TÉCNICO LISBOA

Linux capable RISC-V CPU for IOb-SoC

Pedro Nuno de Melo Antunes

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. José João Henriques Teixeira de Sousa

Examination Committee

Chairperson: ...
Supervisor: ...
Member of the Committee: ...

September 2022

Declaration

I declare that this document is an original work of my own authorship and that it fullfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

A few words about the university, financial support, research advisor, dissertation readers, faculty or other professors, lab mates, other friends and family...

Resumo

Com o avanço das tecnologias desenvolvidas em código aberto torna-se necessário estudar tanto o novo hardware desenvolvido como o software que tira partido deste novo hardware. Nesta dissertação de mestrado pretende-se conseguir correr um sistema operativo baseado em Linux numa variante do *IOb-SoC*. Ao longo deste trabalho à de-se realizar a implementação de um processador *RISC-V* de *32-bits* capaz de correr o Linux no *IOb-SoC*. No final desta tése pretende-se: em primeiro lugar, ser capaz de correr uma simulação do sistema criado, que mostre o seu correto funcionamento; e em segundo lugar, implementar a variante do IOb-SoC desenvolvida numa FPGA e a partir desta FPGA correr o Linux.

Palavras-chave: RISC-V, Linux, Sistema num Chip (SoC), Verilog

Abstract

With the advances in new open-source technologies, it's imperial that the new hardware solutions and software implementation on the new hardware is studied. The aim of this thesis is to successfully run a Linux based OS (Operative System) on an IOb-SoC variant. During this work, the implementation of a 32-bit RISC-V CPU capable of running Linux on the Iob-SoC is going to be developed. At the end of this thesis, it's expected to: firstly, be able to run a simulation of the SoC (System on Chip) used to run the Linux kernel and verify its correct functionality; and secondly, implement the IOb-SoC variant developed in an FPGA and successfully boot Linux.

Keywords: RISC-V, Linux, Systems on Chip (SoC), Verilog

Contents

List of Tables

List of Figures

Listings

List of Acronyms

AGU Address Generator Unit

ALU Arithmetic and Logic Unit

AP Average Precision

API Application Programming Interface

AXI Advanced eXtensible Interface

BRAM Block RAM

CISC Complex Instruction Set Computer

CM Configuration Module

CPU Central Processing Unit

DMA Direct Memory Access

DSP Digital Signal Processing

FF Flip-Flop

FM Feature Map

FP Fixed-Point

FPGA Field Programmable Gate Array

FPS Frames Per Second

FPU Floating-point unit

FU Functional Unit

GPP General-Purpose Processor

GPU Graphical Processing Unit

IOb-SoC IObundle SoC

IoU Intersection over Union

IPv4 Internet Protocol

LRU Least Recently Used

LUT Look-Up Table

MMU Memory Management Unit

PE Processing Element

RAM Random Memory Access

ReLU Rectified Linear Unit

RISC Reduced Instruction Set Computer

ROM Read Only Memory

SFP Static Fixed Point

SIMD Single Instruction Multiple Data

SoC System on a Chip

SRAM Static Random Access Memory

UART Universat Asynchronous Receiver-Transmitter

Chapter 1

Introduction

Introdução

1.1 Motivation

Motivação

1.2 Objectives

Objectivo

1.3 Thesis Outline

Estutura do documento

Chapter 2

Linux on RISC-V Background

Things I need to talk about:

- To do

Chapter 3

Conclusions

Conclued

3.1 Achievements

Talk about what was achived

3.2 Future Work

Talk about what can be improved in future works

Bibliography

[1] M. A. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2015.

[2] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, pp. 541–551, Dec 1989.

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of IEEE*, vol. 86, pp. 2278 – 2324, November 1998.

[4] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," *Proceedings of IEEE*, vol. 105, pp. 2295 – 2329, December 2017.

[5] K. Abdelouahab, M. Pelcat, J. Sérot, and F. Berry, "Accelerating CNN inference on FPGAs: A Survey," *CoRR*, vol. abs/1806.01683, 2018.

