Thesenpapier

Formale Spezifikation mit TLA$^+$

Alexander Weigl

20. November 2011

Resourcen unter: `http://fh-trier.de/~weigla/tla/`

1 Abstrakt

Diese Arbeit zum Fachseminar *Software-Engineering* befasst sich mit der formalen Formulierung von Softwarespezifikationen mit TLA$^+$ *Temporal Logic for Action)*. TLA$^+$ist ein Sprache zum Beschreiben von System unter Verwendung von der Prädikatenlogik und Linearer Temporalen Logik (*LTL*). *LTL* erweitert die Prädikaten (*propositional logic*) um Aspekte einer diskreten zeitlichen Abfolge von aufeinander folgenden Zustände. Mit TLA$^+$werden Syntax und Semantik um Konzepte zur Modulbeschreibung und -wiederverwendung neben vielen neuen Notationen zur uni- oder mehrprozess-orientierten Systemmodellierung eingeführt. Für eine einfache Möglichkeit zur Modellierung wird PlusCal vorgestellt. Dieses stellt eine übergeordnete Sprache zur Algorithmenbeschreibung mit Übersetzung zur TLA$^+$-Modulbeschreibung. Der Abschluss der Arbeit stellt die TLA$^+$-Toolbox und die Vorstellung des Model-Checkers dar.

2 Gliederung

1. Einführung *Wofür formalisieren? Vor- und Nachteil. Beispiel zur Einführung (informell/TLA).*
2. Grundlagen *Erläuterung, das TLA$^+$ auf P/T Logikaufbau, Fairness evtl. nach TLA Kapitel*
   - 2.1. Prädikatenlogik $\forall, \exists, \models$, *Prädikate, Trägermenge*
   - 2.2. Temporale Logik *Operatoren:* $\Box, \Diamond, \bigcirc$, *Semantik, erste Formulierungen von Ausdrücken/Invarianten, Zustände und Schritte, evtl: schon* Suttering Steps
   - 2.3. Fairness und Liveness *vllt. mit obigen verschmelzen, bzw. später bei TLA*
3. Die Sprache TLA
   - 3.1. Syntax/Sematik *Modules, Records, Functions, Tuples*
   - 3.2. Module *Standardmodule: FiniteSets*

3.3. PlusCal *Kurze Erläuterung +Cal: Algorithmenbeschreibung, Erläuterung wie Pascal, Erläuterung Translation nach TLA$^+$*

3.4. Modellierung *Beispiele: HourClock, Memory oder Queue, Euclid oder Fastmutex*

3.5. Modelchecker *Vorgehensweise, Grenzen (Zuviele Zustände, unendliche Mengen.*

4. TLA-Toolbox *ASCII-Syntax einführen, Beschreibungaufbau: Spezifikation/Modell, Erläuterung: Invarianten, ...; PlusCal, Demonstration der Beispiele*

5. Fazit

3 Präsentation

Vorläufig. Fragezeichen markiert optionale Themen.
Ziel: Der Zuhörer sollte nach dem Vortrage grundlegende Konzepte von TLA$^+$und LTL beherrschen und in der Lage sein Semantik und Syntax von TLA$^+$Spezifikationen zu verstehen und Invarianten in LTL zuschreiben.

2min Was bedeutet *formale* Modellierung?

4min Motivation: Was können wir mit TLA. *Beispiel: FastMutex in der Toolbox, Testen des exkl. Ausschlusses*

2min Prädikatenlogik

5min Temporallogik *Operatoren und Zustände/Übergänge, Analogie zur DL-Konfiguration (ThI)*

8min Einführung in TLA *Video: Stirb Langsam 3, Jetzt erst Recht. Beispiel: DieHard.tla, Mit Erläuterung der TLA Syntax*

?5min Records,Funktionen und Tupel in TLA

?6min PlusCal *kurze Sprachvorstellung, Vorgehensweise Translators: +Cal -> TLA$^+$, Automare Schritte, Kurzes Beispiel: Euclid'scher Algorithmus*

5min Erläutern des Modelcheckers

4min Abschluss

$\Sigma = 30..41$ min.
Out-of-Scope: Echtzeit-Anforderungen, Tiefe der LTL, (Records,Funtions,Tuples)?, Instanzen

Literatur

[1] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking.* MIT Press, 2008.

[2] Kaustuv Chaudhuri, Damien Doligez, Leslie Lamport, and Stephan Merz. A TLA$^+$ proof system. In G. Sutcliffe, P. Rudnicki, R. Schmidt, B. Konev, and S. Schulz, editors, *Proc. of the LPAR Workshop Knowledge Exchange: Automated Provers and Proof Assistants (KEAPPA'08)*, number 418 in CEUR Workshop Proceedings, pages 17–37, 2008.

[3] Leslie Lamport. *Specifying Systems.* 1 edition, June 2002.

[4] Leslie Lamport. *TLA^{+2}.* October 2010.

[5] Leslie Lamport. *A PlusCal User's Manual.* 1.5 edition, April 2011.

[6] Stephan Merz. The specification language tla$^+$. In Dines Bjørner and Martin C. Henson, editors, *Logics of Specification Languages*, Monographs in Theoretical Computer Science, pages 401–451. Springer, Berlin-Heidelberg, 2008.