Scalar Efficiency SIG Meeting

May 16, 2024

Derek Hower, Qualcomm

Agenda

- Infrastructure
  - Toolchains & targets
  - Workloads
- Analysis
  - XTheadMemPair and Zilsd

Analysis repository

https://github.com/riscv-software-src/se-sig-analysis

Contains

- Scripts to create/push/pull container with toolchains
- Scripts to build workloads
- Scripts to estimate static code size effect of new instructions
- [TODO] QEMU to estimate dynamic instruction effect of new instructions

Requirements

- singularity container system
- ~1 TB disk space (To build everything)

Toolchains & targets

Table 1. Toolchains/Targets

	Name	Toolchain	Version	Arch	Abi	Flags
RTOS	`rtos32_llvm`	LLVM	18.1.6	`rv32ima_zba_zbb_zbs_zca_zcb_zcmp_zcmt`	`ilp32`	`-Os`
	`rtos32_gcc`	GCC	14.1	`rv32ima_zba_zbb_zbs_zca_zcb_zcmp_zcmt`	`ilp32`	`-Os`
	`rtos64_llvm`	LLVM	18.1.6	`rv64ima_zba_zbb_zbs_zca_zcb_zcmp_zcmt`	`lp64`	`-Os`
	`rtos64_gcc`	GCC	14.1	`rv64ima_zba_zbb_zbs_zca_zcb_zcmp_zcmt`	`lp64`	`-Os`

	Name	Toolchain	Version	Arch	Abi	Flags
Embed Rich OS	linux32_llvm	LLVM	18.1.6	rv32ima_zba_zbb_zbs_zca_zcb_zcmp	ilp32	-Os
	linux32_gcc	GCC	14.1	rv32ima_zba_zbb_zbs_zca_zcb_zcmp	ilp32	-Os
	linux64_llvm	LLVM	18.1.6	rv64ima_zba_zbb_zbs_zca_zcb_zcmp	lp64	-Os
	linux64_gcc	GCC	14.1	rv32ima_zba_zbb_zbs_zca_zcb_zcmp	lp64	-Os
App Rich OS	linux64_app_llvm	LLVM	18.1.6	rv64gcv_zba_zbb_zbs_zcb	lp64d	-Ofast
	linux64_app_gcc	GCC	14.1	rv64gcv_zba_zbb_zbs_zcb	lp64d	-Ofast
	android64_llvm	LLVM	18.0.1	rv64gcv_zba_zbb_zbs_zcb	lp64d	'-Ofast'

Workloads

Class	Name	Version	Status
RTOS	Zephyr (examples TBD)	3.6.0	**Done**
	Embench IoT	1.0	**Done**
	Coremark	Pro	Not started
Embeded Rich OS	Yocto Poky	TODO	Not started
App Rich OS	SPEC CPU	2017	**Done**
	AOSP (userspace)		**Done**
	Linux Kernel	TODO	Not started
	V8	TODO	Not started
	V8 (Sunspider)	TODO	Not started — discussion needed
	V8 (Octane)	TODO	Not started — discussion needed
	V8 (Speedometer)	TODO	Not started — discussion needed

Analysis

- Script that estimates code size savings by finding/replacing instruction sequences

	Suite	Target	Static Size Reduction	Static Instruction Count Reduction
`XTheadMemPair`	AOSP	`android64_llvm`	0.5%	1.5%
	Embench IOT	`rtos32_llvm`	0.5%	1.5%
		`rtos32_gcc`	0.5%	1.5%
`Zilsd`	Embench IOT	`rtos32_llvm`	2.3%	3.5%
		`rtos32_gcc`	2.5%	3.6%

Analysis Breakdown

aosp

:xtheadmempair: :lwd: 27086 :lwud: 0 :ldd: 3598141 :sdd: 981878 :swd: 10110

embench_iot

:xtheadmempair: :lwd: 18312 :lwud: 0 :ldd: 0 :sdd: 0 :swd: 3494 :zilsd: :ld: 31982 :cld: 0 :cldsp: 11741 :sd: 6241 :csd: 110 :csdsp: 0

Instruction database format

- Presented Google Sheet format last month
- Text format suggested to manage concurrent work. See prototype
  - Instruction data specified in YAML files.
  - Vendors can be separate.
  - Script aggregates into Asciidoc table.

Processor classes

- See Draft