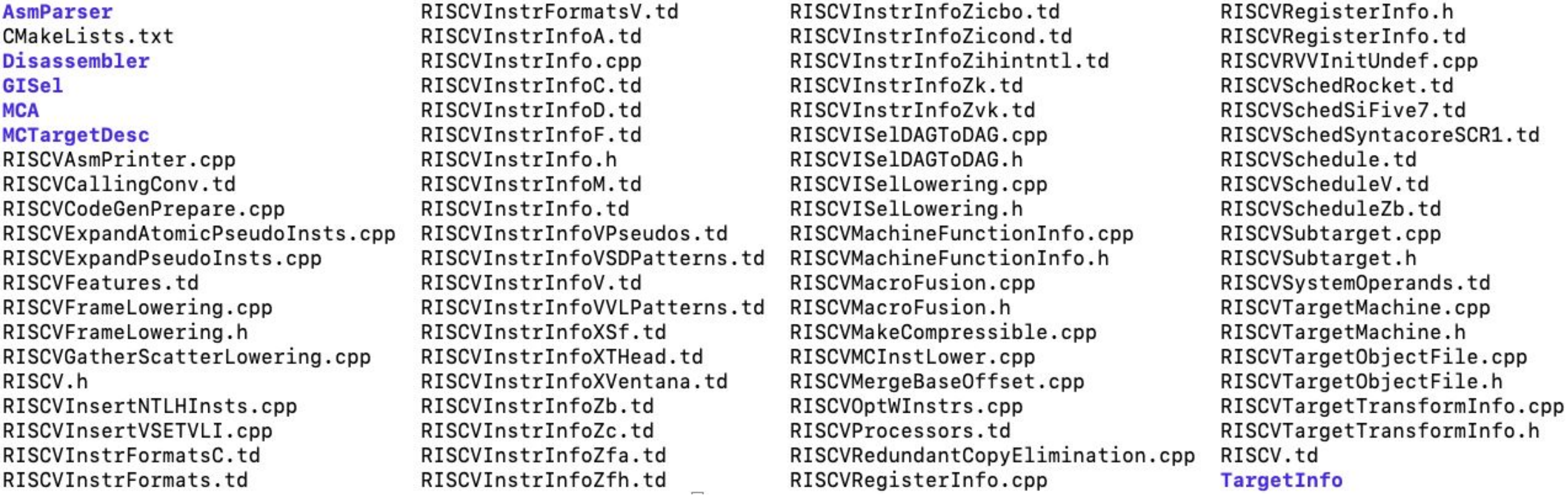
LLVM中RISC-V 子扩展的支持现状

廖春玉
chunyu@iscas.ac.cn

简介 - 来自上次4月22号的报告

- 针对RISCV的代码实现：https://github.com/llvm/llvm-project/tree/main/llvm/lib/Target/RISCV



```
AsmParser                         RISCVInstrFormatsV.td         RISCVInstrInfoZicbo.td             RISCVRegisterInfo.h
CMakeLists.txt                    RISCVInstrInfoA.td            RISCVInstrInfoZicond.td            RISCVRegisterInfo.td
Disassembler                      RISCVInstrInfo.cpp            RISCVInstrInfoZihintntl.td         RISCVRVVInitUndef.cpp
GISel                             RISCVInstrInfoC.td            RISCVInstrInfoZk.td                RISCVSchedRocket.td
MCA                               RISCVInstrInfoD.td            RISCVInstrInfoZvk.td               RISCVSchedSiFive7.td
MCTargetDesc                      RISCVInstrInfoF.td            RISCVISelDAGToDAG.cpp              RISCVSchedSyntacoreSCR1.td
RISCVAsmPrinter.cpp               RISCVInstrInfo.h              RISCVISelDAGToDAG.h                RISCVSchedule.td
RISCVCallingConv.td               RISCVInstrInfoM.td            RISCVISelLowering.cpp              RISCVScheduleV.td
RISCVCodeGenPrepare.cpp           RISCVInstrInfo.td             RISCVISelLowering.h                RISCVScheduleZb.td
RISCVExpandAtomicPseudoInsts.cpp  RISCVInstrInfoVPseudos.td     RISCVMachineFunctionInfo.cpp       RISCVSubtarget.cpp
RISCVExpandPseudoInsts.cpp        RISCVInstrInfoVSDPatterns.td  RISCVMachineFunctionInfo.h         RISCVSubtarget.h
RISCVFeatures.td                  RISCVInstrInfoV.td            RISCVMacroFusion.cpp               RISCVSystemOperands.td
RISCVFrameLowering.cpp            RISCVInstrInfoVVLPatterns.td  RISCVMacroFusion.h                 RISCVTargetMachine.cpp
RISCVFrameLowering.h              RISCVInstrInfoXSf.td          RISCVMakeCompressible.cpp          RISCVTargetMachine.h
RISCVGatherScatterLowering.cpp    RISCVInstrInfoXTHead.td       RISCVMCInstLower.cpp               RISCVTargetObjectFile.cpp
RISCV.h                           RISCVInstrInfoXVentana.td     RISCVMergeBaseOffset.cpp           RISCVTargetObjectFile.h
RISCVInsertNTLHInsts.cpp          RISCVInstrInfoZb.td           RISCVOptWInstrs.cpp                RISCVTargetTransformInfo.cpp
RISCVInsertVSETVLI.cpp            RISCVInstrInfoZc.td           RISCVProcessors.td                 RISCVTargetTransformInfo.h
RISCVInstrFormatsC.td             RISCVInstrInfoZfa.td          RISCVRedundantCopyElimination.cpp  RISCV.td
RISCVInstrFormats.td              RISCVInstrInfoZfh.td          RISCVRegisterInfo.cpp              TargetInfo
```

