PLEASE TYPE

THE UNIVERSITY OF NEW SOUTH WALES
Thesis/Dissertation Sheet

Surname or Family name: Walker

First name: Adam | Other name/s: Christopher

Abbreviation for degree as given in the University calendar: PhD

School: Computer Science and Engineering | Faculty: Engineering

Title: Automatic device driver synthesis

Abstract 350 words maximum: (PLEASE TYPE)

Device drivers are tedious to write and they are notoriously unreliable. Automatic device driver synthesis is a radical approach to creating drivers faster and with fewer defects by generating them automatically from operating system and hardware device specifications. This thesis addresses the two fundamental challenges in driver synthesis: (1) usability and (2) scalability.

The usability challenge is to enable developers to create drivers through a well-defined, predictable sequence of steps. It falls into the following three design objectives: (1) ease of developing specifications for driver synthesis, (2) ease of debugging synthesis errors due to incorrect specifications, and (3) enabling the benefits of automation without sacrificing the flexibility of conventional driver development. In this thesis I develop a new driver synthesis methodology that addresses the first objective through a modular high-level specification language whose syntax and semantics are close to those of familiar imperative programming languages like C. I address the second objective by developing powerful debugging techniques that help the developer identify and fix specification defects through a well-defined process. Finally, the proposed methodology strikes the balance between automation and flexibility by generating driver code in a user-guided fashion, where the user can interactively alter or amend automatically generated code, while the synthesis tool verifies user changes.

The scalability challenge is to create a synthesis algorithm that is capable of handling real-world device specifications. I address this challenge with a new scalable synthesis engine based on predicate-based abstraction refinement. Solutions are developed to the three key problems involved in the implementation of efficient predicate abstraction, which have not been addressed previously in a synthesis setting: (1) keeping abstractions concise, (2) solving abstract games efficiently, (3) and computing abstractions symbolically.

The proposed techniques are implemented in a new tool called Termite-2 and evaluated by synthesising drivers for a number of I/O devices representative of a typical embedded platform. Termite-2 is the first tool to combine the power of automation with the flexibility of conventional development. It is also the first practical synthesis tool based on abstraction refinement. Finally, it is the first synthesis tool to support automated debugging of input specifications.

Declaration relating to disposition of project thesis/dissertation

I hereby grant to the University of New South Wales or its agents the right to archive and to make available my thesis or dissertation in whole or in part in the University libraries in all forms of media, now or here after known, subject to the provisions of the Copyright Act 1968. I retain all property rights, such as patent rights. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

I also authorise University Microfilms to use the 350 word abstract of my thesis in Dissertation Abstracts International (this is applicable to doctoral theses only).

…… …… ……

Signature | Witness | Date

The University recognises that there may be exceptional circumstances requiring restrictions on copying or conditions on use. Requests for restriction for a period of up to 2 years must be made in writing. Requests for a longer period of restriction may be considered in exceptional circumstances and require the approval of the Dean of Graduate Research.

FOR OFFICE USE ONLY Date of completion of requirements for Award: