

Debugging offloaded kernels on AMD GPUs

Linux Plumbers Conference 2021

Andrew Stubbs

| Agenda

Introduction

Demonstration

GCC Implementation

Introduction

About me: Andrew Stubbs

Sourcerer at CodeSourcery / Mentor Graphics, now part of Siemens.

GCC developer working on (most recently):

- AMD GCN backend.
- OpenMP & OpenACC for GCN & NVPTX.
- GCN DWARF support.

Introduction

ROCGDB

The Radeon Open Compute project is porting and adapting GDB to allow debugging programs that combine both CPU code and GPU code (kernels) in one seamless session.

- Developed by AMD.
- Supports AMD GCN5 GPUs upwards.
- Included with ROCm driver installs.
- ROCm is available for Ubuntu, CentOS/RHEL, & SLES.

<https://rocmdocs.amd.com>

https://rocmdocs.amd.com/en/latest/ROCm_Tools/ROCgdb.html

Introduction

GCC “OG11”

The compiler demonstrated here is built from the “OG11” development branch: `devel/omp/gcc-11`

Work continues merging OpenACC and debugging features to GCC mainline.

Build/install instructions: <https://gcc.gnu.org/wiki/Offloading>

Binaries are available from some Linux distributions:

- Ubuntu: `gcc-11-offload-amdgcn`
- SUSE: `cross-amdgcn-gcc11`

| Demonstration

<https://youtu.be/IGWFph4SlpU>

GCC Implementation

Challenge 1: Register sizes (complete)

GCN has 64-bit addresses, but only 32-bit registers, so the stack pointer & frame pointer each consist of a pair of adjacent registers.

- The *only* GCC architecture to do so.
- GCC's target-independent DWARF support needed a full retro-fit.
- DWARF itself does not work naturally.

GCC Implementation

Challenge 2: Early debug information (complete)

GCC emits DWARF in two phases: the “early” debug comes before LTO stream-out, the rest when code generation is complete. These are then combined by the linker.

- “Before LTO stream-out” implies that this is *x86-64 debug code*.
- The binary file containing the early debug must be translated to GCN before it can be linked in.
- “mkoffload” does binary editing in the ELF file.
 - Extract sections from the LTO object.
 - Apply the correct ELF headers.
 - Convert all the relocation codes.
 - Weaken references to host symbols not present in the offload code.

GCC Implementation

Challenge 3: Incomplete DWARF (in progress)

Offload kernels are formed as nested functions within the host function, but that function *does not exist within the GCN binary*.

- GDB simply ignores DWARF for functions that “do not exist”, including functions nested within.
- The DWARF encoding for the “notional” parent functions must be adjusted.
 - The OG11 development branch has an interim fix.
 - Abid Qadeer is working on a real solution for mainline.

GCC Implementation

Challenge 4: Address spaces (to do)

OpenACC “gang-private” variables are located in the local on-chip low-latency memory (LDS).

- GDB cannot extract the variable unless the address space information is set correctly.
- ROCGDB did not support this yet, so GCC support was not added.
 - (Last time I tried it; there have been updates since.)

GCC Implementation

Challenge 5: Scalars in vectors (to do)

GCN has many instructions that are only available for use with vectors and vector registers, therefore GCC often places scalar variables in vector registers.

- GDB does not know that the vector is being used in a scalar mode.
 - GDB expressions using the variable give unexpected results!
- GCC needs to encode the variable location differently.
 - AMD have proposed DWARF extensions permit this.
- This is also a problem for vector variables that use fewer than 64 lanes.

In general, vectorized code can be expected to be hard to debug (which is why the demo didn't do that).



Questions?

| Contact

Published by Siemens 2021

Andrew Stubbs

Sourcerer

Sourcery Tools Services / CodeSourcery / Siemens (formerly Mentor Graphics)

E-mail andrew_stubbs@mentor.com

Disclaimer

© Siemens 2021

Subject to changes and errors. The information given in this document only contains general descriptions and/or performance features which may not always specifically reflect those described, or which may undergo modification in the course of further development of the products. The requested performance features are binding only when they are expressly agreed upon in the concluded contract.

All product designations may be trademarks or other rights of Siemens AG, its affiliated companies or other companies whose use by third parties for their own purposes could violate the rights of the respective owner.