我是位有經驗的compiler開發者,開發過llvm cpu and gpu backend, lld linker, npu/onnx, c++, OpenGl/glsl compiler與simulator,...,對寫compiler感到快樂。 履歷 資格 二十年c/c++軟體開發經驗,十年編譯器相關工具開發經驗,碩士時研究平行處理。 **SKILLS** Linux programming (device driver, usb, cmake...) Quality CPU & GPU design (simulator, Verilog, mips, arm...) Compiler design (clang/llvm, glsl/spirv, onnx, yacc...) Compiler design (clang/llvm, glsl/spirv, onnx, yacc...) Quality: 20 Software engineering (OOP/OOA, design pattern...) UI design (VC, Borland C++, html/css/java script...) Documentation writing(Sphinx, uml, ...) Equipment usage (scope, power meter, ...) 10 15 20 我的開法原始碼專案 很高興我的作品已被LLVM接受,出現在 http://llvm.org/docs/tutorial/#external-tutorials 如何建立LLVM後端系統工具 **()** http://jonathan2251.github.io/lbt/index.html 學歷 1997-1999 碩士班,六月 1999,國立台灣師範大學,台北,主修:資訊科學。 1991-1994 學士班, 六月 1994, 國立台灣科技大學, 台北, 主修: 工業工程。 證照 1995年高考資訊技師及格。 經驗 current O GPU compiler developer for MediaTek March 2023 (Al compiler developer for Lightelligence clang/llvm compiler design for our Optical computer August 2021 GPU compiler developer for Biren Our designed Cuda-like language compiler based on clang/llvm for our GPU October 2020 O Seniror compiler developer for Kneron NPU compiler July 2019 🔷 Principle engineer for Hisilcon glsl/spirv frontend + LLVM backend compiler development for a whole new GPU ISA April 2017 (LLVM compiler November 2016 Senior software engineer for Marvell (Ilvm open source team at my personal time) Ilvm optimization and simulator for ARM SoC March 2013 () Programmer in IIvm open source team August 2012 🔘 Senior software engineer for Motorola September 2004 0 Software engineer for a few Taiwan's companies June 1999 🔷 碩士論文 博士班研究計畫 The Researches of Sorting Network and Related Algorithm The Researches of Column Sort and Related Problems 論文期刊: 上述鏈結網頁搜尋 "行排列法簡化步驟之研究" 其餘作品 修影像處理課程與撰寫: Jpeg decoder程式 網頁與javascript: html簡歷 與 我個人網頁 <u>Graphivz:</u> 如此詳細履歷裡的一些圖學的圖。原始碼: <u>mywork 1.gv</u> and <u>study and apply ch1.gv</u> 工作貢獻 Lightelligence 整理編譯gnu與llvm的RISCV開放原始碼工具鏈。 探查TVM, DPC++,SYCL/OpenCL可行性。 調查與評估符合我們光計算架構,供應商可提供的AI與科學計算應用的軟硬體。 領導 Aurora 硬件產品的軟件開發並自己編寫編譯器後端。 Biren Gpu tensor指令與usharpid處理。 Gpu編譯器最佳化與bug fix。 我們的Cude-like語言<u>async{...</u>}平行處理解法白皮書。 Kneron 改寫我們的NPU編譯器上兩層的IR中間碼轉換程式以提供共同的hardware independent圖形資料結構,以利多種NPU的支 持。 支持加密格式的ONNX與config檔輸入。 確認如何支持MLIR。 Hisilcon GPU編譯器範圍: GPU Online Compiler Flow llvm IR frontend(opengl, vulkan and opencl compiler) backend(llvm) linker 為支援自行設計的手機GPU, 移植ARM的code。20%前端需修改, 50%後端需修改(以行數計算)。 我的工作: 獨立完成80% texture相關的API與optimization (frontend + Ilvm backend)與document撰寫。 指導別的工程師完成其餘20% texture相關的API, 80 APIs totally here, 核對並與texture的架構leader一起合作。 完成Prefetch-Sample optiomization, 讓driver在載入glsl bin與執行sampling指令前就可驅動2D sampling指令。 獨立完成GPU對vulkan load/store RGBA 固定浮點格式(32, 16, 11, 10 and 2 bits; NaN Infinity)支援的指令生成與 document撰寫。 Marvell 設計半自動的軟體系統,自動執行用gcc編譯器編譯benchmark程式,並產生excel比較圖表。 為提升Marvell公司gcc與llvm編譯器軟體工具效能,介紹Polly軟體系統。Polly是針對loop最佳化的開放原始碼專案。 Effect of polly (standard data size) Tile size 32 on BG2 Clang -O3 Polly -tile 提出與實作DSL領域語言解決方案應用在ARM 64位元Csim上。 用cmake替代make於Csim上。 優點: 比make簡潔與跨平台。 我的llvm開放原始碼專案 下半部是Ilvm的流程圖。黃色與綠色分別是我書中(如上,我的開法原始碼專案)。 Compiler flow userprogram bin obj lib Cpu0 flow userprogram Verilog machine lib (libm/softfloat/printf) Mortorola 開發數位機上盒 出社會後的學習並運用於工作 應用於工作 llvm CPU toolchain 學習 修改Cpu0 Verilog CPU: 計算機組織與設計: 軟硬體介面 Cpu0 lld 後端連結器 與 軟體浮點函式庫 計算機結構 - 量的研究 Mips與Broadcom處理器使用手冊 Verilog 設計Cpu0 llvm後端編譯器 NPU: Onnx與深度學習 NPU編譯器 NPU指令集與硬體規格 GPU: GPU編譯器 OpenGL/glsl, spirv規格 以及 Mali前端與後端程式 Cuda-like語言 ARM compiler optimization and simulator GPU指令集與硬體規格 ARM compiler optimization COMPILER: llvm前端與後端,lld與compiler-rt ARM Simulator 編譯系統設計 Compilers – Principles, Techniques, & tools polly open source yacc與lex 數位機上盒 SOFTWARE ENGINEERING: 使用者介面 Design Patterns 物件導向分析書籍與工具(Rational Rose, ...) 設計微作業系統FIFO排程機制 OS: Linux Device Driver

陳鍾樞

撰寫I/O轉向機制