[6] Y. Ma, Y. Cao, S. Vrudhula, and J. Seo, "Optimizing the Convolution Operation to Accelerate Deep Neural Networks on FPGA," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, pp. 1354–1367, July 2018.

[7] K. Guo, S. Zeng, J. Yu, Y. Wang, and H. Yang, "[DL] A Survey of FPGA-based Neural Network Inference Accelerators," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 12, pp. 2:1–2:26, Mar. 2019.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, (USA), pp. 1097–1105, Curran Associates Inc., 2012.

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[10] K. Guo, L. Sui, J. Qiu, S. Yao, S. Han, Y. Wang, and H. Yang, "Angel-Eye: A Complete Design Flow for Mapping CNN onto Customized Hardware," in *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pp. 24–29, July 2016.

[11] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *CoRR*, vol. abs/1506.02640, 2015.

[12] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *arXiv preprint arXiv:1612.08242*, 2016.

[13] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[14] J. Redmond, "darknet." `https://github.com/pjreddie/darknet`, 2018.

[15] I. Lda, "IOb-SoC." `https://github.com/IObundle/iob-soc`, 2020.

[16] C. Wolf and et. al., "PicoRV32 - A Size-Optimized RISC-V CPU." `https://github.com/cliffordwolf/picorv32`, 2019.

[17] K. Cheng and et. al., "RISC-V GNU Compiler Toolchain." `https://github.com/riscv/riscv-gnu-toolchain`, 2020.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *CoRR*, vol. abs/1512.03385, 2015.

[19] A. Kathuria, "What's new in YOLOv3?." `https://medium.com/analytics-vidhya/yolo-v3-theory-explained-33100f6d193`, Visited in 22 Nov 2019, Apr. 2018.

[20] J. Hui, "mAP (mean Average Precision) for Object Detection." `https://medium.com/@jonathan_hui/map-mean-average-precision-for-object-detection-45c121a31173`, Visited in 7 Jan 2020, Mar. 2018.

[21] A. Vidhya, "Yolov3 theory explained." `https://medium.com/analytics-vidhya/yolo-v3-theory-explained-33100f6d193`, Visited in 22 Nov 2019, July 2019.

[22] T.-J. Yang, Y.-H. Chen, and V. Sze, "Designing Energy-Efficient Convolutional Neural Networks Using Energy-Aware Pruning," pp. 6071–6079, 07 2017.

[23] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," *CoRR*, vol. abs/1405.0312, 2014.

[24] Yufei Ma, N. Suda, Yu Cao, J. Seo, and S. Vrudhula, "Scalable and modularized RTL compilation of Convolutional Neural Networks onto FPGA," in *2016 26th International Conference on Field Programmable Logic and Applications (FPL)*, pp. 1–8, Aug 2016.

[25] M. Lin, Q. Chen, and S. Yan, "Network In Network," 2013.

[26] M. Wijtvliet, L. Waeijen, and H. Corporaal, "Coarse grained reconfigurable architectures in the past 25 years: Overview and classification," in *2016 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation (SAMOS)*, pp. 235–244, July 2016.

[27] V. Mário, "Deep Versat: A Deep Coarse Grain Reconfigurable Array," Master's thesis, Instituto Superior Técnico, November 2019. Master's Thesis.

[28] A. Charana, "Development Environment for a RISC-V Processor." Master's Thesis, July 2020.

[29] J. Roque, "Development Environment for a RISC-V Processor: Cache." Master's Thesis, Jan 2021.

[30] D. Pestana, "Object Detection and Classification on the Versat Reconfigurable Processor." Master's Thesis, Jan 2021.

[31] J. D. Lopes, "Versat, a compile-friendly reconfigurable processor-architecture," Master's thesis, Instituto Superior Técnico, November 2017.

[32] J. Redmon, "Darknet: Open Source Neural Networks in C." `http://pjreddie.com/darknet/`, 2013–2016.

[33] J. Redmon, "YOLO: Real-Time Object Detection." `http://pjreddie.com/darknet/yolo/`, visited in 20 Nov 2020, 2018.

[34] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CoRR*, vol. abs/1311.2524, 2013.

