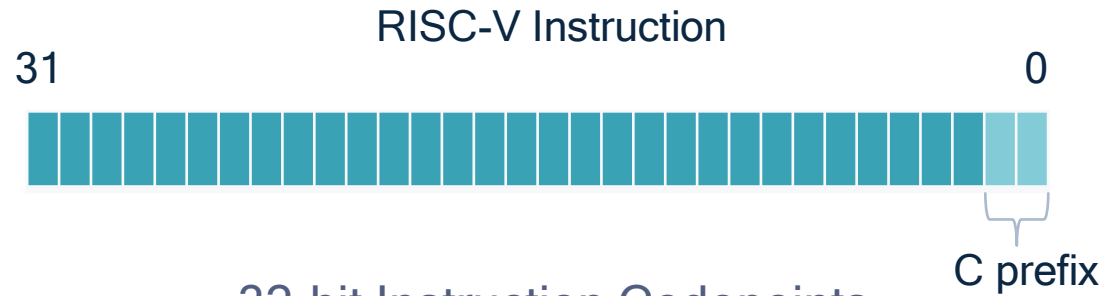


# A Case to Remove C from App Profiles

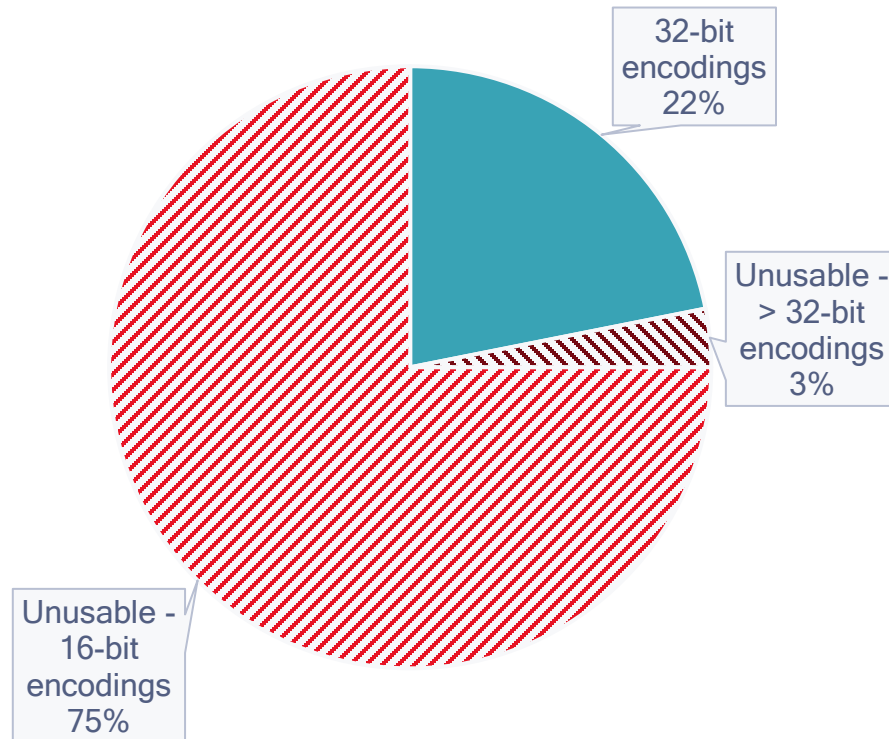
Derek Hower, James Ball, Conrado Blasco, Manu Gulati  
Qualcomm Technologies, Inc.



# C Extension - Compressed (16-bit) instruction encodings



32-bit Instruction Codepoints



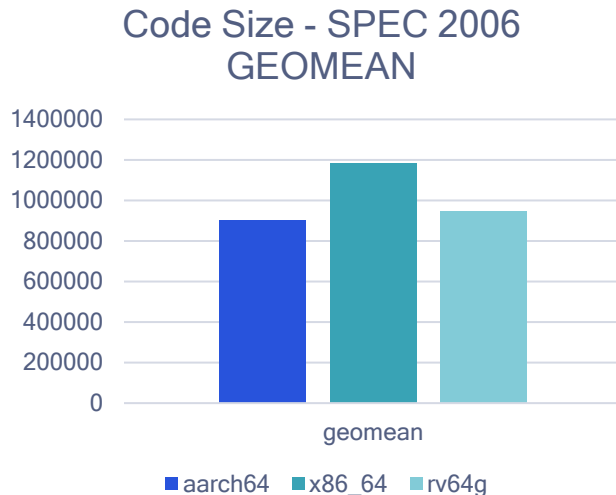
## Facts

- C defines compressed re-encodings of common 32-bit instructions
  - **No new instructions** - always 1:1 mapping between 16/32-bit
- 2 LSBs of instruction identify length
  - 3 of 4 values indicate 16-bit
  - 1 of 4 values indicate  $\geq$  32-bit
- C binaries are packed: **32-bit encodings can be unaligned**

