PERFORMANCE EVALUATION OF STARFIVE VISIONFIVE V2 AND COMPARISON ON INTEL UP BOARD AND RASPBERRY PI 4B

BY

LOH LE ZHOU

A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF COMPUTER SCIENCE (HONOURS)
Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2024

REPORT STATUS DECLARATION FORM

Title: **PERFORMANCE OF STARFIVE VISIONFIVE V2 AND COMPARISON ON INTEL UP BOARD AND RASPBERRY PI 4B**

Academic Session: JAN 2024

I, LOH LE ZHOU declare that I allow this Final Year Project Report to be kept in Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:

7-08, Berjaya Condominium,

Jalan Berjaya, Kampung Baru,

14000, Bukit Mertajam,

Pulau Pinang.

Wong Chee Siang

Supervisor's name

Date: 20/4/2024

Date: 23/4/2024

Universiti Tunku Abdul Rahman			
Form Title : **Sample of Submission Sheet for FYP/Dissertation/Thesis**			
Form Number: **FM-IAD-**	Rev No.: **0**	Effective Date: **21 JUNE**	Page No.: **1 of 1**

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 20/4/24

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ______***Loh Le Zhou***______________________________ (ID No: __***2101871***) has completed this final year project/ dissertation/ thesis* entitled "Performance Evaluation of StarFive VisionFive V2 and Comparison on Intel UP Board and Raspberry Pi 4B" under the supervision of Ts. Wong Chee Siang (Supervisor) from the Department of Computer Science, Faculty/Institute* of Information and Communication Technology.

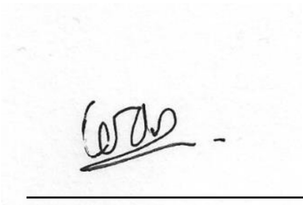
I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

LOH LE ZHOU

DECLARATION OF ORIGINALITY

I declare that this report entitled "**PERFORMANCE EVALUATION OF STARFIVE VISIONFIVE V2 AND COMPARISON ON STARFIVE VISIONFIVE V1 AND INTEL UP BOARD**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.



Signature : ____________________________

Name : _______Loh Le Zhou________

Date : _______20/4/2024____________

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Ts. Wong Chee Siang who has given me this bright opportunity to engage in a single board computer benchmark project. It is my first step to establish a career in Single Board Computer field. A million thanks to you.

I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

RISC-V is an open instruction set architecture that defines the set of instructions that a machine can understand. Since it is an open ISA, RISC-V is free to implement without having to pay royalties. VisionFive 2 is a RISC-V single board computer with an integrated GPU that has improved processor work frequency and multimedia processing capabilities compared to its previous version, VisionFive 1. VisionFive has been one of the competitors of Raspberry Pi [1]. Over the span of years, users have been comparing different coprocessors to understand the relative performance of thesedevices. This can help them understand which coprocessor is the best for a specific workload. By comparison and performance benchmark users can also find potential improvements to the device. PTS benchmarking also known as Phoronix Test Suite benchmarking is an open-source benchmarking software that is widely used to assess performance of Linux systems [2]. This thesis aims to perform PTS benchmarking to measure and evaluate performance of Starfive Visionfive 2, Starfive Visionfive 1 and Raspberry Pi 4B.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT	Internet of Things
ISA	Instructions Set Architectures
RPi	Raspberry Pi
GCC	GNU Compiler Collection
PTS	Phoronix Test Suite
RISC	Reduced Instruction Set Computing
MFLOPS	Million Floating Point Operation Per Second
TPS	Transactions Per Second
MP/s	Megapixels Per Second
HDMI	High-Definition Multimedia-Interface
SSH	Secure Shell
QEMU	Quick Emulator
LLVM	Low-level Virtual Machine
IoT	Internet of Things

Chapter 1

Introduction

In this chapter, we present the background and motivation of our research topic, problem statement, our contributions to the field, and the outline of the thesis.

1.1 Project Background

VisionFive, built on the RISC-V architecture, takes open-source to the next level by giving developers more freedom and ability to create and construct industry-leading solutions [3]. Compared to ISA such as x86-64 that are proprietary, RISC-V has potential to improve as open-source tends to have an advantage on the flexibility of development processes, more iteration and lower cost. VisionFive 2 is a development board that aims to achieve high performance with low-cost and it is a full open-source RISC-V single board computer [3]. VisionFive 2 is equipped with JH7110, which is a powerful system-on-chip (SoC) that has greater power efficiency. It is also equipped with the IMG-BXE-4-32-MC1 graphics processor which supports the newest API trends, including Vulkan 1.3, and is optimized for RISC-V as the target application processor.

In the past, comparison between these processors has also been done to evaluate their interoperability and perform standardization to benchmark these processors. As application of the RISC-V architecture starts to increase, it bears valuable insight and great potential as a research project. This thesis will be using PTS benchmark to evaluate and compare performance of VisionFive 2, VisionFive 1, Raspberry Pi 4b and the Intel UP Board.

1.2.1 Problem Statement

There is a substantial knowledge gap about the performance capabilities and constraints of embedded computer systems such as Raspberry Pi, Intel UP Board and notably in the context of the StarFive VisionFive development board, where its ISA infrastructure is significantly different than other ISAs in the market. Although these single board computers are a strong platform for numerous applications like computer vision, artificial intelligence, and robotics, neither academics nor developers have access to extensive performance testing data. Hence, by providing quantitative insights about the VisionFive board, users can utilize it effectively and creates potential for innovative projects for diverse industries.

Currently, existing documentation lacks in-depth analysis and just provides summary performance numbers for different workloads and circumstances. As a result, developers can encounter confusion when deciding which use cases are suited for the VisionFive board, which results in less-than-ideal application implementations and untapped performance potential.

Compiler flags play a pivotal role in influencing code generation, optimization, and execution on a hardware platform. Currently, there exists a gap in the understanding of which compiler flags yield the best performance outcomes for SiFive-based systems. This shortcoming hinders effective software creation and prohibits programmers from making the most of the SiFive architecture's potential.

Next, measuring power consumption and efficiency of SiFive processors under various workloads and operational circumstances can bring insight into building sustainable solutions using RISC-V architecture boards.

In the machine learning (ML) applications field, the lack of comprehensive benchmarks statistics related to RISC-V architecture Single Board Computers (SBCs) is a significant issue. By conducting experiments in the field, we can achieve a better understanding of hardware selection, algorithm optimization, and system design for ML workloads on RISC-V platforms.

The problem statement of this project is to explore the capabilities and limitations of VisionFive 2 through benchmarks and comparison of the benchmarks on various embedded boards. Performance metrics such as CPU, memory, GPU, cache, multimedia encoding, machine learning, filesystem, database, python, web server and computational fluid dynamics performance as well as power consumption and different compiler flag optimizations options will be measured. These procedures will identify the strengths and weaknesses of this board, enabling us to realize the potential and weaknesses of the board.

1.3 Motivation

The motivation for this study derives from the growing importance of open-source hardware and the requirement for comprehensive performance evaluation in embedded computer systems. As VisionFive 2 represents a substantial improvement in RISC-V architecture, it is critical to fully understand its strengths and limits. We hope to give developers and academics with useful insights by doing testing and comparisons with other prominent embedded boards such as the Raspberry Pi 4B and Intel UP.

Furthermore, our evaluation of compiler flag optimizations and power consumption measurements adds to our research by providing insights into improving system performance and energy efficiency for SiFive-based devices. This component of our research helps to advance sustainable computing techniques. Moreover, our assessment of performance in machine learning applications enables some insights and better system design optimization for machine learning workloads.

Over the course of decades, instruction sets have been constantly changing, and the fundamental issue is that firms have been relying on ISAs held by companies [4]. As a consumer, we don't have a lot of choice in making technological decisions, which puts us in a bind. RISC-V can overcome this issue due to its flexibility and diversity when compared to proprietary ISAs.

By improving this board, it becomes a step closer to showcase to the market and draw small businesses into implementing RISC-V architecture since it has a low cost and fordevelopers to construct systems that have is closer to client requirements.

1.4 Project Scope

The project's main output will be a thorough examination of power consumption, covering single-thread to four-thread workloads and measuring the energy requirements of the CPU at various degrees of computational intensity. The project will also examine and record the effects of compiler flag settings, including, among others, the parameters -O1, -O2, -O3, -march=rc64gc, and -mcpu=sifive-u74. The Phoronix Test Suite benchmarks will be included, along with additional selected benchmarks, to further improve the project's thorough evaluation of SiFive processor performance.

Some tests and their performance metrics [17] :

- AOBench: Render time (lower is better), frames per second (higher is better).
- CacheBench: Memory read/write latency, memory bandwidth.
- Coremark: Total number of iterations per second (higher is better).
- Dolfyn: Floating-point operations per second (FLOPS), simulation time.

- FFTE: Execution time (lower is better), memory bandwidth.
- FLAC: Encoding/decoding speed (higher is better).
- GraphicsMagick: Image processing time, operations per second.
- Himeno Benchmark: Execution time (lower is better), performance score.
- Perl Benchmark: Time taken for various Perl language operations.
- PostMark: Transactions per second, throughput, latency.
- Tinymembench: Memory read/write latency, memory bandwidth.
- WebP Image Encode: Encoding time for WebP images.
- t-test1: Execution time (lower is better), memory usage.
- PyPerformance: Time taken for various Python language operations.
- PHPBench: Time taken for various PHP language operations.
- Numpy: Execution time for various numerical operations.
- OpenCV: Image processing capabilities.
- NCNN: Measures the performance and efficiency of neural network inference tasks.

1.5 Project Objectives

Performance assessment using the Phoronix Test Suite, the project aims to carry out an exhaustive assessment of the SiFive processor's performance. To gain a comprehensive understanding of the processors' capabilities, highlight their strengths, and pinpoint areas for development, this review includes testing across a variety of workloads.

Power consumption test scaling from single thread to four threads, the project seeks to examine power consumption, concentrating on both real-time power fluctuations and customary static data. The initiative intends to provide practical insights to industries for improving power efficiency across various workloads or projects by evaluating power usage trends.

Testing different compiler flag optimization settings, through compilation and testing, the project seeks to determine the best compiler flag settings for SiFive CPU, especially RISC-V compilation flag settings. The goal is to identify combinations that improve performance by analyzing various flag settings and different workloads.

1.6 Contributions

The project's integration of power consumption testing, compiler flag evaluations, and benchmarking using the Phoronix Test Suite has important ramifications for the field of computers and technology. This initiative has the potential to alter how we approach hardware and software optimization, influencing the future of computer systems in an era where resource efficiency and sustainable practices are at the forefront of technological growth.

The techniques used to evaluate the performance of SiFive processors is the Phoronix Test Suite. This is advantageous for academics and developers looking to improve programs and opens the door for more transparent and knowledgeable processor selection, which eventually leads to increased system efficiency and lower energy usage.

In addition, power consumption test using a real time power meter to measure the average energy consumed on workloads scaling from single thread to four threads can help users comprehend how SiFive processors use power under diverse workloads and their power consumption patterns. The project's emphasis on power efficiency is fully in line with global initiatives to reduce energy use and promote environmentally friendly and sustainable computing methods.

Through performance benchmarking and optimization, developers and engineers can construct powerful and efficient embedded vision and AI systems, such as robotics and IoT devices by utilizing the integrated GPU of the VisionFive V2. Findings in this research can create a substantial impact in fields including industrial automation, healthcare, security and more. By improving this board, VisionFive 2 can compete with other RISC-V architectures, as a result creates additional options for computer vision and AI applications.

1.7 Background Information

An understanding of an instruction set architecture (ISA) is crucial to understanding the relevance of RISC-V. ISA defines a processor design and specifies how instructions are encoded and performed. Historically, the computing industry has been dominated by processor architectures like x86 and ARM. In a UC Berkeley research project, the researchers developed a reliable ISA, and is now run by RISC-V International. Contrary to proprietary ISAs, RISC-V is open source and values collaboration and openness, enabling programmers to modify processors to suit requirements. In addition, it encourages innovation and cooperation between businesses, researchers, and enthusiasts.

The RISC-V ecosystem includes a wide range of applications, from embedded devices to supercomputers, opening the door for processor architectures that are optimized for purposes.

The Phoronix Test Suite is a powerful tool for evaluating hardware and software performance concurrently with the RISC-V revolution. By providing a comprehensive set of tests that cover many facets of system performance, from compute throughput to graphics rendering, the Phoronix Test Suite streamlines benchmarking. As part of the project, the VisionFive 2 board's RISC-V processors will be rigorously benchmarked using the Phoronix Test Suite. These benchmarks offer numerical understandings of the processor's performance under various workloads and conditions. Power consumption test will be performed by running tests in phoronix test suite while using a power meter to measure.

The project also investigates compiler flag optimizations. Compilation flags control how source code is converted into machine code, affecting things like memory utilization and execution speed. Compilation flags are a crucial factor to consider because they have a substantial impact on performance during software development.

Chapter 2

Literature Review

2.1 Previous Works on Performance Benchmarking RISC-V processor using CoreMark

CoreMark benchmark is a CPU benchmark that exercises the hardware by running programs. It uses synthetic benchmarks, which are code developed for the specific benchmark. Compared to Dhyrstone, it has the similar tests except more additional tests were done by using CoreMark. One of the past research projects has used this benchmark to evaluate the Wally RISC-V processor [5]. This method evaluates the processor performance according to a scoring called CoreMark Scoring, it stands for the number of iterations completed per second when running the benchmark. CoreMark benchmark measures performance of a processor core by running basic arithmetic and data operations. Hence, the operations in the program that are used for testing includes finding and sorting the linked list (memory test), matrix manipulation (multiplier and adder test), State machine (test branch logic) and CRC (exercises general processor).

After compiling the program, the memory file is loaded into the processor and the workload is simulated using ModelSim. The measured performance is 2.42 CM/MHZ and the cycles/instruction (CPI) before cache implementation is 1.69 CM/MHZ [5]. The outcome of the benchmark is they were able to identify bugs in the system and implemented improvements through bug fixes and microarchitectural optimizations. However, the Wally RISC-V processor has not been able to reach the average score of a standard level processor which is 4.42 CM/MHz, and some performance bottlenecks were remained unsolved.

Previously, Starfive officially released their CPU performance benchmark results on VisionFive 2 using CoreMark and Dhrystone, achieving a score of 5.09 CM/MHz on CoreMark and 2.64 CM/MHz on Dhrystone [6].

2.2 Previous Works on Performance Benchmarking RISC-V processor using QEMU Virtual Platform Tool

For this research, ten different benchmarks were done to test the performance of RISC-V processors on QEMU [7]. QEMU are based on two different processors that runs on Ubuntu 18.04. The source file type used are C and CPP, while the necessary ISA consists of Integer, Multiply and Floating-point. The benchmarks done are as follows: Fibonacci sequence, Dhrystone, Linpack (test floating-point performance of comptuer system), Speed of basic mathematical operations, Matrix multiplication, Signal filter (test digital signal processing performance), Simple neural network, Image process, Fast Fourier transform (test performance of executing a quick Fourier transform algorithm) and Data compression.

Platforms available that are based on RISC-V that can be used as a "virt" platform to run tests are Microchip PolarFire SoC Icicle Kit, Shakti C Reference Platform, SiFive HiFive Unleashed and Generic Virtual Platform [7]. However, the source code of these platforms lacks a cache design.

They also attempted to run the benchmarks on QEMU without an operating system. However, there are errors that can occur when address is larger then 0x8000_0000, hence the article set the actual address of memory at 0x8200_0000 [7]. Also, the run time of the benchmark was not tested as they have not found a method to test it. After that, they used a debugger named GNU symbolic debugger as a test tool.

The benchmarks are put in the Linux system simulated by QEMU, and the results were measured using the RV32GC or RV64GC platform [7]. The table below is their Dhrystone results.

Dhrystone				
ISA	**User-mode**	**System-mode**	**Instructions**	**Speed[MIPS]**
RV64I	58.989s	77.53s	34,650,003,031	446.924
RV64IM	50.060s	78.29s	34,650,003,031	442.585
RV64IMFD	49.945s	82.15s	34,650,003,031	421.789
RV32IM	31.804s	68.45s	32,450,004,233	474.069
RV32IMFD	30.404s	71.89s	32,450,004,233	451.384

Figure 2. 1 Dhrystone results on RV34GC and RV64GC ISA

2.3 Previous Work on Performance Benchmarking RISC-V processor StarFive VisionFive 2

According to a review by Jeff Geerling, the firmware had to be upgraded by running a special buildroot image and logging in with SSH to install the latest OS [8]. A bug was also found on the HDMI, as the GPU should be able to handle 4K, but it was not working with monitor. Eventually he had to use Atomos Ninja V, a high-quality monitor and recorder with the size of a handheld camera. Jeff suggested that the test results may vary in the future due to the newness of RISC-V platform. It resulted a single-core score of 78 and a multi-core score of 276 [9]. The figure below shows the Geekbench 5 scores for VisionFive 2 compared to Raspberry Pi 4 and Raspberry Pi 3 B+.

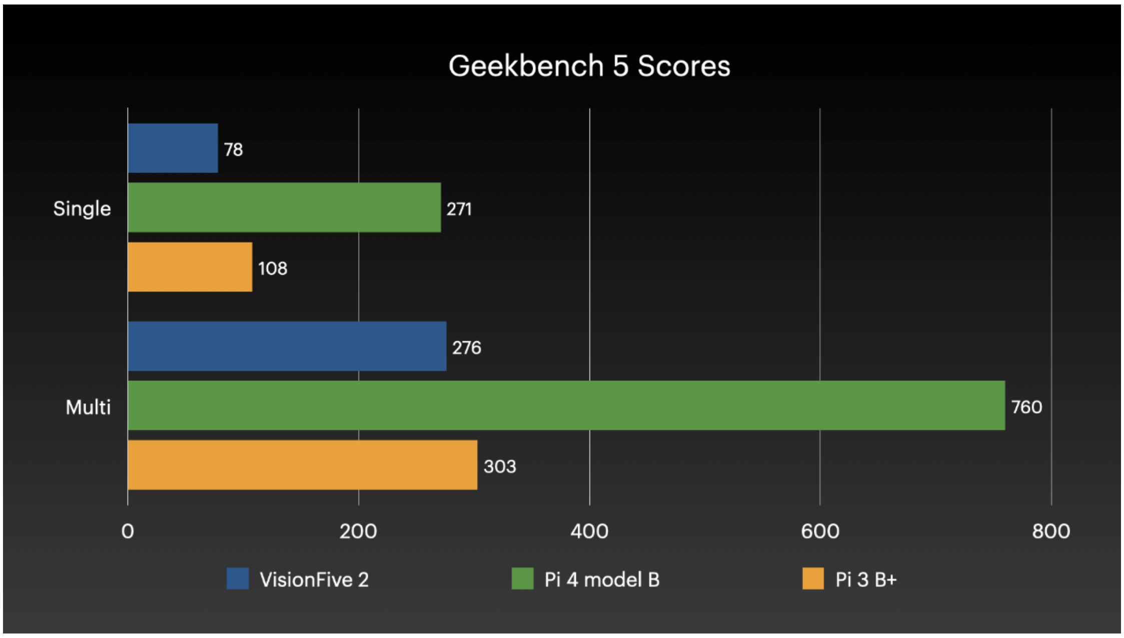


Figure 2. 2. Geekbench 5 scores and comparison of VisionFive 2 with Raspberry Pi 4 model B and Pi 3 B+

It shows that VisionFive 2 is significantly slower. The results further shows that VisionFive 2 scored low on image processing tests. However, the model used for the image-based test is old and not optimized for the chip on the board [8]. Linpack test on floating point performance failed due to problems with compiling the Python cryptography library. IO performance of M.2 slot was measured at a speed of only

250MB/sec, whereas the built in microSD card slot was measured at a speed of 24MB/sec, which is slower than Pi 4. One of the highlights of VisionFive 2 is that it is

the first RISC-V computer with an intergrated GPU, however experience from users show that Single Board Computers can hardly get GPU to do anything in Linux, and the results show just that. However, there are users that already posted about succeding to get the AMD graphics card to work by using a PCle x16 adapter. The power consumption/efficiency is measured to be using 3W at idle and 5W fully stressed.

