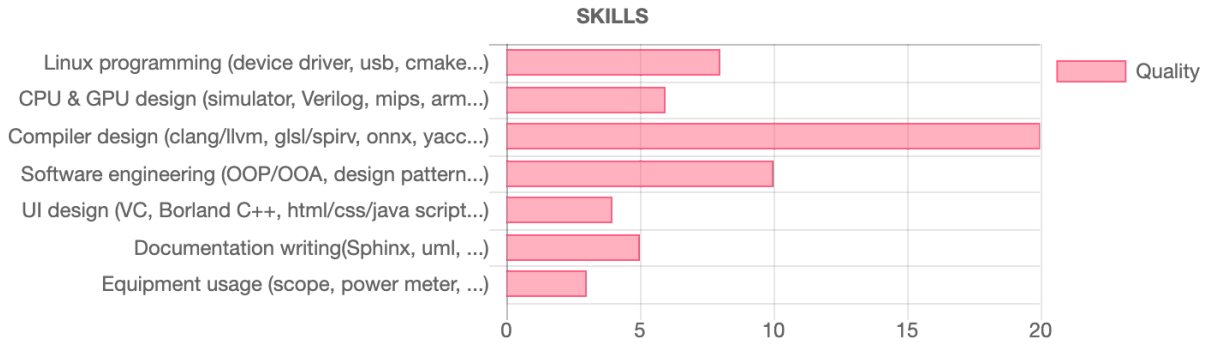
陈钟枢

我是一位编译器开发者，拥有在 LLVM CPU 和 GPU 后端、LLD 链接器、NPU/ONNX、C++、OpenGL/GLSL、模拟器等方面的扎实经验。我热衷于编译器及相关技术的开发工作。

履历

资格

二十年c/c++软体开发经验，十三年编译器相关工具开发经验，硕士时研究平行处理。



我的开放原始码专案

很高兴我的作品已被LLVM接受，出现在 http://llvm.org/docs/tutorial/#external-tutorials

如何建立LLVM后端编译器 http://jonathan2251.github.io/lbd/index.html

如何建立LLVM后端系统工具 http://jonathan2251.github.io/lbt/index.html

GPU编译器概念 http://jonathan2251.github.io/lbd/gpu.html

学历

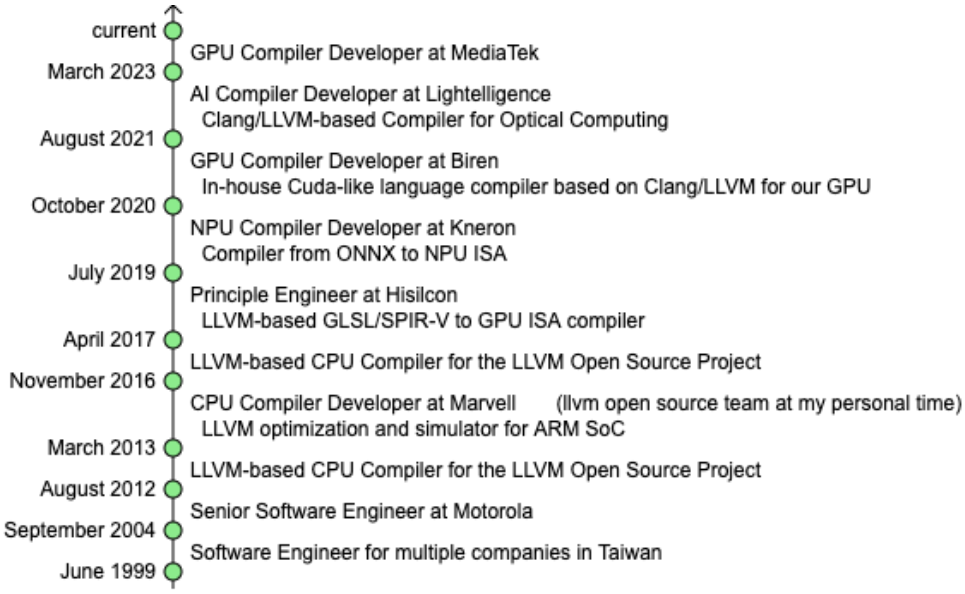
1997-1999 硕士班，国立台湾师范大学，台北，主修：资讯科学。

1991-1994 学士班，国立台湾科技大学，台北，主修：工业工程。

证照

1995年 国家高考（公职专业技师）资讯技师及格。

经验



硕士论文

The Researches of Column Sort and Related Problems

论文期刊：上述链结网页搜寻 "行排列法简化步骤之研究"

