen. Es hilft einem auch als Programmierer ziemlich weiter zu wissen wie ein Compiler unter der Haube funktioniert, man kann sich so viel besser merken, wie eine bestimmte Funktionalität einer Programmiersprache verwendet wird, die sonst ziemlich willkürlich erscheinen würde, wenn man die Technische Umsetzung dahinter im Compiler kennt.

Ich wollte mich auch noch dafür Bedanken, dass mir ein so ergiebiges und interessantes Thema als Bachelorarbeit vorgeschlagen wurde und vor allem, dass auch das Vertrauen in mich gesteckt wurde, dass ich am Ende auch einen funktionsfähigen, sauber programmierten und gut durchdachten Compiler implementiere.

³<https://github.com/michel-giehl/Reti-Emulator>.

Inhaltsverzeichnis

Abbildungsverzeichnis	I
Codeverzeichnis	II
Tabellenverzeichnis	III
Definitionsverzeichnis	IV
Grammatikverzeichnis	V
Literatur	A

Abbildungsverzeichnis

Codeverzeichnis

Tabellenverzeichnis

Definitionsverzeichnis

Grammatikverzeichnis

Literatur