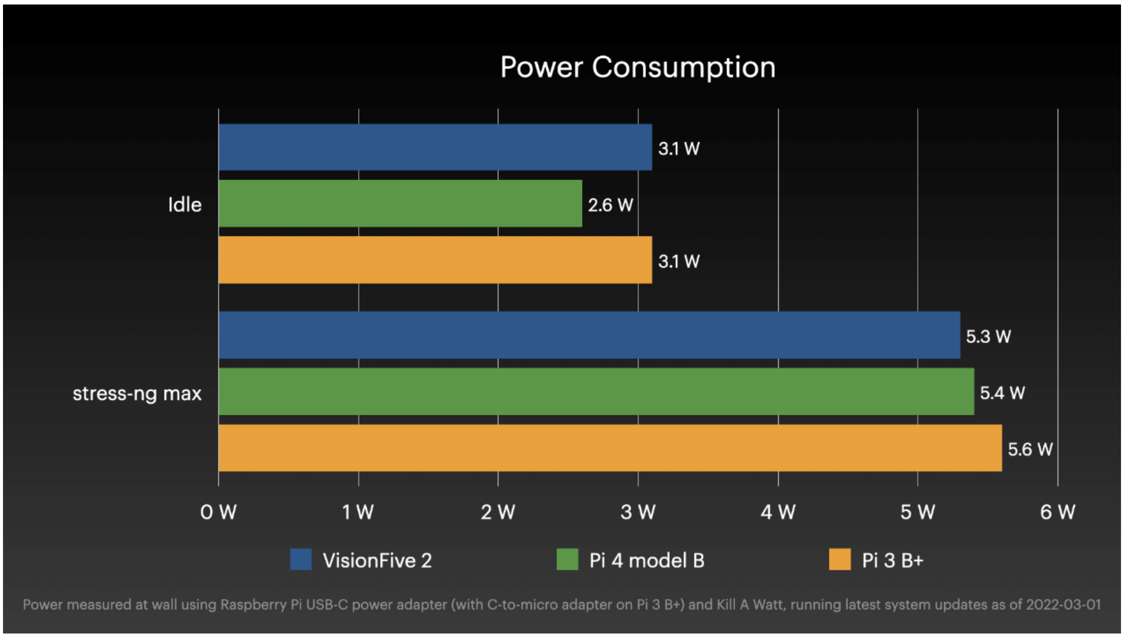


Figure 2. 3. Power consumptions of VisionFive 2, Pi 4 Model B, Pi 3 B+, in idle and fully stressed.

According to a review by James A. Chambers, although it has tested to have a great performance score, the board was criticised due to the lack of built-in WiFi and the boot loader's inability to support monitor/HDMI [10]. The table below shows some benchmark results done by James [11]. Samsung 960 EVO is used as SSD and StarFive VisionFive 2 as system board.

According to a review by Jonathan Bennett, the board lacks utility on OS and software support [12]. A major challenge is most of the PTS tests failed to run and install due to the newness of the RISC-V platform. For tests that succeeded to run, it shows that the performance is still weak compared to Raspberry Pi 4. However, the performance is expected with the possibility to improve as software becomes more mature.

Table 2. 1. Benchmark done on StarFive VisionFive V2 with Samsung 960 EVO as SSD

Benchmark Results	
DD Write:	107 MB/s
HDParm (Disk):	181.92 MB/s
HDParm (Cache):	181.63 MB/s
FIO 4KRandW:	9,006 IOPS
FIO 4KRandR:	44,425 IOPS
FIO 4KRandW:	36,024 KB/s
FIO 4KRandR:	177,700 KB/s
IOZone 4KR:	64,462 KB/s
IOZone 4KW:	43,701 KB/s
IOZone 4KRandR:	35,419 KB/s
IOZone 4KRandW:	75,853 KB/s

2.4 RISC-V Compiler Performance: A Comparison between GCC and LLVM/clang

The author of this study ran tests to compare the functionality of the two CPU architectures, RISC-V and ARM. Utilizing two well-known compilers, GCC and LLVM/Clang, the study's main objective was to assess the effects of different compilation optimization settings (-O0, -Os, -O1, -O2, -O3). Dhrystone and Coremark were used as the benchmark tests for the research [13].

The studies revealed some interesting results, one of which was that LLVM/Clang had lower performance and even crashed during compilation. This shows that LLVM/Clang might not be ideal for code compilation on these boards or needs additional optimization [13].

The -O0 and -Os optimization levels were the subject of another interesting finding. These optimization levels on the RISC-V architecture led to decreased performance when compared to ARM. It was also odd that when using GCC, the -Os optimization level was slower than -O0. This implies that specific optimization levels can affect RISC-V differently than ARM due to architectural differences [13].

The experiments with LLVM/Clang demonstrated that optimization levels above -O0 did not significantly improve performance. This suggests that for LLVM/Clang, the level of optimization may not have a significant effect on the speed at which code is executed on these boards. It's important to note that LLVM/Clang had trouble compiling Coremark for RISC-V, which is a critical limitation in this case. The difference in performance between the -O1 and -O2 optimization levels for Dhrystone was negligible, indicating that for this benchmark, performance may not be considerably impacted by the optimization level employed. But when applying -O3 optimization, ARM outperformed RISC-V far more, demonstrating that ARM was more susceptible to high degrees of optimization [13].

Overall, the study emphasizes how crucial it is to consider both the compiler choice and the precise optimization settings when working with diverse processor architectures as RISC-V and ARM. It also highlights the need for more research into the efficiency and compatibility of LLVM/Clang for RISC-V compilation [13].

2.5 Limitations of Previous Studies

In the previous studies using CoreMark benchmark, while the benchmark makes no assumptions about the architecture used, it fails to test memory enough [5]. Besides, the workload is synthetic and not realistic enough, which might raise integrity concerns about the benchmark. Synthetic workloads, which are produced artificially for testing, have shortcomings. Due to their lack of complexity and replication of user behavior, they might not adequately reflect real-world scenarios. System optimization may not match actual performance and can cause results to become bias. Limited coverage and impractical resource-intensive workload production are two possible outcomes. There are scaling and adaptability issues, and benchmark results may not be completely reliable. To resolve these problems, a hybrid approach combining synthetic and real-world workloads is frequently utilized to provide a more thorough and precise evaluation of system performance.

The benchmark itself is not enough the test the performance of the board as CoreMark only targets CPU performance specifically rather than a broad-based embedded software workload. Previously, Starfive officially released their CPU performance benchmark results on VisionFive 2 using CoreMark and Dhrystone, achieving a score of 5.09 CM/MHz on CoreMark and 2.64 CM/MHz on Dhrystone [6]. Although CoreMark are often recognized to be more reliable than Dhrystone, we should test differentbenchmarks as they can produce different results which can give us different insights. The concern of insufficient benchmark tools in previous research highlights the need for a more thorough approach to performance evaluation. It implies that the constraints in benchmark variety and comprehensiveness may lead to an incomplete understanding of system performance. The proposed approach emphasizes the selection of a varied collection of benchmarking instruments to solve this shortcoming. These tools need to accommodate a variety of workloads and use cases, ensuring that the performance evaluation is comprehensive and indicative in any environment.

Chapter 3
System Model

3.1 Design Specifications

The benchmarking of RISC-V processors, specifically the VisionFive 2 board, and evaluation of their performance under varied workloads and compilation flag settings are the foundation of the project's methodology. We have developed a multi-step process to achieve this:

Firstly, we will perform benchmark selection and configure these benchmarks to be able to run on the VisionFive 2 board. Then, experiment is conducted on the performance, compilation flag settings and power consumption of the board. After that, the data is recorded and will proceed to analyze it and interpret the findings to get useful insights. The data is then visualized using graphs, charts, and tables to provide clear representation of performance variations. Based on the analysis, recommendations are stated to form optimization strategies.

Putty will be used to connect with the board and execute commands remotely. For power consumption test, a PZEM-OO4T module will be used to measure the energy, and a python script from TheHWcave that provides an interface to monitor the metrics obtained from the power meter. During the power consumption test, the top command is executed to view the tests executed in different number of threads.

Phoronix test suite will provide the benchmark tests, Examples of system performances are iterations per second, floating-point operations per second, execution time, performance scores and more depending on the benchmark.

3.2 Hardware Specifications

3.2.1 VisionFive V2

Table 3. 1. Specifications of VisionFive V2

Item	Description
Processor/SOC	StarFive JH7110 64bit SoC with RV64GC, up to 1.5GHz
Memory	LPDDR4, 2GB/4GB/8GB
Storage	TF card slot
	Flash for Uboot
Video output	HDMI 2.0
	MIPI-DSI
Multimedia	Camera with MIPI CSI, up to 4k@30fps
	H.264 & H.265 4k@60fps Decoding H.265 1080p@30fps Encoding JPEG encoder/decoder
	1/ 4-pole stereo audio 2. HDMI
Connectivity	2x RJ45 Gigabit Ethernet
	2x USB2.0 + 2x USB 3.0
	M.2 M-Key
Power	USB-C port, 5V DC via USB-C with PD, up to 30W
	GPIO Power in, 5V DC via GPIO header (minimum 3A+)
	PoE
GPIO	40 pin GPIO header
Dimensions	100 x 72mm
Compliance	RoHS, FCC, CE
Button	Reset Button
Other	Debug Pin Headers

3.2.2 VisionFive V1

Table 3. 2. Specifications for VisionFive

Processor	U74 Dual-Core
Operating System	Linux
Memory	8GB LPDDR4
Video Processing	Video Encoder/Decoder (H264/H265) up to 4K@60FPS
	Dual channels of ISP, each channel support up to 4K@30FPS
	2 x MIPI-CSI, 1 x MIPI-DSI
	1 x HDMI 1.4 (up to 1080P@60FPS)
	Support MIPI-CSI TX for video output after ISP and AI processing
	JPEG Encoder/Decoder
Peripherals	4 x USB3.0 Ports
	1 x Gigabit Ethernet
	1 x 3.5mm Audio jack
	Support TRNG and OTP
	Support DMAC, QSPI and other peripheral
	40 Pin GPIO Header (28 x GPIO, I2C, I2S, SPI, UART)
	MicroSD card slot for operating system and data storage
	1 x 2.4GHz Wi-Fi and Bluetooth 4.2
	1 x Reset button and 1 x Boot button

3.2.3 Raspberry Pi 4b

Below are the hardware specifications for Raspberry Pi 4b: [14]

Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @1.8GHz

8GB LPDDR4-3200 SDRAM

2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE

Gigabit Ethernet

2 USB 3.0 ports; 2 USB 2.0 ports.

Raspberry Pi standard 40 pin GPIO header (fully backwards compatible withprevious boards)

2 × micro-HDMI ports (up to 4kp60 supported)

2-lane MIPI DSI display port

2-lane MIPI CSI camera port

4-pole stereo audio and composite video port

H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)

OpenGL ES 3.1, Vulkan 1.0

Micro-SD card slot for loading operating system and data storage

5V DC via USB-C connector (minimum 3A*)

