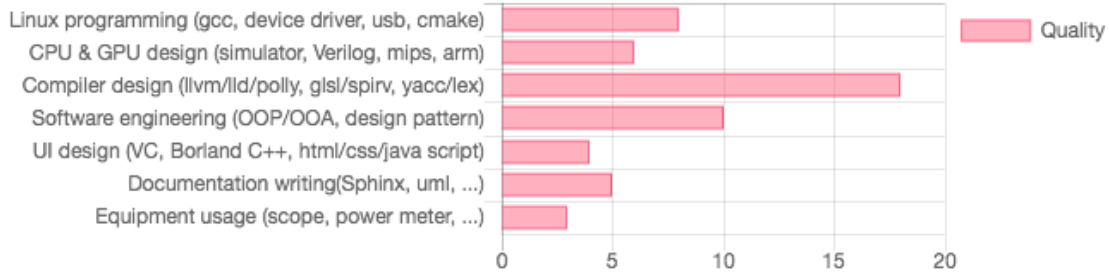*Chung-Shu Chen 陳鍾樞*
QUALIFICATION
Compiler developer with good experience in llvm cpu and gpu backend, lld linker, npu/onnx, c++, OpenGl/glsl and simulator for 8 years, over 20 years experience in c/c++ programming and research in parallel processing for master degree.

RESUME

SKILLS

MY OPEN SOURCE PROJECT

I am proud of my work is accepted by LLVM documentation, appears at http://llvm.org/docs/tutorial/#external-tutorials

Tutorial: Create an LLVM Backend compiler   http://jonathan2251.github.io/lbd/index.html

Tutorial: Create an LLVM Backend Toolchain   http://jonathan2251.github.io/lbt/index.html

The concept of GPU compiler   http://jonathan2251.github.io/lbd/gpu.html

EDUCATION

1997-1999 Master, June 1999, National Taiwan Normal University (國立台灣師範大學), Taipei, Major: Information Science.

1991-1994 B.S., June 1994, National Taiwan Technology University of Science and Technology (國立台灣科技大學), Taipei, Major: Industry Engineer.

LICENSE

Taiwan National Computer Engineer license, 1995 高考資訊技師及格.

EXPERIENCE

September 2004 - June 1999:

Proton 2014/3 – 2014/9 Manager Digital TV programming, Abocom 2013/6 – 2014/3, Senior Engineer 802.11b programming,

DBTEL 2011/11 – 2013/6 Engineer DECT wireless phone programming, Symmetry 2011/2 – 2011/11 Engineer, SGSN and GGSN for GPRS&3G programming,

Cando 2010/7 – 2011/2 Engineer CAM programming, Spirox 2009/12 – 2010/7, Engineer CAM programming, Intech 2009/6 – 2009/12 Engineer CAM programming

THESIS OF MASTER DEGREE

The Researches of Column Sort and Related Problems at library (input "column sort")

Publish Conference: 陳鍾樞，林順喜，2000，行排列法簡化步驟之研究，2000網際網路與分散式系統研討會，台南成功大學，台灣，中華民國，Vol.2，頁85-93。

PROPOSAL OF PHD STUDY

The Researches of Sorting Network and Related Algorithm

Research report

林順喜，陳鍾樞，賴有得，1999，The Design and Research of Lower-Cost Sorting Networks，NSC88-2213-E003-002, Sponsored by Ministry of Science and Technology

OTHER WORK

Take course "Image processing" and program: Jpeg decoder

Web and javascript: As my resume and my personal web site

Graphivz: as some graph diagrams used in this CV. Source code: mywork_1.gv and study_and_apply.gv

ACHIEVEMENT
Hisilcon

GPU compiler scope:

GPU Just In Time Compiler Flow

frontend(opengl, vulkan and opencl compiler) → llvm IR → backend → obj → linker

To support an our new designed GPU for cell phone, ported from ARM. 20% of frontend is changed, 50% of backend is changed in aspect of number of code lines.

My work:

Implement compiler (fontend + llvm backend) for 80% of texture related API and optimization by myself alone and document writing.
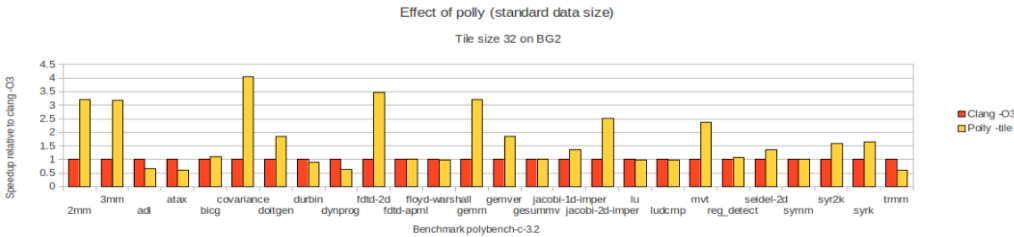
Instruct and help other engineers for the other 20% of texture related API, review their implementation and co-work with the leader of texture part of architect.

Implement compiler supporting our GPU's load/store for RGBA fixed floating point format of vulkan (32, 16, 11, 10 and 2 bits; NaN Infinity) alone and document writing.

Marvell

Implement semi-auto software system of running benchmark and generating report for gcc toolchain.

Demonstrate polly and the concept of polyhedral optimization model for Marvell llvm and gcc toolchain optimization. Polly is a software for loop optimization.