博士班研究计画

The Researches of Sorting Network and Related Algorithm

更详细的履历

其余作品

修影像处理课程与撰写:Jpeg decoder代码

网页与javascript: html简历 与 我個人網頁

Graphivz: 如此详细履历里的一些图学的图。原始码： mywork_1.gv and study_and_apply_ch2.gv

工作贡献

Lightelligence

实作TaskGraph并开发与TVM-compiler以及Runtime合作介面使得我们的平台可以支持深度学习的图编译。

实现 TaskGraph 并开发与 TVM-compiler 以及 Runtime 合作接口，使得我们的平台可以支持深度学习的图编译。

为 Lightelligence 的基于 RISC-V 的光学 NPU 开发后端编译器，内容包括：

1. 基于 GCC、LLVM 和 QEMU/Gem5 等开源项目，构建完整的 RISC-V 编译器工具链。评估 RISC-V 供应商并进行价格协商，利用我们内部从开源自行构建工具链的能力作为谈判优势。

2. 主导 Aurora 硬件产品的软件开发，并亲自负责编译器后端的程序设计。

3. 在 C++ 编译器中开发 TaskGraph 组件及其与Runtime模块的接口，实现我们平台上的深度学习计算图功能。

Biren

GPU tensor指令与usharpid处理。

GPU编译器优化与bug fix。

我们的Cude-like语言async{...}平行处理解法白皮书。

Kneron

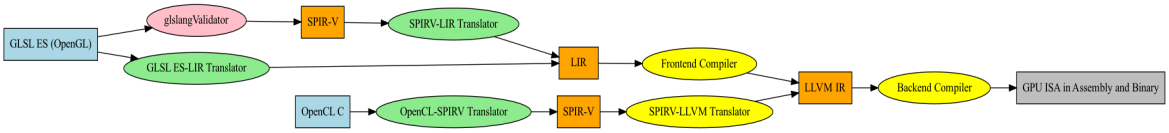
改写我们的NPU编译器上两层的IR中间码转换代码以提供共同的hardware independent图形数据结构, 以利多种NPU的支持。

支持加密格式的ONNX与config档输入。

确认如何支持MLIR。

Hisilcon

GPU编译器范围:



为支援自行设计的手机GPU, 移植ARM的code。如上图黄色部分：20%前端需修改, 50%后端需修改(以行数计算)。

贡献:

独立完成80% texture相关的API, 80 APIs totally here , (frontend + LLVM backend)与document撰写。

指导别的工程师完成其余20% texture相关的API, 核对并与texture的架构leader一起合作。

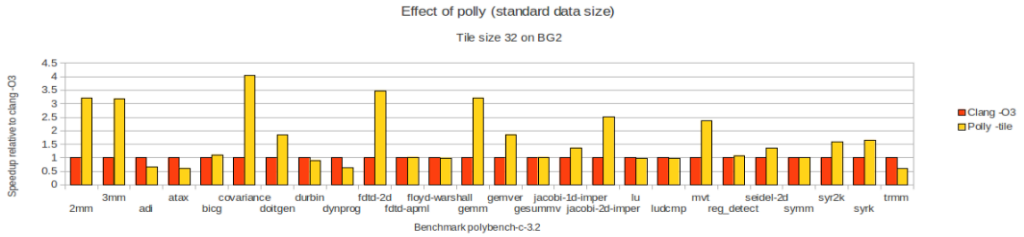
完成Prefetch-Sample optiomization, 让driver在载入GLSL bin与执行sampling指令前就可驱动2D sampling指令。

独立完成GPU对Vulkan load/store RGBA 固定浮点格式(32, 16, 11, 10 and 2 bits; NaN Infinity)支援的指令生成与document撰写。