5V DC via GPIO header (minimum 3A*)

Power over Ethernet (PoE) enabled (requires separate PoE HAT)

Operating temperature: 0 – 50 degrees C ambient

3.2.4 Intel UP Board

Below are the specifications for the Intel UP Board [15]:

Table 3. 3 Specifications of Intel UP Board

System	
Processor	Intel® Atom™ x5-Z8350 Processor SoC
Graphics	Intel® HD 400 Graphics, 12 EU GEN 8, up to 500MHz Support DX*11.1/12, Open GL*42, Open CL* 1.2 OGL ES3.0, H.264, HEVC(decode), VP8
I/O	HDMI x 1 I2S Audio port x 1
Camera	MIPI-CSI (4 MEGA pixel)
USB	USB 2.0 x4 USB 2.0 pin header (10 pins in total) x 2 USB3.0 OTG x 1
Expansion	40 pin General Purpose bus, supported by Altera Max V. ADC 8-bit@188ksos
RTC	Yes
Power	5V DC-in @ 4A 5.5/2.1mm jack
Dimension	3.37"x 2.22"(85.6mm x 56.5mm)
Memory	1GB/ 2GB/ 4GB DDR3L-1600
Storage	eMMC 16GB / 32GB/ 64GB
Display Interface	DSI/eDP
Ethernet	Gb Ethernet (full speed) RJ-45 x 1
OS Support	Win 10 Linux (ubilinux, Ubuntu, Yocto) Android Marshmallow
Operating Temperature	32°F ~ 140°F (0°C ~ 60°C)
Operation Humidity	10% ~ 80% relative humidity, non-condensing
Certification	CE/FCC Class A, RoHS Compliant

3.2.5 PZEM-004T Module

Below are the specifications for the PZEM-004T Module [16]:

Table 3. 4 Specifications of PZEM-004T Module

Function	Measuring range		Starting measure current/power		Resolution	Measure-ment accuracy	Display format
	10A	100A	10A	100A			
Voltage	80~260V				0.1V	0.5%	
Current	0~10A	0~100A	0.01A	0.02A	0.001A	0.5%	
Active power	0~2.3kW	0~23kW	0.4W		0.1W	0.5%	<1000W, it display one decimal, such as: 999.9W; ⩾1000W, it display only integer, such as: 1000W
Power factor	0.00~1.00				0.01	1%	
Frequency	45Hz~65Hz				0.1Hz	0.5%	
Active energy (Reset energy: use software to reset)	0~9999.99kWh				1Wh	0.5%	<110kWh, the display unit is Wh(1kWh=1000Wh), such as: 9999Wh; ⩾10kWh, the display unit is kWh, such as: 9999.99kWh
Over power alarm	Active power threshold can be set, when the measured active power exceeds the threshold, it can alarm						
Communication interface	RS485 interface						
size	Length * width * height=73.7*30*14.3mm (Bare pager)						
Power Supply	The power supply of single-phase power-frequency network supplies power to the main circuit through resistance-capacitance step-down, TTL output communication interface and Main circuit optocoupler isolation, for passive output, communication needs to provide external 5V power supply						
working temperature	-20°C~+60°C						

3.3 Benchmarking tools and performance metrics

The Phoronix Test Suite is an effective tool for measuring and assessing system performance. It makes benchmarking easier with its user-friendly UI and automated operations. Its modular architecture enables the integration of new test profiles and suites, giving users an extensive range of benchmarking options. The suite stores extensive data and provides a variety of analytical tools for interpreting results. Overall, the Phoronix Test Suite is a helpful tool for researchers, industry professionals, and enthusiasts who seek to analyze and optimize system performance.

CPU Benchmarks	
AOBench	AOBench is a lightweight ambient occlusion renderer, written in C. The size of 2048 x 2048 is being used by the test profile. In order to assess the CPU's computational performance, the benchmark calculates how long it takes the CPU to finish rendering. The benchmark involves ray tracing and rendering, which are common tasks for comparing the performance of various processors and graphics cards.
Coremark	Coremark is a common benchmark for measuring the performance of central processing units (CPUs) in embedded systems. Coremark is a collection of activities that include list processing, matrix manipulation, state machine emulation, and fundamental arithmetic operations which are typical operations in embedded systems. The benchmark measures the time it takes for a processor to execute the set of tasks. The primary metric used in Coremark is the number of iterations of the benchmark that can be completed per second.
Himeno	The Himeno benchmark evaluates computer systems' performance in solving the pressure Poisson equation with the point-Jacobi method. The Himeno benchmark typically measures a system's

	performance in terms of the execution time required to solve the pressure Poisson equation for a given grid size and convergence criterion. Lower execution times indicate improved performance.
FFTE	FFTE, created by Daisuke Takahashi, is a package designed for computing Discrete Fourier Transforms (DFTs) of sequences with lengths that are powers of 2, 3, and 5. FFTE places significant demands on the CPU in terms of computational resources, memory bandwidth, and cache utilization. The performance metric for this benchmark is MFLOPS (million floating point operations per second). MFLOPS is the rate at which a processor or algorithm can do floating-point arithmetic operations per second.
Perl Benchmark	The Perl benchmark suite for pod2html and the Perl interpreter compares the performance of multiple Perl utilities as well as the Perl interpreter itself. Perl benchmarking's performance metric is execution time, which is measured in seconds. Lower execution times indicate improved performance. Pod2html measures the time it takes for the CPU to convert POD files to HTML. Perl interpreter test estimates the time it takes the CPU to run Perl scripts using the Perl interpreter.
Memory Benchmarks	

Tinymembench	Tinymembench is a benchmarking tool for system memory (RAM). It measures many elements of memory performance, such as read, write, and copy speeds. Tinymembench measures bandwidth for sequential read, write, and copy operations. Higher bandwidths (MB/s) indicate better memory performance.
RAMspeed SMP	RAMspeed SMP is a benchmark that performs a series of memory copy operations with integer data. Operations involve copying blocks of memory from one area to another within the RAM. The key performance metric of RAMspeed SMP is memory copy bandwidth, which is measured in megabytes per second (MB/s). Higher bandwidth indicates better memory performance.
Cache Benchmarks	
Cachebench	CacheBench runs a series of memory access patterns to stress different levels of cache hierarchy. It does this by accessing memory in a controlled manner, using different stride patterns, block sizes, and access modes. CacheBench's performance metrics is memory and cache bandwidth, which is measured in megabytes per second (MB/s). Higher bandwidth indicates better performance.
GPU Benchmarks	

GraphicsMagick	GraphicsMagick performs various image processing operations on a high-resolution JPEG image. These operations include converting colors to the HWB color space, rotating the image, applying sharpening filters, adding swirl effects, adding Gaussian noise, resizing the image, and enhancing the image. The performance metric of this benchmark is "iterations per minute." This metric indicates how many times the benchmarked operations can be performed on the sample image in one minute. Higher values indicate better performance.
OpenCV	The OpenCV (Computer Vision) library's built-in performance tests cover a wide range of computer vision workloads. Core: Tests OpenCV's functions and utilities. Features 2D: Tests algorithms for feature recognition, extraction, and matching. Image Processing: Tests various image processing operations, including filtering, transformation, and enhancement. Video Processing: Tests video-related operations such as capture, processing, and analysis. Object Detection: Tests algorithms for detecting objects in images or video streams.

	DNN: Tests Deep neural network models and inference performance. Stitching: Tests image stitching algorithms for creating panoramas from multiple images. Graph API: Evaluates the performance of OpenCV's graph API for graph-based algorithms. The performance metric is the execution time of each test, measured in milliseconds, lower execution times indicate better performance.
Machine Learning Benchmarks	
NCNN	NCNN is a high-performance neural network inference framework developed from Tencent, targeted for embedded systems and other platforms. NCNN's performance metric is the time taken to execute inference on a given input data or batch of data, measured in milliseconds. Higher inference speeds indicate faster processing and better real-time performance.
Filesystem Benchmarks	
PostMark	This is a test of NetApp's PostMark benchmark designed to simulate small file testing like the tasks endured by web and mail servers. This test profile will set PostMark to perform 25,000 transactions with 500 files simultaneously with the file sizes ranging between 5 and 512 kilobytes.

	The performance metric measured is transactions per second (TPS), where higher amounts indicate better performance.
Multimedia Encoding Benchmarks	
FLAC Audio Encoding (Audio Encoding)	The benchmark measures the time required to encode a sample WAV file to FLAC (Free Lossless Audio Codec) audio format ten times using the best preset settings. The performance metrics is seconds where lower execution time indicates better performance.
WebP Image Encode (Image Encoding)	This test uses the cwebp program to encode an example JPEG image into WebP format with different quality and compression options. The test image is a sample JPEG with dimensions of 6000x4000 pixels. The performance metric is the encoding process's throughput in terms of megapixels processed per second (MP/s). Higher throughput suggests better performance.
Database Benchmarks	
SQLite Speedtest	SQLite Speedtest is designed to test the performance of various database operations, such as inserts, updates, selects, and deletes, with an increased problem size of 1,000. The performance metric is the total time taken by SQLite to execute the benchmark test with the increased problem size of 1,000 measured in seconds where lower execution time indicate better performance.
Programming Language Benchmarks	

PyPerformance	PyPerformance is the standard Python performance benchmark suite, testing different aspects of Python's performance across tasks and workloads. go: Measures time required to execute a simple "go" command in Python. 2to3: Measures the performance of Python's 2to3 tool, which translates Python 2 code to Python 3 code. raytrace: Measures the time required to render a scene using ray tracing techniques. crypto_pyaes: Measures the time required to complete cryptographic operations using the PyAES library. python_startup: Measures the Python interpreter's startup time. nbody: Measures the time required to model the gravitational interaction between astronomical bodies. Chaos: Measuring the time required to simulate python chaotic systems. float: Evaluates the performance of floating-point arithmetic operations in Python. json_loads: Measures the time required to parse JSON data, which converts JSON-formatted data into Python objects.

	regex_compile: Measures the speed of compiling regular expressions in Python using the re.compile() function. pathlib: Measures the time required to perform various file path operations using Python's pathlib package. Performance metrics is time where lower execution time indicates better performance.
Web Server Benchmarks	
PHPBench	PHPBench is a PHP benchmark suite. It runs tests to evaluate different elements of the PHP interpreter. PHPBench is used to compare hardware, operating systems, PHP versions, PHP accelerators and caches, compiler parameters, and more. Higher score indicates better performance.
Computational Fluid Dynamics Benchmarks	
Dolfyn	The Dolfyn test profile measures the execution time of the computational fluid dynamics demos included with Dolfyn. The performance metrics is seconds, where lower execution time indicates better performance.
Other System Performance Benchmarks	
T-Test1	T-test1 is a basic memory allocator benchmark. Users can choose to run the benchmark single-thread or double-thread. The performance metrics is seconds where lower execution time indicates better performance. Note the overall time does include the warmup time of the custom t-test1 compilation.