RISC-V使用

- 尝试使用：https://godbolt.org/

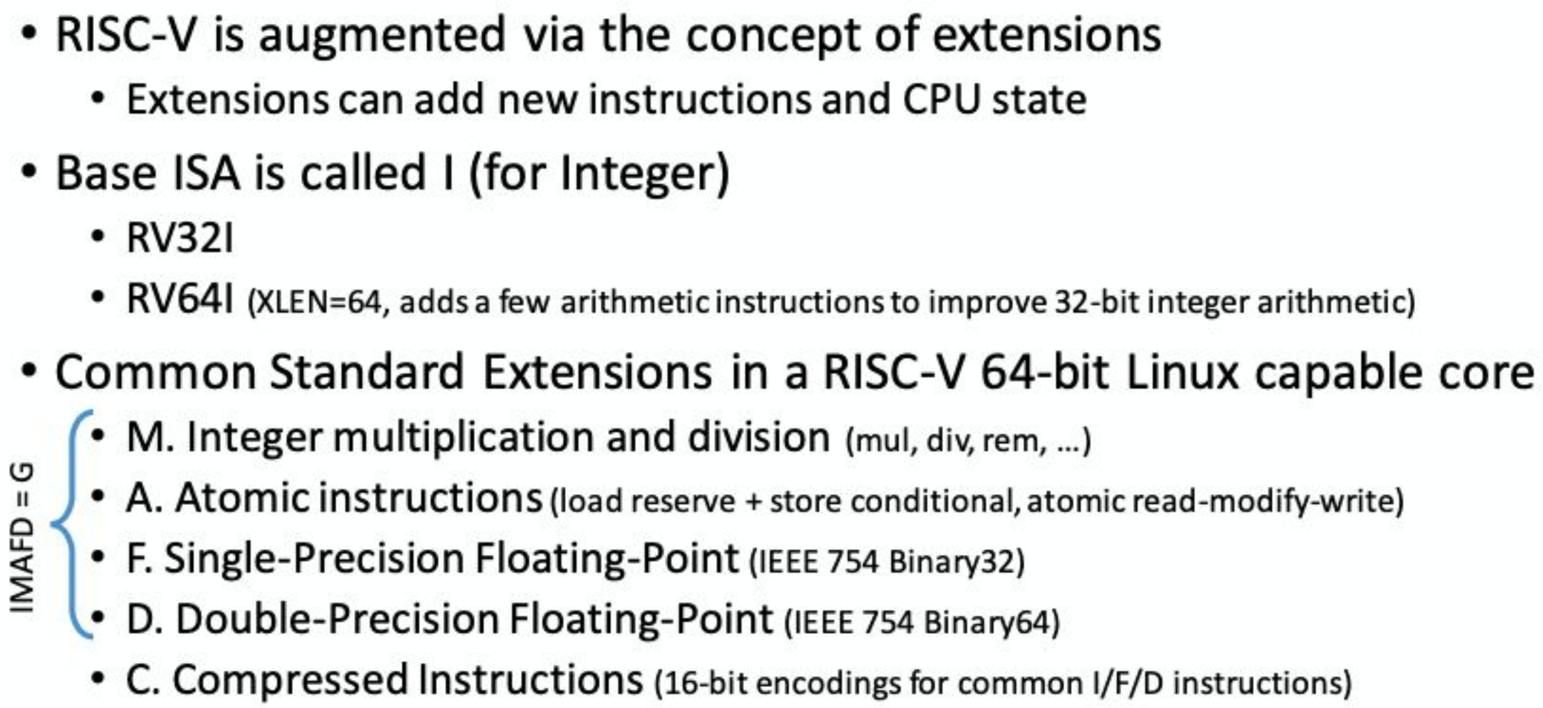
- 本地编译Clang/LLVM
- git clone git@github.com:llvm/llvm-project.git
- cd llvm-project
- mkdir build
- cd build
- cmake -DLLVM_TARGETS_TO_BUILD="X86;RISCV" -DLLVM_ENABLE_PROJECTS="clang" -DCMAKE_BUILD_TYPE="Release" -G Ninja ../llvm
- ninja
- ninja check

