

継続を基本とした OS Gears OS

清水 隆博^{1,a)} 河野 真治^{2,b)}

概要：継続を基本とする C と互換性のある言語、Continuation Based C (CbC) を用いて OS の実装を考案した。状態遷移単位で OS の処理を実装することで、処理の入出力が明確化され、定理証明支援系に適した表現形式で処理が記述可能である。ここでは現在 CbC を用いて開発している OS、GearsOS についての実装及び今後の展望について考察する。

キーワード：システムプログラミング, CbC, 軽量継続, OS

1. 証明可能な OS

コンピュータ上で動作するあらゆるソフトウェアや資源を管理する OS は、高い信頼性が保証されてほしい。信頼性の保証にはテストプログラムを用いた検証や、形式手法を用いた証明を使う手法が存在する。頻繁に並列処理を行う OS では、スレッド間の共通資源の競合などの非決定的な実行を行う。このため、OS の信頼性を保証する上で、テストやデバッグを用いる手法では、発生しうる状態を完全に保証するのは困難である。テストを用いる方法ではなく、形式手法的なアプローチを用いて OS の信頼性を保証したい。そのためには定理証明支援系などで証明が可能な形式と、等価な形式で OS を記述する必要がある。現在開発している GearsOS は、継続を基本とする言語 Continuation Based C (CbC) で実装されている。CbC は状態遷移単位での実行であり、他の状態に遷移する際に今までの環境を持たない。そのため、CbC で実装した処理は入出力が明確化され、

定理証明支援系で表現可能な形式にする事が可能である。

2. GearsOS のクロスコンパイル

GearsOS は RaspberryPi 上での動作を目指している。RaspberryPi は ARM の CPU が搭載されている為、動作には ARM のバイナリファイルが必要となる。しかし RaspberryPi を利用して GearsOS 自身のビルドを行うと、マシンパワーからビルドに莫大な時間が掛かってしまう。従って資源が潤沢な別のマシンで ARM のバイナリをビルドするのが望ましい。MacBook などの資源が潤沢な x84 マシンが多く、x86 マシン上から ARM バイナリの CbC のクロスコンパイラを構成する必要がある。GCC 上に実装している CbC コンパイラは、ARM を出力するように再構築する必要があった。他方 LLVM/clang 上に実装している CbC コンパイラは、ARM のライブラリは必要であるものの、本体を再度ビルドすることなくクロスコンパイラとして利用可能である。今回は RaspberryPi のデフォルト OS である Raspbian から、ARM のライブラリを x86 マシン上に転送し、LLVM/clang 上に実装した CbC コンパイラを用いてビルドした。

¹ 琉球大学大学院理工学研究科情報工学専攻

² 琉球大学工学部情報工学科

a) anatofuz@cr.ie.u-ryukyu.ac.jp

b) kono@ie.u-ryukyu.ac.jp

参考文献

- [1] 光希宮城, 優 桃原, 真治河野: Gears OS のモジュール化と並列 API, 技術報告 11, 琉球大学大学院理工学研究科情報工学専攻, 琉球大学大学院理工学研究科情報工学専攻, 琉球大学工学部情報工学科 (2018).
- [2] 並列信頼研究室: CbC_gcc, 琉球大学 (online), available from http://www.cr.ie.u-ryukyu.ac.jp/hg/CbC/CbC_gcc/ (accessed 2018-11-21).
- [3] 並列信頼研究室: CbC_llvm, 琉球大学 (online), available from http://www.cr.ie.u-ryukyu.ac.jp/hg/CbC/CbC_llvm/ (accessed 2018-11-21).