Machine Learning Models Tested	
Mobilenet	MobileNet is a convolutional neural network (CNN) model created for mobile and embedded vision applications. It uses depth-wise separable convolutions to reduce computational complexity while retaining high accuracy.
Mobilenet V2	MobileNetV2 is an enhancement to the original MobileNet architecture, incorporating inverted residuals and linear bottlenecks to improve performance.
Mobilenet V3	MobileNet v3: MobileNetV3 refines the architecture by focusing on improving the speed-accuracy trade-off using neural architecture search (NAS) techniques.
Shufflenet-V2	ShuffleNetV2 is another efficient CNN design that uses channel shuffling and pointwise group convolution to minimize computing costs.
mnasnet	MnasNet is a mobile-friendly neural architecture discovered by automated neural architecture search methods, aim to achieve high accuracy while minimizing computational cost.
Efficientnet-b0	EfficientNet is a series of CNN models that delivers cutting-edge accuracy through highly efficient architectural scaling methods.
blazeface	BlazeFace is a lightweight face detection model designed for mobile and embedded devices that

	prioritizes speed and accuracy in face identification tasks.
googlenet	GoogleNet, also known as Inception-v1, is a deep CNN architecture which make optimal use of computational resources by employing different filter sizes at each layer.
Vgg16	VGG16 is a deep CNN architecture known for its simplicity and efficiency. It has 16 convolutional layers and is commonly used as a baseline for many computer vision tasks.
Resnet18, Resnet50	ResNet (Residual Network) is a deep CNN design that uses skip connections to solve the vanishing gradient problem. ResNet18 and ResNet50 represent variations with 18 and 50 layers, respectively.
alexnet	AlexNet is one of the first deep CNN architectures, having won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012.
Yolov4-tiny	YOLO (You Only Look Once) is a popular object detection technique with real-time performance. YOLOv4-Tiny is a lightweight version designed for speed.
Squeezenet_ssd	SqueezeNet is a CNN architecture that is optimized for efficiency with a small number of parameters. SqueezeNet SSD is a model optimized for object detection tasks that employ single shot multibox detection (SSD).

Regnety_400m	RegNet is a family of effective CNN architectures that prioritize performance and scalability under a range of resource limitations.
Vision_transformer	Vision Transformer (ViT) is a transformer-based architecture for computer vision problems. It replaces traditional convolutional layers with self-attention mechanisms.
FastestDet	FastestDet is a lightweight and effective object detection model designed for real-time inference on mobile and edge devices,

3.4 System workflow

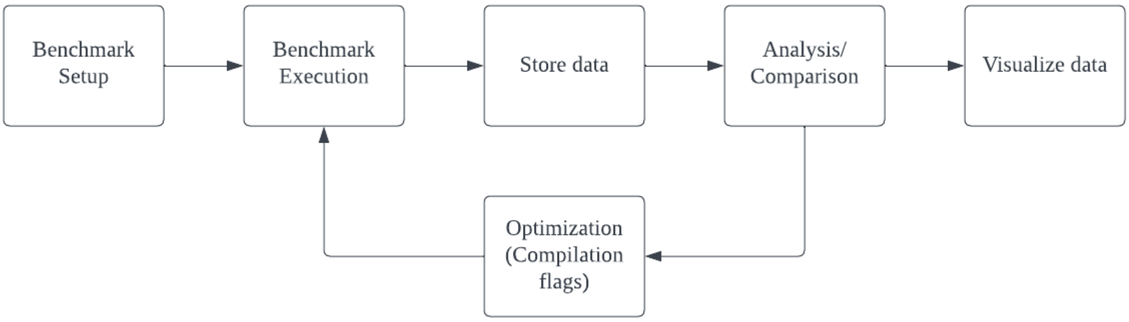


Figure 3. 1 Performance Benchmark Workflow

After selection of benchmarks, the dependencies for the benchmark are required to be installed for the benchmark installation. In this process, the selected benchmarks can be batched together to undergo the whole workflow instead of doing it one by one. After the benchmarks are installed, it will be executed and the results are stored. Results will be viewed to compare it with other boards that are benchmarked with the same test. In compilation flags testing, before benchmark execution, optimization is configured before benchmark is executed. Lastly, the data will be visualized to get a better understanding of the results obtained.

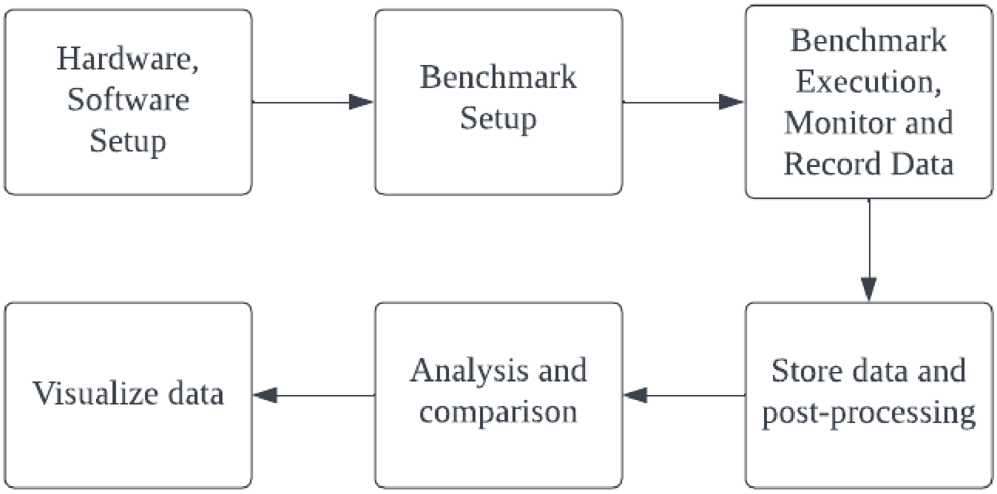


Figure 3. 2 Power consumption test workflow

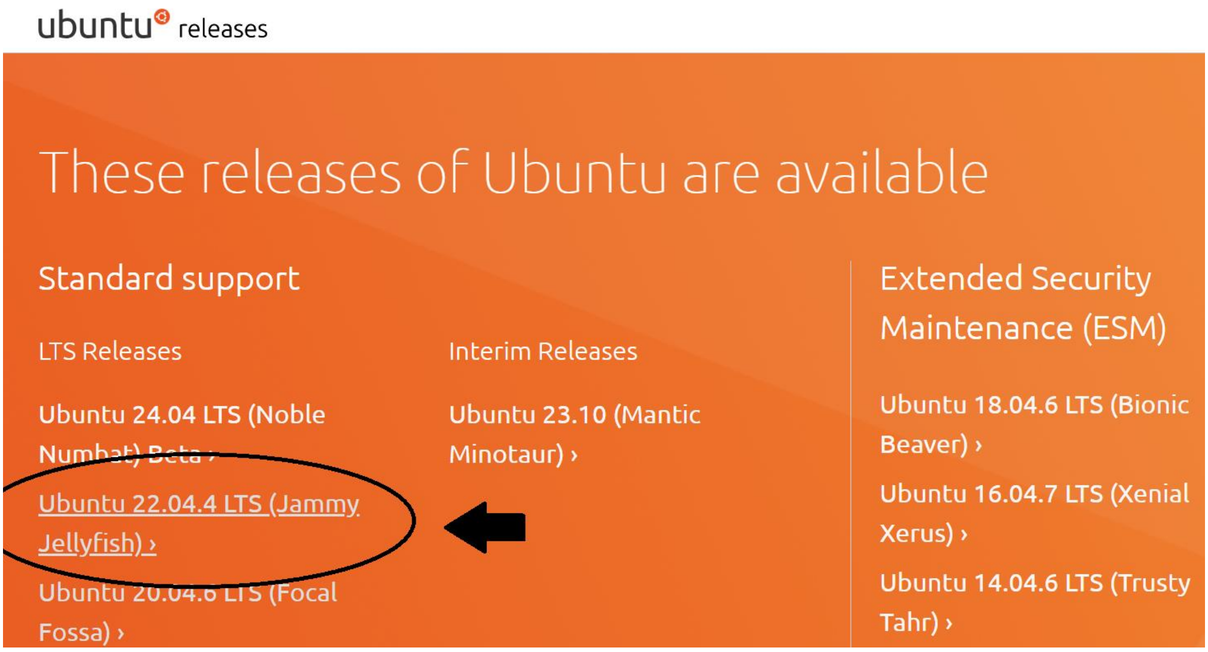
During power consumption test. The benchmark setup consists of downloading dependencies and installing the tests. After selecting the number of tests to run concurrently, the benchmark will be executed. During execution, the data is monitored and recorded using a python script called TheHWcave PowerMeter. The data will be stored inside a csv file. Data post-processing is required to delete inaccurate data that is measured during the test. After analysis and comparison, the data of power consumption scaling from one thread to four threads are visualized using graphs.