What's new for RISC-V in LLVM 16

- https://muxup.com/2023q1/whats-new-for-risc-v-in-llvm-16
  - 优化
  - 子扩展
  - lld/lldb等

- User Guide for RISC-V Target https://llvm.org/docs/RISCVUsage.html --后面看

RISC-V扩展



- RISC-V is augmented via the concept of extensions
  - Extensions can add new instructions and CPU state
- Base ISA is called I (for Integer)
  - RV32I
  - RV64I (XLEN=64, adds a few arithmetic instructions to improve 32-bit integer arithmetic)
- Common Standard Extensions in a RISC-V 64-bit Linux capable core

IMAFD = G

  - M. Integer multiplication and division (mul, div, rem, ...)
  - A. Atomic instructions (load reserve + store conditional, atomic read-modify-write)
  - F. Single-Precision Floating-Point (IEEE 754 Binary32)
  - D. Double-Precision Floating-Point (IEEE 754 Binary64)
  - C. Compressed Instructions (16-bit encodings for common I/F/D instructions)

https://eupilot.eu/wp-content/uploads/2022/11/RISC-V-VectorExtension-1-1.pdf

RISC-V扩展

- RISC-V有多少个扩展呢？？？？
- 几个由我们小队实现的扩展，每个扩展都有独特的的意义
    - z*inx
    - zc*
    - k扩展

z*inx

- SPEC
  - 很早之前: https://github.com/riscv/riscv-zfinx
  - 现在也可以: https://github.com/riscv/riscv-isa-manual/releases
- LLVM
  - 大部分都可以从upstream中直接获取
  - 最难的部分rv32 zdinx : https://reviews.llvm.org/D149743

zc*

- SPEC
  - https://github.com/riscv/riscv-code-size-reduction
- psABI
  - Zcmt, https://github.com/riscv-non-isa/riscv-elf-psabi-doc/pull/349
- LLVM
  - 汇编器全部支持
  - zcmp codegen 还在进行中https://reviews.llvm.org/search/query/wO31sK76dGV_/#R
  - zcmt lld 还在进行中 https://reviews.llvm.org/D134600

RISC-V扩展

- Spec
  - 用途，编码，别名，pseudo 。。。。
  - https://github.com/riscv/riscv-isa-manual
- intrinsic标准
  - rvv intrinsic见证发展的艰辛
    https://github.com/riscv-non-isa/rvv-intrinsic-doc
- 厂商指令

LLVM中已经支持的扩展

- User Guide for RISC-V Target https://llvm.org/docs/RISCVUsage.html

如何在LLVM中贡献RISC-V扩展

- 大方向没有争议的扩展downstream
- upstream
    - mc
    - codegen
    - clang
    - …..

如何在LLVM中贡献RISC-V扩展

- 大方向有争议的扩展
- 积极参与各种实现方式都尝试
    - zmmul https://reviews.llvm.org/D103313
    - no-div

现在还有机会吗？

- 公共子扩展
  - 关注RISC-V
  - 关注其他编译器/软件
- 厂商扩展- CORE-V
  - https://github.com/openhwgroup/cv32e40p/blob/master/docs/source/instruction_set_extensions.rst
  - https://github.com/riscv-non-isa/riscv-toolchain-conventions/pull/29

各个扩展的具体实现

- vscode - k-ext
- fclass.d
- 还有什么想看的扩展吗？现场一起学习