Linaro CONNECT
ASIA 2013 (LCA13)

Optimisation and Assembly

Steve McIntyre <steve.mcintyre@linaro.org>

Summary – scanning for assembly

Final total: 1435 packages analysed

- Package lists from Ubuntu and Fedora
- Tedious work!
- Takes a long time to scan thoroughly

Analysis

- Looking for:
  - Assembly source files
  - Files containing inline assembly
- **Not** looking for
  - Use of intrinsics for SIMD work
    - Maybe worth revisiting in future

Methods

- Two lists of packages provided
  - Ubuntu list from Kiko and Matthias Klose
  - Fedora list from Al Stone and Jon Masters
- Two different scan method to generate these lists
- One person (me!) reading the source
  - Unpack
  - Look for all likely-looking source files (*.[sS] *.asm *.ASM etc.)
  - Look for inline assembly in other source files
  - See how (and if!) the assembly code is actually used

Categories

- Broad classification of results for each package:
  - Atomics (10.0%)
  - Embedded library code (18.1%)
  - False positives (11.1%)
  - Lowlevel (38.1%)
  - OtherOS (9.3%)
  - Performance (30.4%)
  - Symbols/sections etc. (2.9%)

- Ignorable packages too...
  - architecture-specific packages

Common findings

- Lots and lots (and lots) of assembly for
  - Trivial byteswap code
  - Hardware identification (CPUID)
  - Timer access (RDTSC)
  - SIMD (MMX/SSE/Altivec/NEON) – media libs
  - Trivial atomics
  - FPU control

Far too many embedded libraries

- gnulib
- gettext (well, not really)
- gc
- sqlite
- zlib
- libjpeg

  ...

Patterns

- The Good
  - Many packages are using gcc intrinsics
  - Some newer versions of software moving away from inline asm
- The Bad
  - Lots of cargo-culted asm
  - **Far** too many embedded libraries
    - especially in larger projects like browsers
- The Ugly^WComical
  - Compiler workarounds from 1990s
  - VAX(!) assembly in 10 packages
  - Asm in place, but not actually enabled/used

Priorities for porting work

- Arbitrarily assigned priorities thus far...
- What's considered important for LEG – server workloads
- Dependencies
  - Looking at Ubuntu-recommended server packages as a guide
  - Boost priorities for webserver workloads for now
- Low priority for some types of package
  - Desktop apps
  - Multimedia libs
  - Games

Top 20? It's not that easy!

- Lots of toolchain and core libs are already done or in hand
  - gcc
  - (e)glibc
  - klibc
  - libffi
  - binutils
  - gdb
  - device-tree-compiler
  - libunwind
  - openjdk
  - mpfr
  - llvm

LEG priorities

- grub2
- TBB
- openssl
- libatomic-ops
- gmp
- libgcrypt
- php
- postgres
- mysql
- libaio

...

Wider importance

- Existing code used all over the place / highly embedded
  - zlib
  - gc
  - libjpeg
  - dlmalloc
  - nss
  - gnulib

  …

Not important for LEG

- But will matter to other people
  - Multimedia libs
    - libav
    - libmad

      ...
  - Games

- Plenty of things for us to point developers at...
  - Maybe GSOC project or similar?

Provide help for developers

- Guide them to do the right thing
  - Code examples
  - Use the compiler, don't work around it
- Provide library support for common functions
  - Glibc / elsewhere?
- Talk to them...

Conclusions

- Most software doesn't need porting!
- Even most of those including assembly will already work on ARMv7 and ARMv8
- Most assembly is (essentially) pointless
- We have things to work on, let's go!

Data in Google Docs at http://goo.gl/6z2C8

More about Linaro Connect: www.linaro.org/connect/
More about Linaro: www.linaro.org/about/
More about Linaro engineering: www.linaro.org/engineering/