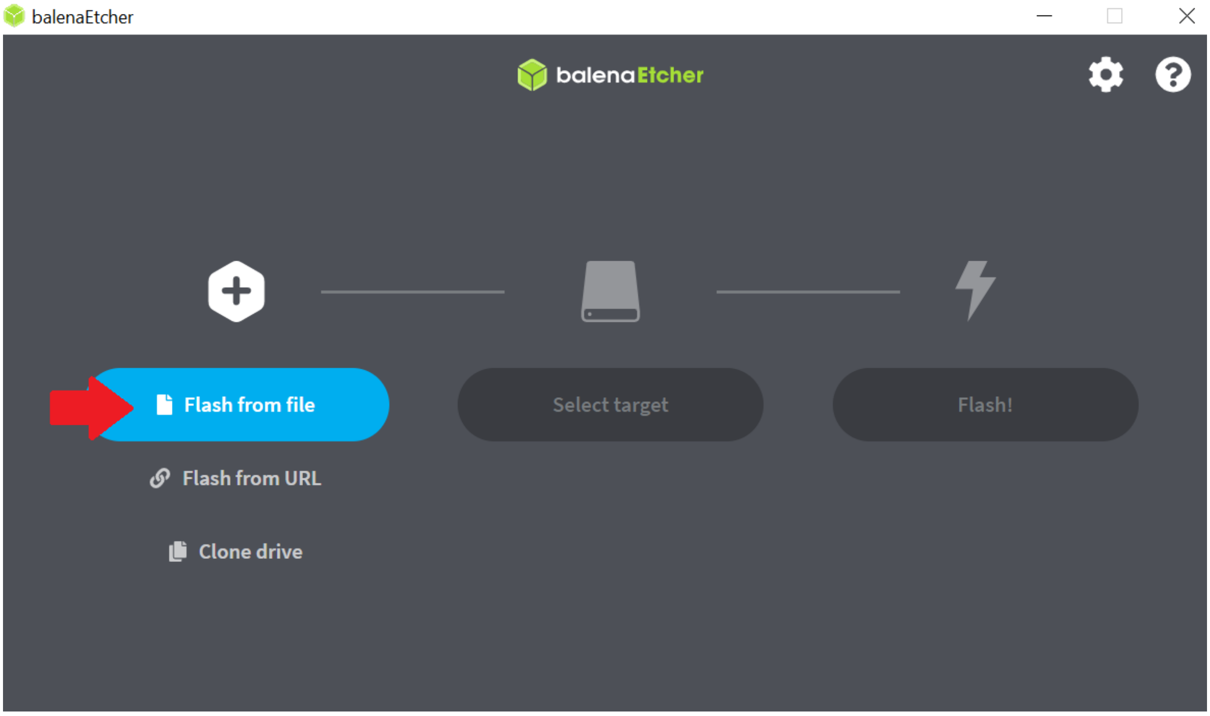
Chapter 4
EXPERIMENT

4.1 Intel UP Board Setup

4.1.1 Installing Ubuntu OS on Intel UP Board

The operating system that will be used in this project is Ubuntu 22.04. First, download the desktop image from the ubuntu releases website, then the image is burned into a flash drive using a software called balenaEtcher. After the operating system is installed on the board, the sudo apt update command is executed to get the latest updates.



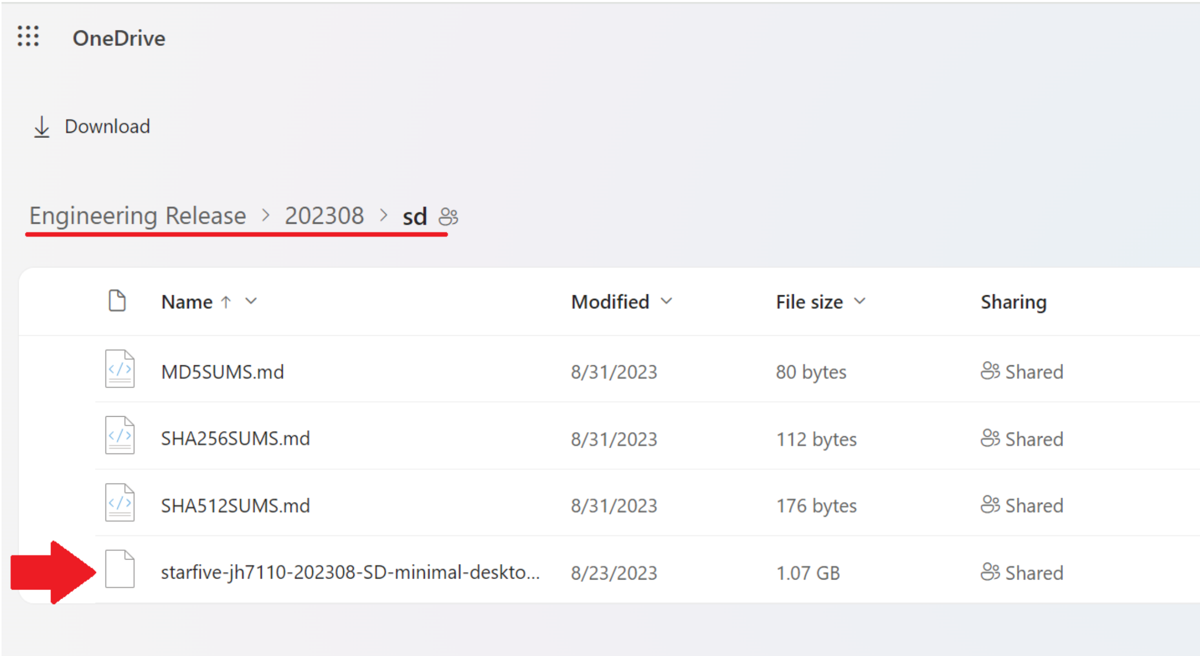


4.2 Starfive VisionFive V2 Setup

Insert a micro-SD card into the computer through a micro-SD card reader, or by a built-in card reader on a laptop.

Download the latest Debian image from the link provided in the quick start guide.

The image selected for this project is 202308, choose sd folder and download the image which is in img.bz2 compressed format, use 7-zip to extract the image.



Then, use BalenaEtcher software to flash the Debian image to a micro-SD card.

After that, insert the micro-SD card into starfive visionfive v2. Ensure the boot mode is set to sdio, like this:

Turn on power and connect display.

After logging into debian, we need to extend the partition on sd card.

Use the following command to list available elements.

```
~# df -h
```

Example output:

```
Filesystem      Size  Used Avail Use% Mounted on
udev            3.7G     0  3.7G   0% /dev
```

```
tmpfs           793M  3.1M  790M   1% /run
/dev/mmcblk1p4  2.0G  1.9G   88M  96% /
tmpfs           3.9G     0  3.9G   0% /dev/shm
tmpfs           5.0M   12K  5.0M   1% /run/lock
tmpfs           793M   32K  793M   1% /run/user/107
tmpfs           793M   24K  793M   1% /run/user/0
```

Run fdisk command fdisk /dev/mmcblk<x> and key in the prompt as follows:

```
root@starfive:~# fdisk /dev/mmcblk1

Welcome to fdisk (util-linux 2.38.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

GPT PMBR size mismatch (4505599 != 62929919) will be corrected by write.
This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.
Command (m for help): d
Partition number (1-4, default 4): 4

Partition 4 has been deleted.

Command (m for help): n
Partition number (4-128, default 4): 4
First sector (34-62929886, default 221184):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (221184-62929886, default 62928895):

Created a new partition 4 of type 'Linux filesystem' and of size 29.9 GiB.
Partition #4 contains a ext4 signature.

Do you want to remove the signature? [Y]es/[N]o: N
```

```
Command (m for help): w

The partition table has been altered.
Syncing disks.
```

Resize the /dev/mmcblk<X>p4 partition by running the resize2fs command to fully utilize the unused block.

Example output:

```
root@starfive:~# resize2fs /dev/mmcblk1p4
resize2fs 1.46.6-rc1 (12-Sep-2022)
Filesystem at /d[  295.372617] EXT4-fs (mmcblk1p4): resizing filesystem from 535291 to 7838464 blocks
ev/mmcblk1p4 is mounted on /; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 4
[  295.993163] EXT4-fs (mmcblk1p4): resized filesystem to 7838464
The filesystem on /dev/mmcblk1p4 is now 7838464 (4k) blocks long.
```

To verify the new size of partition, we can execute command df -h.

```
Tp root@starfive:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            3.7G     0  3.7G   0% /dev
tmpfs           793M  3.1M  790M   1% /run
/dev/mmcblk1p4   30G  1.9G   28G   7% /
tmpfs           3.9G     0  3.9G   0% /dev/shm
tmpfs           5.0M   12K  5.0M   1% /run/lock
tmpfs           793M   32K  793M   1% /run/user/107
tmpfs           793M   24K  793M   1% /run/user/0
```

Next, there are some packages that are not available through apt-get, and is provided by StarFive, to install these dependencies we can use following command:

```
wget https://github.com/starfive-tech/Debian/releases/download/v0.8.0-engineering-release-wayland/install_package_and_dependencies.sh
chmod +x install_package_and_dependencies.sh
sudo ./install_package_and_dependencies.sh
```

4.3 Performance Benchmarking

4.3.1 Using PuTTy to establish SSH connection.

To access the terminal of the boards directly from my laptop, I downloaded PuTTy to establish SSH connection. Firstly, I get the IP address of the board by using command ifconfig. The IP address usually starts with 192.168.X.X.

Then, at the terminal from my laptop, use command ssh [username]@[ip_address], password will be asked upon login.

4.3.2 Installing Phoronix Test Suite

To install the phoronix test suite, go to the phoronix test suite website. Since the OS installed on Intel UP Board is Ubuntu/Debian, that package will be downloaded. After extracting the archive, open it in terminal and type the sudo command for the installation sudo ./install.sh and the installation will be done. To access the test suite, simply type the command phoronix-test-suite in the terminal, and a list of command and instructions can be found.

4.3.3 Installing a test.

To install a test in phoronix-test-suite, we can execute command phoronix-test-suite interactive to run and install a test through an interactive dashboard or execute command phoronix-test-suite benchmark [benchmark_name] and will proceed to install and execute the test.

```
loh@loh-UP-CHT01:~$ phoronix-test-suite benchmark aobench

    Evaluating External Test Dependencies .........................

Phoronix Test Suite v10.8.4

    To Install:    pts/aobench-1.0.1

    Determining File Requirements ..................................
    Searching Download Caches ......................................

    1 Test To Install
        1MB Of Disk Space Is Needed
        2 Seconds Estimated Install Time

    pts/aobench-1.0.1:
        Test Installation 1 of 1
        1 File Needed
        File Found: aobench-20180207.zip
        Approximate Install Size: 1 MB
        Estimated Install Time: 2 Seconds
        Installing Test @ 08:33:06
```

Figure 4. 1 Installing a test.

To benchmark a batch of tests, we can execute command phoronix-test-suite build-suite, and it will prompt user to build a custom test suite.

```
Phoronix Test Suite v10.8.4
Test Suite Creation


Enter name of suite: batch-01

1: System
2: Processor
3: Disk
4: Graphics
5: Memory
6: Network
7: OS
8: Other
Select test type: 8

Enter suite maintainer name: loh

Enter suite description: a batch of tests
    Bind current test profile versions to test suite (Y/n): y

Available Options:

1: Add Test
2: Add Sub-Suite
3: Save & Exit
Select next operation:
```

Figure 4. 2 Creating a test suite.

