Report on the use of the IBM OpenPower Server:
Developing occam/occam-π for the PowerPC family of processors

Prepared by Fred Barnes[†], November 2005

Project Description

The original aim of this project was to develop the PPC target for the occam [1] and occam-π [2, 3] programming languages, retargeting the KRoC occam-π system developed at the University of Kent [4].

The KRoC system is designed to be retargetable to different platforms and architectures with a modest amount of effort. The most widely used and supported platform/architecture combination is Linux on Intel IA32 processors (Pentium and related processors). Earlier versions of KRoC (pre open-source) targeted a variety of processors, with PPC amongst them [5]. The open-source version of KRoC has working support for Windows/IA32 via Cygwin, and experimental support for Solaris/Sparc8 and Linux/MIPS. The same framework is also used to generate 16-bit virtual-transputer byte-code for embedded (largely robotic) targets running the Transterpreter [6].

The original motivation for this project was providing a port of the open-source KRoC system for Macintosh users — more generally for any platforms using the PowerPC architecture. Since the inception of this project, Macintosh have decided to switch to the Intel architecture, and other architectures with PowerPC elements have come to light (most notably IBM's CELL processor, which was the focus of a keynote talk at CPA-2005 [7]).

Project Status

Development of the PPC port of KRoC is now well underway. Much of the work done has been in the translator portion of KRoC (the now mis-named '`tranx86`' program), that now generates 32-bit PPC code in addition to IA32, MIPS and Sparc. Although most of the KRoC system now builds on PPC, only around two-thirds of the instructions are catered for properly in the translator — libraries will build, but may not work correctly. Simple test programs (e.g. 'hello world') compile and run successfully; anything involving extensive arithmetic or floating-point may fail.

Development is intended to continue over the new few months, hopefully providing a working port of KRoC/Linux for the PPC architecture by the new year. Once the basic port is completed successfully (passes the extensive compiler test-suite), further development effort will look at building KRoC for multi-processor PPC hardware. Multi-processor development for KRoC/Linux has traditionally been hindered by Linux's inadequate support for threading on multi-processors — i.e. having n threads 'gang-scheduled' on n CPUs simultaneously (as provided by SGI's IRIX '`sysmp`' system-call). Having such facilities makes multi-programming significantly easier (and avoids the potentially unpleasant situation of extended spin-locking).

Development on the PPC port, so far, has been exclusively done by myself. It is hoped that the work done so far on the PPC port, which users can check out of the subversion repository, will stimulate further development from others (students, faculty and industry). Of future interest are 64-bit ports and developing for other PPC 'flavoured' architectures, such as the CELL.

We are grateful to the University of Augsburg, Germany for providing access to the IBM OpenPower server.

References

[1] Inmos Limited. occam 2.1 Reference Manual. Technical report, Inmos Limited, May 1995. Available at: `http://wotug.org/occam/`.

[2] P.H. Welch and F.R.M. Barnes. Communicating mobile processes: introducing occam-pi. In A.E. Abdallah, C.B. Jones, and J.W. Sanders, editors, *25 Years of CSP*, volume 3525 of *Lecture Notes in Computer Science*, pages 175–210. Springer Verlag, April 2005.

[3] Frederick R.M. Barnes. *Dynamics and Pragmatics for High Performance Concurrency*. PhD thesis, University of Kent, June 2003.

[4] P.H. Welch, J. Moores, F.R.M. Barnes, and D.C. Wood. The KRoC Home Page, 2000. Available at: `http://www.cs.kent.ac.uk/projects/ofa/kroc/`.

[5] T. Sheen, A.R. Allen, A. Ripke, and S. Woo. oc-X: an optimising multiprocessor occam system for the PowerPC. In P.H. Welch and A.W.P. Bakkers, editors, *Architectures, Languages and Patterns for Parallel and Distributed Applications, Proceedings*

[†]Computing Laboratory, The University of Kent, Canterbury, KENT, United Kingdom. CT2 7NF. email: `F.R.M.Barnes@kent.ac.uk`

of WoTUG 21, volume 52 of *Concurrent Systems Engineering*, pages 167–186, Amsterdam, The Netherlands, April 1998. WoTUG, IOS Press. ISBN: 90-5199-391-9.

[6] Christian Jacobson and Matthew C. Jadud. The Transterpreter: A Transputer Interpreter. In I. East, J. Martin, P. Welch, D. Duce, and M. Green, editors, *Communicating Process Architectures 2004*, volume 62 of *WoTUG-27, Concurrent Systems Engineering, ISSN 1383-7575*, pages 99–106, Amsterdam, The Netherlands, September 2004. IOS Press. ISBN: 1-58603-458-8.

[7] H. Peter Hofstee. Communication and synchronization in the cell processor. In J. Broenink, H. Roebbers, J. Sunter, P. Welch, and D. Wood, editors, *Communicating Process Architectures 2005*. IOS Press, September 2005. Keynote talk.