[35] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015.

[36] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[37] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature Pyramid Networks for Object Detection," *CoRR*, vol. abs/1612.03144, 2016.

[38] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," *CoRR*, vol. abs/1708.02002, 2017.

[39] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv 1409.1556*, 09 2014.

[40] C. Zhang, D. Wu, J. Sun, G. Sun, G. Luo, and J. Cong, "Energy-Efficient CNN Implementation on a Deeply Pipelined FPGA Cluster:," pp. 326–331, 08 2016.

[41] N. Suda, V. Chandra, G. Dasika, A. Mohanty, Y. Ma, S. Vrudhula, J.-s. Seo, and Y. Cao, "Throughput-Optimized OpenCL-based FPGA Accelerator for Large-Scale Convolutional Neural Networks," pp. 16–25, 02 2016.

[42] J. Qiu, S. Song, Y. Wang, H. Yang, J. Wang, S. Yao, K. Guo, B. Li, E. Zhou, J. Yu, T. Tang, and N. Xu, "Going Deeper with Embedded FPGA Platform for Convolutional Neural Network," pp. 26–35, 02 2016.

[43] G. Cybenko, "Approximation by superpositions of a sigmoidal function. Math Cont Sig Syst (MCSS) 2:303-314," *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 303–314, 12 1989.

[44] J. Qiu, S. Song, Y. Wang, H. Yang, J. Wang, S. Yao, K. Guo, B. Li, E. Zhou, J. Yu, T. Tang, and N. Xu, "Going Deeper with Embedded FPGA Platform for Convolutional Neural Network," pp. 26–35, 02 2016.

[45] E. Nurvitadhi, S. Subhaschandra, G. Boudoukh, G. Venkatesh, J. Sim, D. Marr, R. Huang, J. Hock, Y. Liew, K. Srivatsan, and D. Moss, "Can FPGAs Beat GPUs in Accelerating Next-Generation Deep Neural Networks?," pp. 5–14, 02 2017.

[46] T. Fujii, S. Sato, H. Nakahara, and M. Motomura, "An FPGA Realization of a Deep Convolutional Neural Network Using a Threshold Neuron Pruning," pp. 268–280, 03 2017.

[47] I. Tsmots, O. Skorokhoda, and V. Rabyk, "Hardware Implementation of Sigmoid Activation Functions using FPGA," in *2019 IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM)*, pp. 34–38, 2019.

[48] A. Ahmad, M. A. Pasha, and G. J. Raza, "Accelerating Tiny YOLOv3 using FPGA-Based Hardware/-Software Co-Design," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2020.

[49] Z. Yu and C. Bouganis, *A Parameterisable FPGA-Tailored Architecture for YOLOv3-Tiny*, pp. 330–344. 03 2020.

[50] S. Oh, J. H. You, and Y. K. Kim, "Implementation of Compressed YOLOv3-tiny on FPGA-SoC," in *2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia)*, pp. 1–4, 2020.

[51] ARM, *AMBA AXI and ACE Protocol Specification*. ARM, February 2013.

[52] Xilinx, *UltraScale Architecture-Based FPGAs Memory IP v1.4*. Xilinx, June 2020.

[53] Xilinx, *AXI Interconnect v2.1*. Xilinx, December 2017.

[54] Xilinx, *Zynq UltraScale+ MPSoC Data Sheet v1.8*. Xilinx, October 2019.

[55] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 2^{nd} ed., 2006. ISBN:978-0387303031.

[56] A. Jameson, N. A. Pierce, and L. Martinelli, "Optimum aerodynamic design using the Navier–Stokes equations," in *Theoretical and Computational Fluid Dynamics*, vol. 10, pp. 213–237, Springer-Verlag GmbH, Jan. 1998.

[57] A. C. Marta, C. A. Mader, J. R. R. A. Martins, E. van der Weide, and J. J. Alonso, "A methodology for the development of discrete adjoint solvers using automatic differentiation tools," *International Journal of Computational Fluid Dynamics*, vol. 99, pp. 307–327, Oct. 2007. doi:10.1080/10618560701678647.