4.3.4 Running the benchmark.

Before beginning the benchmarking process, we will set the initialization level to 3 by using the following command:

```
sudo init 3
```

This step will prevent any potential interference or inconsistencies that may occur throughout the benchmarking process and create a stable and regulated environment, which improves the accuracy and dependability of our benchmarking results.

Once the initialization level has been set up, we may proceed to run the created test suite. To start the benchmarking process for the full suite, run the following command: 'phoronix-test-suite benchmark [test-suite].'

```
Saved -- to run this suite, type: phoronix-test-suite benchmark batch-01

loh@loh-UP-CHT01:~$ phoronix-test-suite benchmark batch-01

    Evaluating External Test Dependencies ...................................
```

Figure 4. 3 Running a batch of tests.

Individual tests within the suite can also be run individually. To run tests separately, use the following command.

```
phoronix-test-suite benchmark [test-name]
```

4.4 Power Consumption Test

4.4.1 PZEM-004T Driver installation

First, get the PL2303 Prolific Driver installer and run the installer. After installation, run these commands in the terminal.

```
C:\WINDOWS\system32\cmd.exe
C:\Users\lezho\OneDrive\Desktop>cd C:\Users\lezho\OneDrive\Desktop\

C:\Users\lezho\OneDrive\Desktop>Regsvr32 isAnalogLibrary.ocx
```

Figure 4. 4 Registering an ActiveX Control

Running the software.

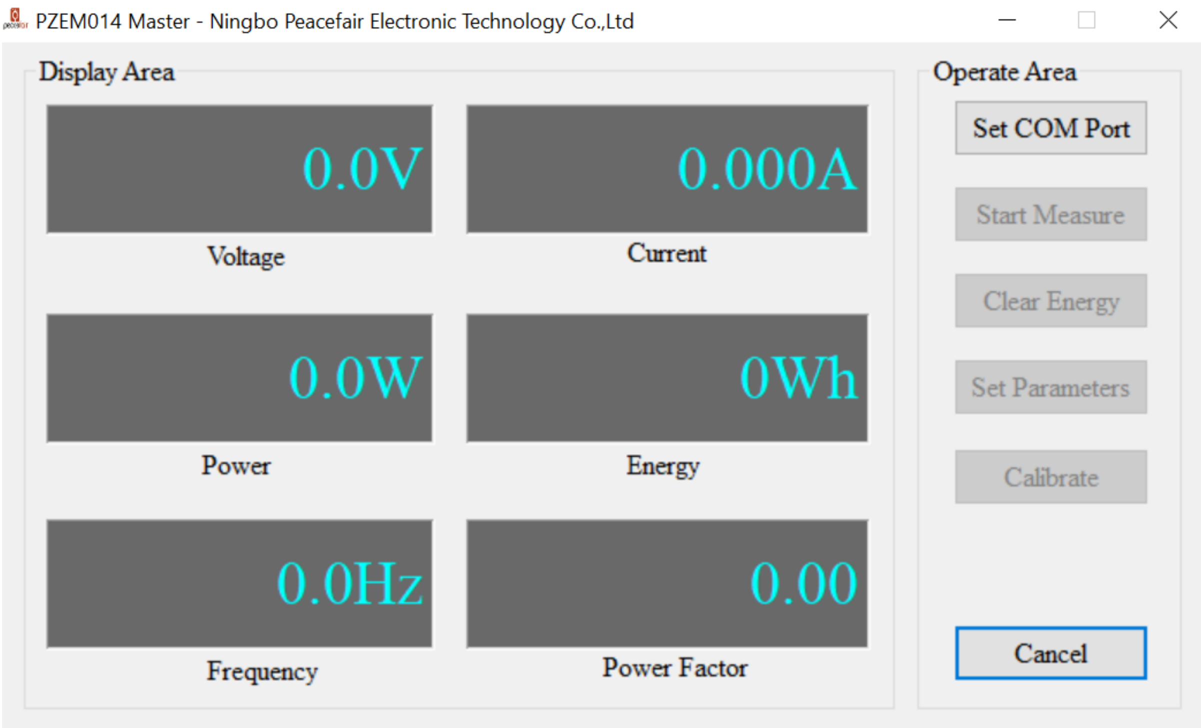


Figure 4. 5. Running PZEM014 Master

Before setting the COM port, plug in the USB from the PZEM-004T Module.

To set COM port, go to the search bar beside Windows and type device manager, then expand ports (COM & LPT) section, find the prolific driver and its COM port, set the COM port to that prolific driver's COM port.

In the power consumption test later, we will use a python script from TheHWcave to measure the power consumption. The interface will show data after typing the COM port, during the test was executed, we can press the record button and the metrics below will be recorded into a csv file.

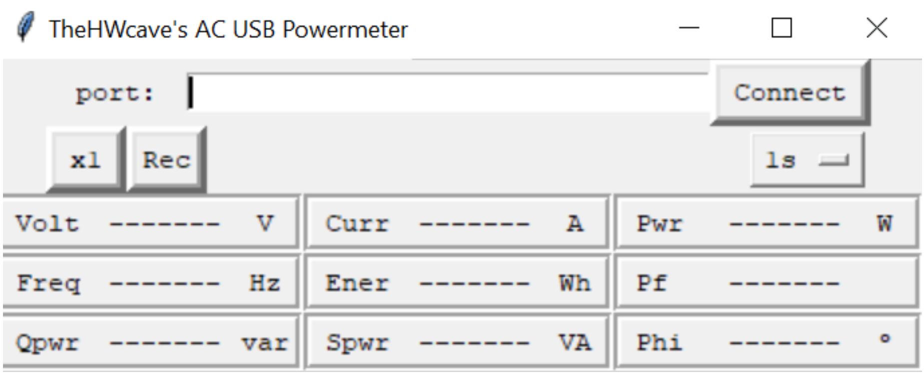
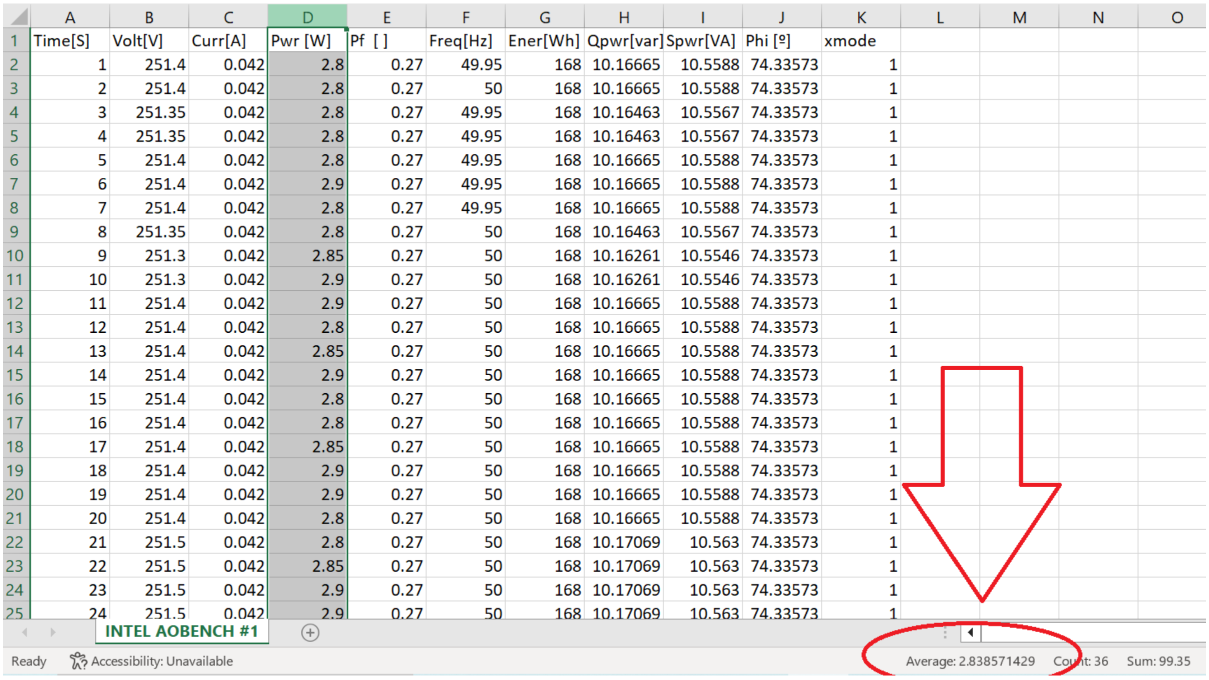


Figure 4. 6. Python script for interface of measuring power consumption



Time[S]	Volt[V]	Curr[A]	Pwr [W]	Pf [ ]	Freq[Hz]	Ener[Wh]	Qpwr[var]	Spwr[VA]	Phi [º]	xmode
1	251.4	0.042	2.8	0.27	49.95	168	10.16665	10.5588	74.33573	1
2	251.4	0.042	2.8	0.27	50	168	10.16665	10.5588	74.33573	1
3	251.35	0.042	2.8	0.27	49.95	168	10.16463	10.5567	74.33573	1
4	251.35	0.042	2.8	0.27	49.95	168	10.16463	10.5567	74.33573	1
5	251.4	0.042	2.8	0.27	49.95	168	10.16665	10.5588	74.33573	1
6	251.4	0.042	2.9	0.27	49.95	168	10.16665	10.5588	74.33573	1
7	251.4	0.042	2.8	0.27	49.95	168	10.16665	10.5588	74.33573	1
8	251.35	0.042	2.8	0.27	50	168	10.16463	10.5567	74.33573	1
9	251.3	0.042	2.85	0.27	50	168	10.16261	10.5546	74.33573	1
10	251.3	0.042	2.9	0.27	50	168	10.16261	10.5546	74.33573	1
11	251.4	0.042	2.9	0.27	50	168	10.16665	10.5588	74.33573	1
12	251.4	0.042	2.8	0.27	50	168	10.16665	10.5588	74.33573	1
13	251.4	0.042	2.85	0.27	50	168	10.16665	10.5588	74.33573	1
14	251.4	0.042	2.9	0.27	50	168	10.16665	10.5588	74.33573	1
15	251.4	0.042	2.8	0.27	50	168	10.16665	10.5588	74.33573	1
16	251.4	0.042	2.8	0.27	50	168	10.16665	10.5588	74.33573	1
17	251.4	0.042	2.85	0.27	50	168	10.16665	10.5588	74.33573	1
18	251.4	0.042	2.9	0.27	50	168	10.16665	10.5588	74.33573	1
19	251.4	0.042	2.9	0.27	50	168	10.16665	10.5588	74.33573	1
20	251.4	0.042	2.8	0.27	50	168	10.16665	10.5588	74.33573	1
21	251.5	0.042	2.8	0.27	50	168	10.17069	10.563	74.33573	1
22	251.5	0.042	2.85	0.27	50	168	10.17069	10.563	74.33573	1
23	251.5	0.042	2.9	0.27	50	168	10.17069	10.563	74.33573	1
24	251.5	0.042	2.9	0.27	50	168	10.17069	10.563	74.33573	1

INTEL AOBENCH #1

Ready Accessibility: Unavailable Average: 2.838571429 Count: 36 Sum: 99.35

The average power consumption for the benchmark, measured in Watt can be obtained from the csv file.