# Problems with C in high performance designs

## Pushes RISC-V to large opcodes

- RISC-V is nearly out of 32-bit opcodes
- **Forced transition** to > 32-bit opcodes will *degrade code size*
  - Eventually, *RV64 with 32-bit encodings will have better code size than RV64C + large encodings* as new instructions are added and used



## Substantial design complexity

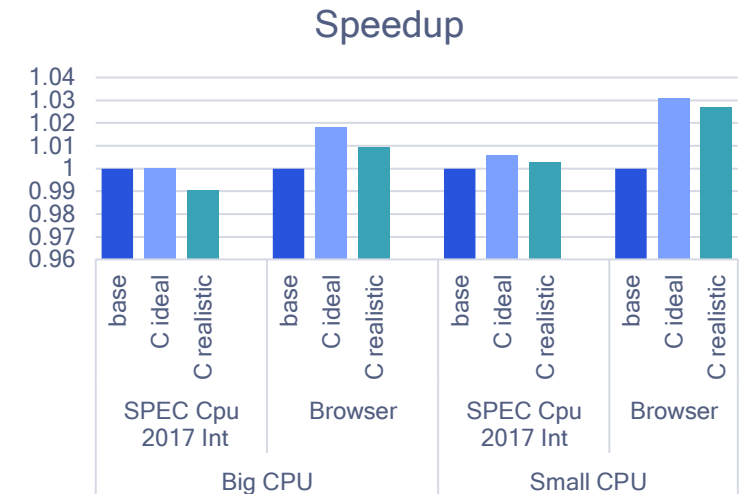
- Unaligned fetch is **challenging to design, verify**
  - Cache line, page crossing instructions
  - Increased wire delay/muxing
- Leads to designs that are:
  - **more expensive** (NRE)
  - **slower** (extra pipe stages)
  - **buggy** (see [Intel Jump Code Conditional](#))

“Under complex micro-architectural conditions involving branch instructions bytes that span multiple 64 byte boundaries (**cross cache line**), unpredictable system behavior may occur.”  
- Intel Xeon Erratum SKX102

Fix results in **0-8% slowdown**.