Marvell

设计半自动的软体系统，自动执行用GCC编译器编译benchmark程式，并产生Excel比较图表。

为提升Marvell公司gcc与llvm编译器软体工具效能，介绍Polly软体系统。 Polly是针对loop最佳化的开放原始码专案。



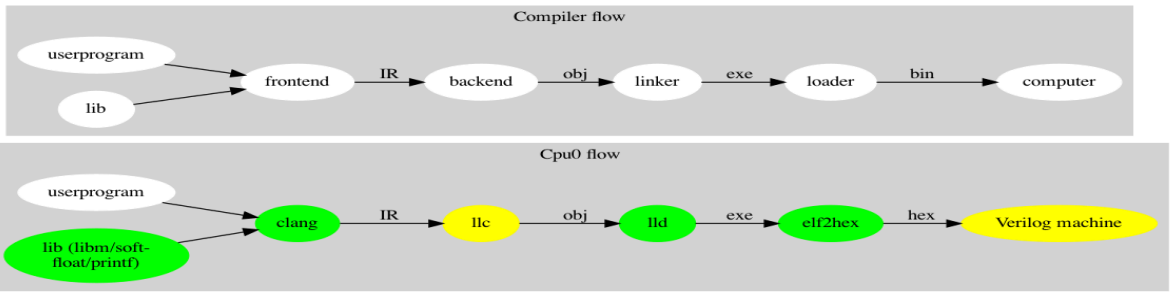
提出与实作DSL领域语言解决方案应用在ARM 64位元Csim上。

用cmake替代make于CSim上。

优点: 比make简洁与跨平台。

我的llvm开放原始码专案

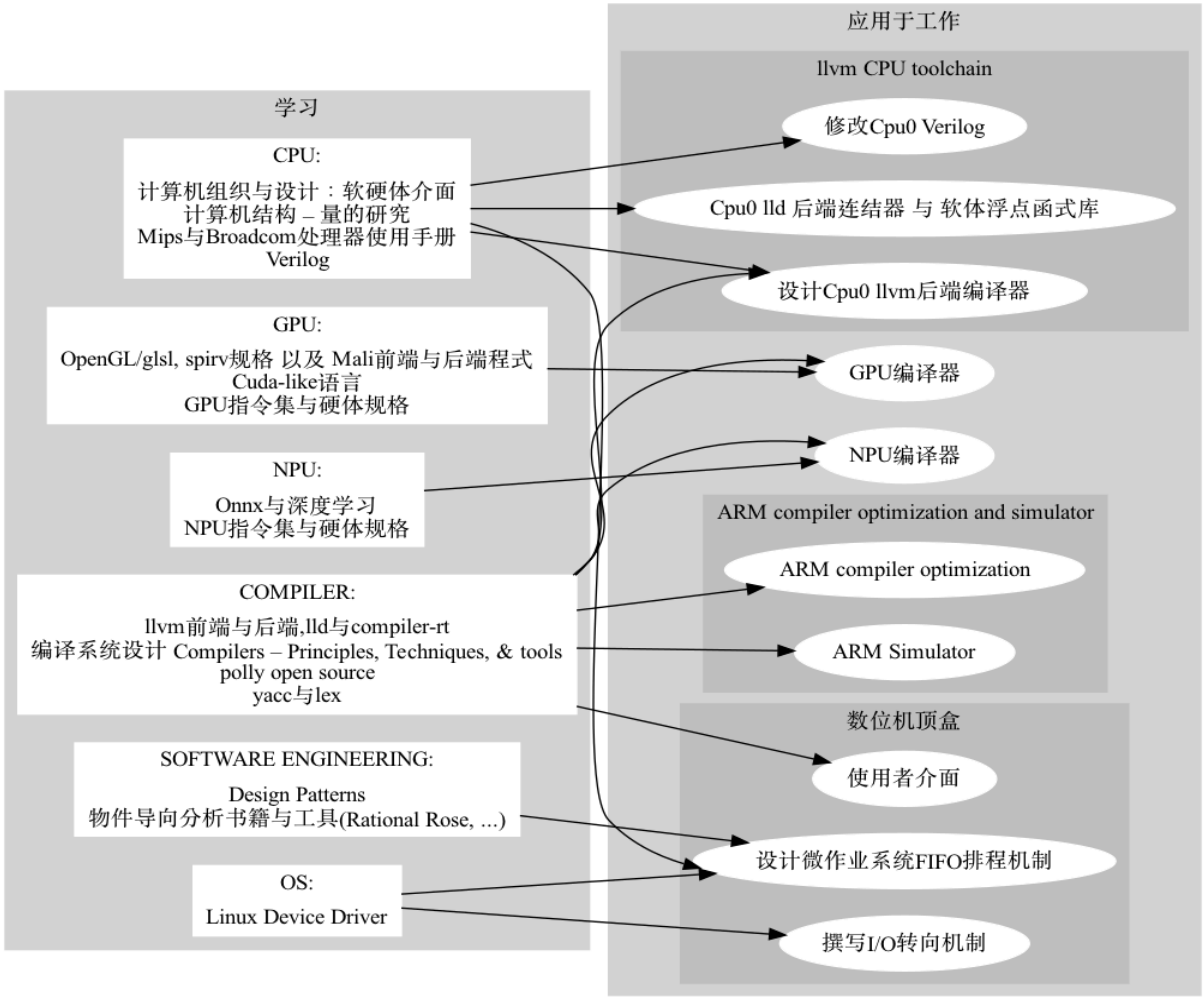
下半部是llvm的流程图。黄色与绿色分别是我书中（如上，我的开法原始码专案）。



Mortorola

开发数位机顶盒

出社会后的学习并运用于工作



推荐函

前主管推荐函: https://jonathan2251.github.io/ws/ch1/RL_Marvell.pdf