Going Arm: SVE User's meeting

Tools for SVE code development

16 Nov 2017

Ashok Bhat
Product Manager, HPC tools

Compilers – SVE support in GNU and LLVM

Open source compiler support

GNU plan

GCC 7 – No support for SVE

GCC 8 – Q2, 2018 - Initial SVE support with limited auto-vectorization

GCC 9 – Q2,2019 – Full auto vectorization support

LLVM/Clang plan

LLVM 5.0 – No support for SVE

LLVM 6.0 – Q1, 2018 – Incomplete SVE support

LLVM 7.0 – Q3, 2018 – Initial SVE auto vectorization support

LLVM 8.0 – Q1, 2019 – Full SVE auto-vectorization support

arm

Commercial compilers with SVE support

Provided by Arm

Bare Metal

DS-5 product from Arm has Arm Compiler 6 with support for bare-metal SVE code generation

Visit http://developer.arm.com/ds-5 for further information

Linux User-Space

Arm Allinea Studio has Arm C/C++ Compiler and Arm Fortran Compiler, both of which support user-space SVE code generation.

Visit www.arm.com/hpc-tools for further information

arm

Simulation – Arm Fixed Virtual Platform (FVP)

https://developer.arm.com/products/system-design/fixed-virtual-platforms

Provides Programmer's view

Functionally accurate – Not cycle accurate

Useful for

- ✓ Software development
- ✓ Software profiling
- ✓ OS bring up

Base Platform FVP

Freely available

Runs on x86 host platform

Ready for Armv8-A with SVE extension

50-200 MIPS performance

Two sets of 1-4 cores each

Boots Linux in ~30 seconds

arm

Open source emulators – gem5 and QEMU

SVE support is being added, not complete yet

gem5

Tool for computer-system architecture research and exploration

Supports Arm and other architectures

Arm Research contribute to gem5 community.

Initial patches published for SVE. More patches expected soon.

Much slower than Fast Model and ArmIE

QEMU

Generic and open source machine emulator.

Usually faster than models

Two engineers are working in Linaro to add SVE support for QEMU

Likely to be added in H2,2018

arm

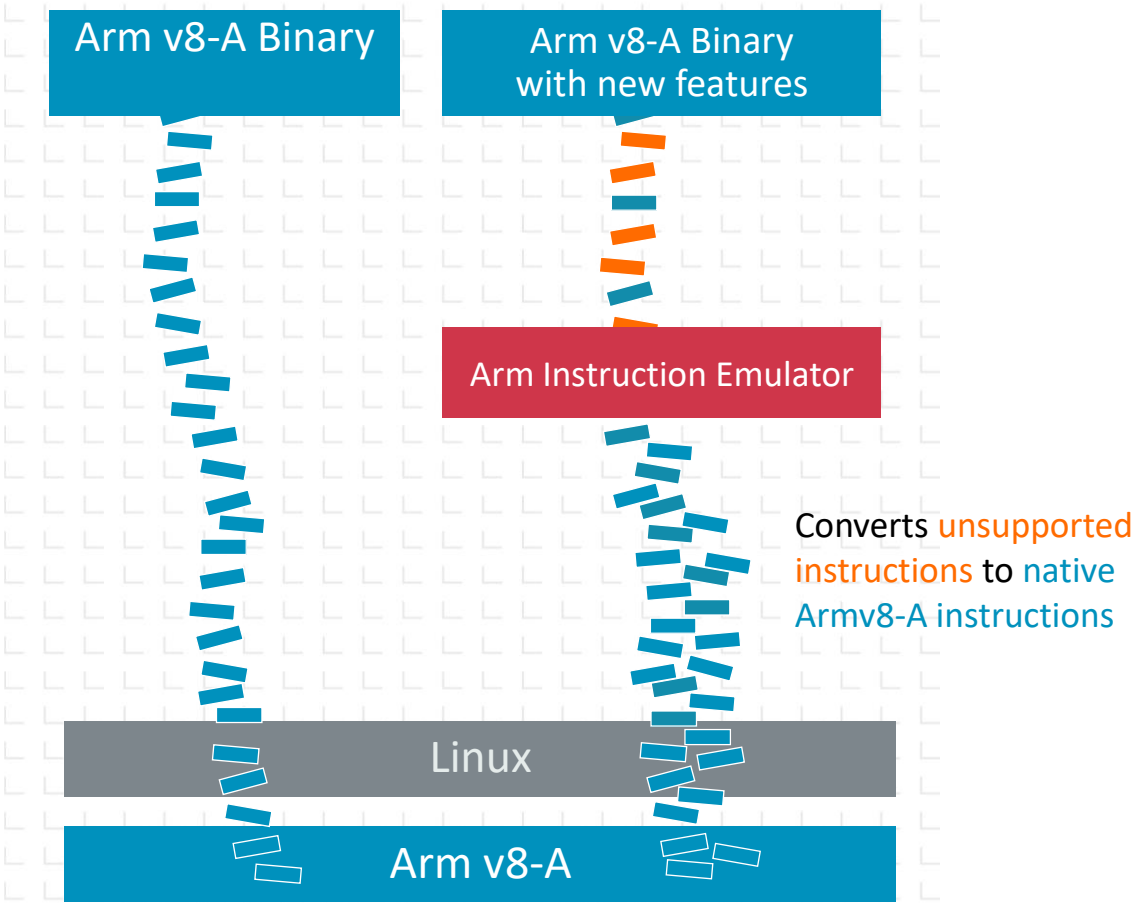
Arm Instruction Emulator

Develop your user-space applications for future hardware today

Run Linux user-space code that uses new hardware features (SVE) on current Arm hardware

Simple “black box” command line tool

```
$ armclang hello.c --march=armv8+sve
$ ./a.out
Illegal instruction
$ armie -a=armv8+sve ./a.out
Hello
```



arm

Instruction Emulator – What next?

Current solution is based on SIGILL trap based solution.

Plan is to migrate to DynamoRIO based solution.

What is DynamoRIO?

- An BSD-licensed open-source dynamic binary instrumentation framework.
  Similar to Intel's PIN framework.
- On github, managed by Google
- Supports Intel and Arm

Why?

- Allows doing more complex analysis like memory tracing, instruction counting, basic block analysis.