Effect of polly (standard data size)
Tile size 32 on BG2

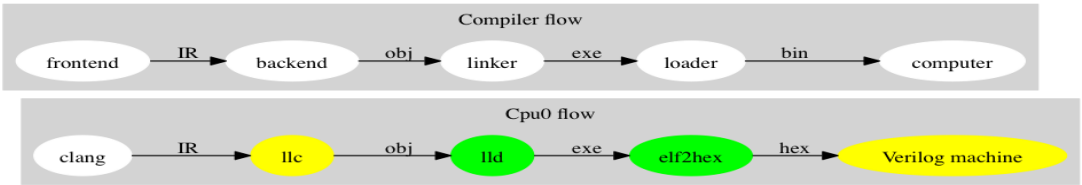Implement co-simulator for a few Marvell's ARM based 64-bit cpu.

Propose and implement DSL on simulator to save tens of system verification in c++ coding.

Complete cmake to replace make for Csim.

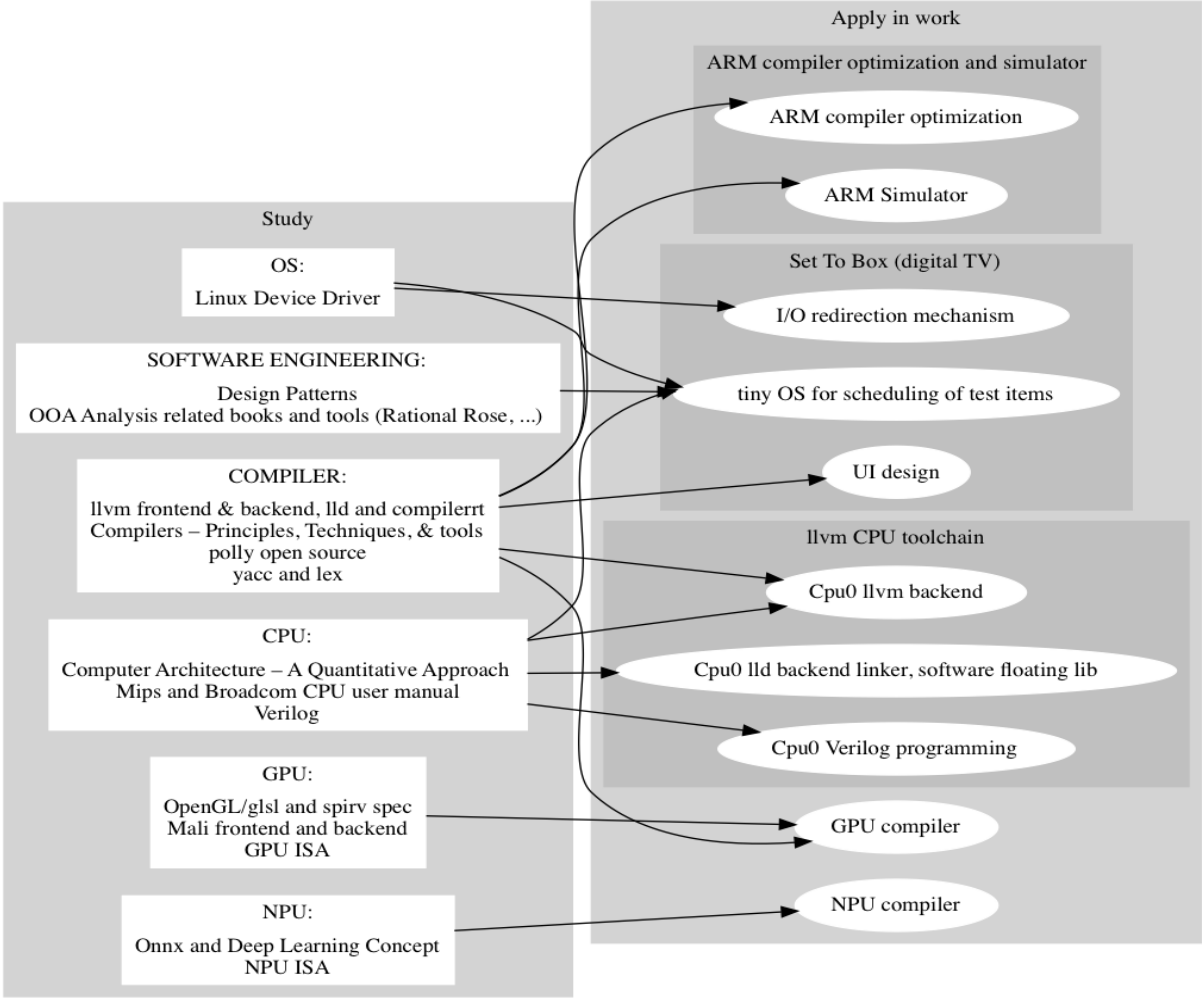Advantage: simpler and cross-os-platform than make.

LLVM open source project

The lower half is the my llvm backend's work flow. Yellow and green parts are my implementation in my books.

Compiler flow
frontend → IR → backend → obj → linker → exe → loader → bin → computer

Cpu0 flow
clang → IR → llc → obj → lld → exe → elf2hex → hex → Verilog machine

Mortorola

Develop Set Top Box's software framework.

Learning after school & applying in work

References

My former manager's recommendation letter: https://jonathan2251.github.io/ws/en/RL_Marvell.pdf