

A5 - Simulatoropgaven

Opgave

Der skal implementeres en disassembler og simulator for RISC-V, 32-bit base instruktionssæt og udvidelsen for 32-bit multiplikation/division. Dog skal enkelte specielle instruktioner ikke indgå. Derudover skal simulatoren håndtere nogle få systemkald. Detaljer gives nedenfor. Simulatoren skal skrives i C.

A5 - Simulatoropgaven

Til hjælp udleveres en ufærdig simulator som skal færdiggøres. Den udleverede kode tager hånd om simulation af lager og indlæsning af et program til dette lager. For at løse opgaven skal I tilføje filen "simulate.c" og der implementere funktionen "simulate()", som allerede er erklæret i "simulate.h". Endvidere skal I tilføje filen "disassemble.c" og der implementere funktionen "disassemble()" som allerede er erklæret i "disassemble.h".

Der er tidligere udleveret nogle programmer (find dem i "resources/tiny_riscv" i kursets offentlige repo). De fire af dem ("hello.c", "echo.c", "fib.c" og "erat.c") skal kunne afvikles på den færdige simulator. Disse programmer er skrevet i C og du kan bruge krydscompileren introduceret i tidligere forelæsning til at oversætte dem til et format som den færdige simulator kan læse.

Der skal opstilles og gennemføres en afprøvning af både disassemblering og simulering af alle instruktionerne.

A5 - Simulatoropgaven

Besvarelsen skal indeholde:

- En kort rapport. Her bør I kort beskrive:
 - Jeres overordnede design/tilgang
 - Hvordan I afkoder instruktionerne (herunder genererer den korrekte "immediate" hvis relevant)
 - Hvordan I har afprøvet og hvad jeres afprøvning viser.
 - Hvis der er noget i ikke lykkedes med at implementere, hvorfor tror i så det var?
- Implementationen af simulerings-funktionen
- Implementationen af disassemblerings-funktionen
- Dokumentation for afprøvning af simulatoren. For hver instruktion bør i anføre:
 - Hvad i afprøver
 - Hvordan i konkret afprøver det i form af et minimalt assembler program
- Dokumentation for at de nævnte 4 C programmer kan afvikles af simulatoren

A5 - Hvilke instruktioner

Den autoritative reference (herefter kaldet "riscv spec") for instruktionernes indkodning er her:

Detaljer: <https://github.com/riscv/riscv-isa-manual/releases/download/20240411/unpriv-isa-as>

Bemærk især figurene i afsnit 2 i dette dokument.

Simulatoren skal håndtere RISC-V 32 bit base instruktionssæt (I) og udvidelsen for 32 bit multiplikation (M). Instruktionerne i base instruktionssættet er alle nævnt på "det grønne ark" som indgår i COD. Arket nævner også RV64M udvidelsen (64 bit multiplikation), men det er altså 32-bit udvidelsen som skal implementeres. Denne udvidelse ligner meget men er ikke identisk. Find den beskrevet i riscv spec.

Detaljer om hver instruktion findes i riscv spec.

Dog skal følgende instruktioner fra det grønne ark IKKE implementeres:

ebreak, fence, fence.i, CSRW, CSRRS, CSRR, CSRRWI, CSRRSI og CSRRCI.

A5 Systemkald

Systemkald udføres med "ecall" instruktionen. Værdien i A7 angiver hvilket systemkald der er tale om. Der skal implementeres følgende fire systemkald.

| Kald: | Effekt: |
|---------|---|
| 1 | returner "getchar()" i A0 |
| 2 | udfør "putchar(c)", hvor c tages fra A0 |
| 3 og 93 | afslut simulationen |

