Arm Instruction
Emulator Exercises
This is a simple example

We build and run the HACCKernels code

No SVE instructions
  - Only native NEON

Execute the code

02_HACC_Inscount

- This example makes use of a very simple client

- Inscount_emulated
  - Counts the number of natively executed and emulated instructions

- Testing for an emulated 128-bit SVE width

- Show how to use -only_from_app flag to only count application instructions
03_HACC_SVELength

• This example builds upon the previous one

• Runs HACCKernel through ArmIE
  • Varies the SVE width

• Investigates the impact
04_HACC_Opcodes

• This example makes use of a new client

• Opcodes_emulated

• Don’t just count the instructions, look at what they are

• Both native and emulated
05_HACC_Memtrace

• In this example we move away from instructions and look at memory operations

• Memtrace_simple
  • Records every memory instruction
  • Classifies them (read/write)
  • Data size
  • SVE Bundle

• Generates vast amounts of data
  • Requires code instrumentation to enable

• Some basic post processing
06_Fortran_Memtrace

• Repetition of 07_HACC_Memtrace but with a Fortran example

• ISO_C_Bindings for calling profiling interface

• Test from CloverLeaf Kernel Drivers (Accelerate)
  • Easy to reproduce in own code
07_sve2_vecdot_f16

- Demonstrates SVE2 instructions with ArmIE
- Calls VECDOT on float16
- Does not run natively on Fujitsu A64FX, which implements SVE (not SVE2)
- Use ArmIE to run on any Arm CPU, even if it doesn’t implement SVE
08_sve2_histogram_vla

- Demonstrates SVE2 instructions with ArmIE
- Does not run natively on Fujitsu A64FX, which implements SVE (not SVE2)
- Use ArmIE to run on any Arm CPU, even if it doesn’t implement SVE
09_sve2_skipwhitespace

- Demonstrates SVE2 instructions with ArmIE
- Does not run natively on Fujitsu A64FX, which implements SVE (not SVE2)
- Use ArmIE to run on any Arm CPU, even if it doesn’t implement SVE
Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
tוודה