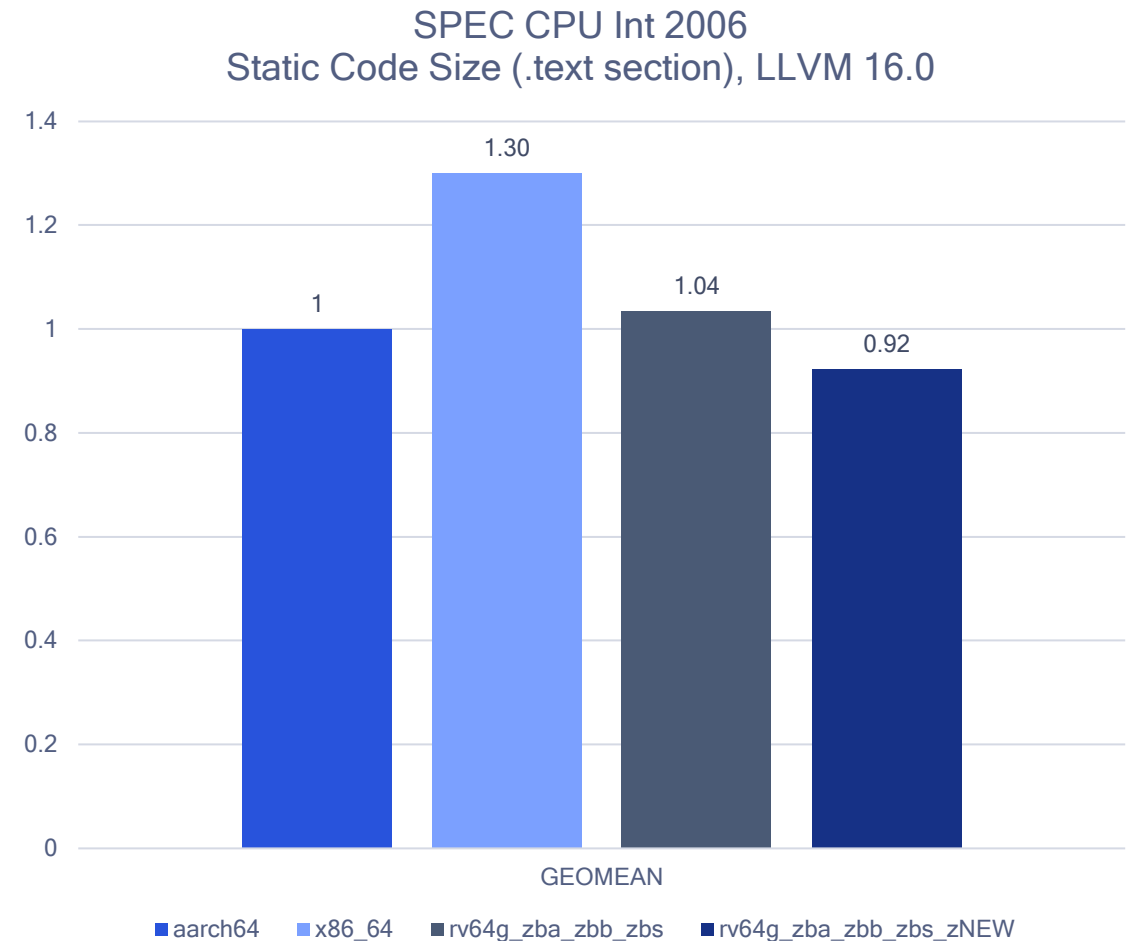
## Performance benefit is modest

- Best case: 2-3% speedup
- Often: slowdown (net negative when program fits in icache)

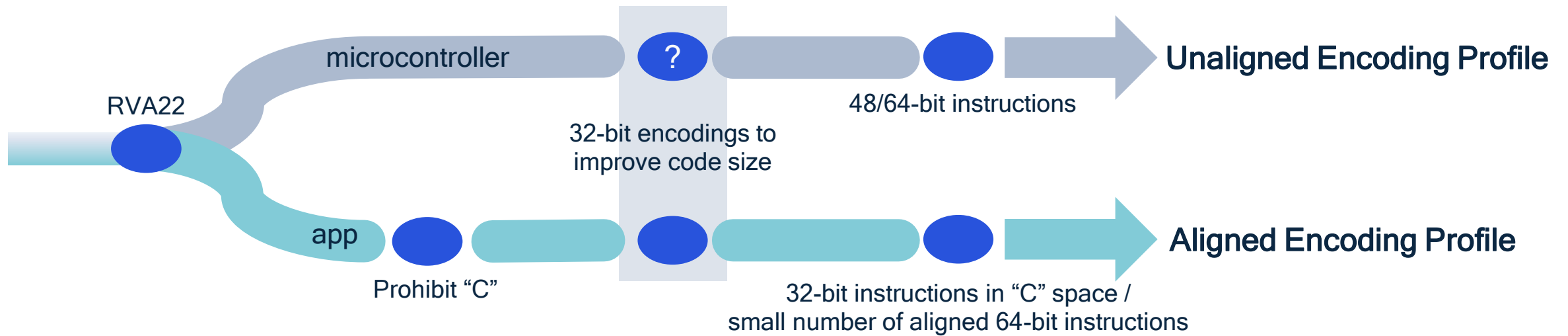


# C is not needed

- RV64G is already competitive on code size
- RV64G + 32-bit instructions for code size is *best in class*
  - More ld/st addressing modes
  - Ld/st pair
  - Conditional immediate branches
  - Move pair
- RV64GC has little performance benefit in beefy designs



# Diverging profiles



- Remove C from application profiles
- Once removed, the C opcode space can be reclaimed to keep code size down long term
- **It's not too late:** commercial distros have not picked a base
  - But, time is running short

# Thank you



Follow us on: [!\[\]\(5eb1325dfdc3f1cad8426726c0db51cd\_img.jpg\)](#) [!\[\]\(312638b5686dbc3f6ff8424fd17b3fb2\_img.jpg\)](#) [!\[\]\(88e39a015d99d67943a7ca963c140a17\_img.jpg\)](#) [!\[\]\(8d24dd9a445af8db71ca36d03e35a691\_img.jpg\)](#) [!\[\]\(f6ca82cef8bde55d35e7e5dc5ce39a28\_img.jpg\)](#)

For more information, visit us at:

[qualcomm.com](http://qualcomm.com) & [qualcomm.com/blog](http://qualcomm.com/blog)

Qualcomm is a trademark or registered trademark of Qualcomm Incorporated. Other products and brand names may be trademarks or registered trademarks of their respective owners.

References in this presentation to "Qualcomm" may mean Qualcomm Incorporated, Qualcomm Technologies, Inc., and/or other subsidiaries or business units within the Qualcomm corporate structure, as applicable. Qualcomm Incorporated includes our licensing business, QTL, and the vast majority of our patent portfolio. Qualcomm Technologies, Inc., a subsidiary of Qualcomm Incorporated, operates, along with its subsidiaries, substantially all of our engineering, research and development functions, and substantially all of our products and services businesses, including our QCT semiconductor business.

Snapdragon and Qualcomm branded products are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.