4.5 Compilation Flag Optimizations

To run the tests with compiler flag optimizations, -O1, -O2 and -O3, we can put the tests into a batch first, then execute the following commands. The following commands will run the batch of tests in -O1, -O2, and -O3 optmization settings.

CFLAGS="-O1" CXXFLAGS="-O1" TEST_RESULTS_NAME="result-name" phoronix-test-suite batch-benchmark batch-01

CFLAGS="-O2" CXXFLAGS="-O2" TEST_RESULTS_NAME="result-name" phoronix-test-suite batch-benchmark batch-01

CFLAGS="-O3" CXXFLAGS="-O3" TEST_RESULTS_NAME="result-name" phoronix-test-suite batch-benchmark batch-01

To use the command lines for RISC-V options, open file systems in starfive visionfive and show hidden folders. Then navigate to. phoronix-test-suite/pts/test-profiles and select the tests to be optimized. Open the install.sh of the test and make modifications to include the compiler flags -march=rv64gc and -mcpu=sifive-u74, for example:

Before configuration:

```
#!/bin/sh

tar -xf himenobmtxpa-2.tar.xz

if [ $OS_TYPE = "Linux" ]
then
  if grep avx2 /proc/cpuinfo > /dev/null
  then
    export CFLAGS="$CFLAGS -mavx2"
  fi
fi

cc himenobmtxpa.c -O3 $CFLAGS -o himenobmtxpa
echo $? > ~/install-exit-status

echo "#!/bin/sh
./himenobmtxpa s > \$LOG_FILE 2>&1
echo \$? > ~/test-exit-status" > himeno
chmod +x himeno
```

Figure 4. 7. Install.sh file of Himeno Benchmark before configuration

After configuration:

```
#!/bin/sh

tar -xf himenobmtxpa-2.tar.xz

if [ $OS_TYPE = "Linux" ]
then
    if grep avx2 /proc/cpuinfo > /dev/null
    then
        export CFLAGS="$CFLAGS -mavx2"
    fi
fi

export CFLAGS = "$CFLAGS -O3 -march=rv64gc -mcpu=sifive-u74"
cc himenobmtxpa.c -O3 $CFLAGS -o himenobmtxpa
echo $? > ~/install-exit-status

echo "#!/bin/sh
./himenobmtxpa s > \$LOG_FILE 2>&1
echo \$? > ~/test-exit-status" > himeno
chmod +x himeno
```

Figure 4. 8. Install.sh of Himeno Benchmark after optimizations

4.6 Analysis and Reporting

The analysis of benchmarking results and conclusions generated from the obtained data is a methodical procedure that seeks to extract useful insights. Initially, the tests are conducted repeatedly to ensure the result's accuracy and relevance to the benchmarking study's objectives. Raw data is processed into structured formats and turning it into useful figures or averages relevant to the performance evaluation criteria. Maintaining consistency and accuracy throughout this process allows for reliable analysis and conclusions.

Benchmarking studies rely heavily on comparative analysis, which allows for the comparison of results across multiple test instances, hardware configurations, and software versions. Visualizations such as histograms, box plots, and scatter plots are used to show comparative analysis results, making them easier to analyze and understand.

Root cause analysis is used to evaluate any outliers or unexpected results discovered during the benchmarking process. Examining further into the underlying reasons of such abnormalities can provide insights into performance bottlenecks or system performance concerns.

The results are presented with the use of graphs and visuals to improve understanding. The analysis's main findings and conclusions are briefly discussed and backed up by statistics to give more perspective. The benchmarking study's key conclusions are summarized in the report, along with any ramifications for further research or choice.

Chapter 5

Results

5.1 Benchmark results and result analysis

The results below are tests that are conducted on intel up board, starfive visionfive board, starfive visionfive v2 board and raspberry pi 4b.

5.1.1 CPU Benchmark result

Overall, the Raspberry Pi 4B performs well across a variety of CPU benchmarks. The Intel UP board also performs well across all benchmarks, notably in floating-point computations. The StarFive v2 board runs well, however it falls behind the Raspberry Pi 4B and Intel UP in most benchmarks. The StarFive v1 board has the lowest performance across all benchmarks, indicating that its CPU is less powerful than the other boards.

Considering the difference in core count between the StarFive v1 board (2 cores) and the other boards (quad-core), it is clear why the StarFive v1 board consistently performs lower across the benchmarks. The StarFive v1 board's capacity to handle multi-threaded tasks is inherently limited because it only has two cores, as opposed to quad-core boards.

Despite this limitation, the results highlight the advancements made to the StarFive v2 board over its predecessor. These enhancements are significant given the shift from a dual-core to a quad-core design.

However, despite these improvements, the StarFive v2 board still trails the Raspberry Pi 4B and the Intel UP board in most benchmarks, which may be due to variables other than core count, such as architecture differences clock speeds, and optimizations.

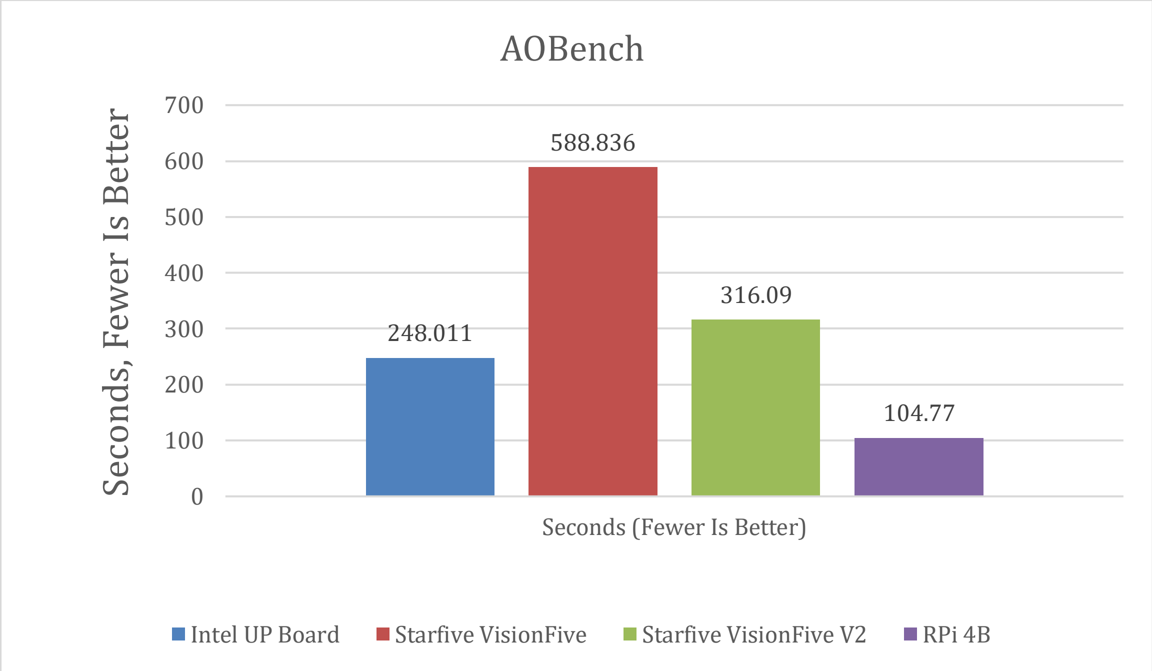


Figure 5. 1 AOBench Results

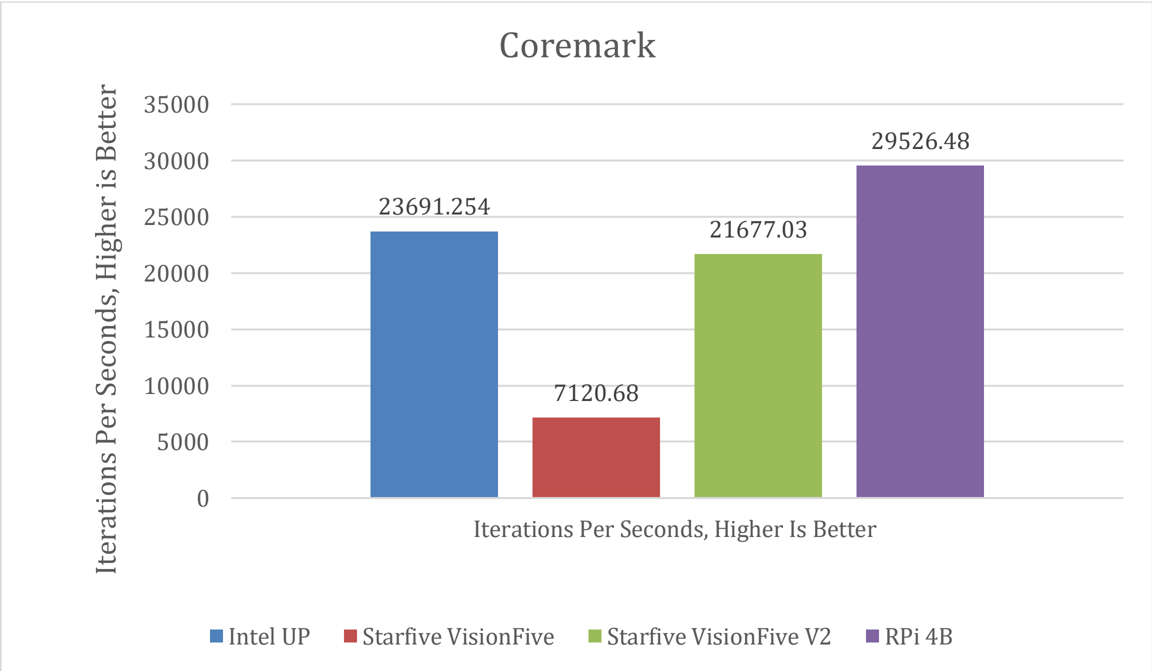


Figure 5. 2 Coremark Results

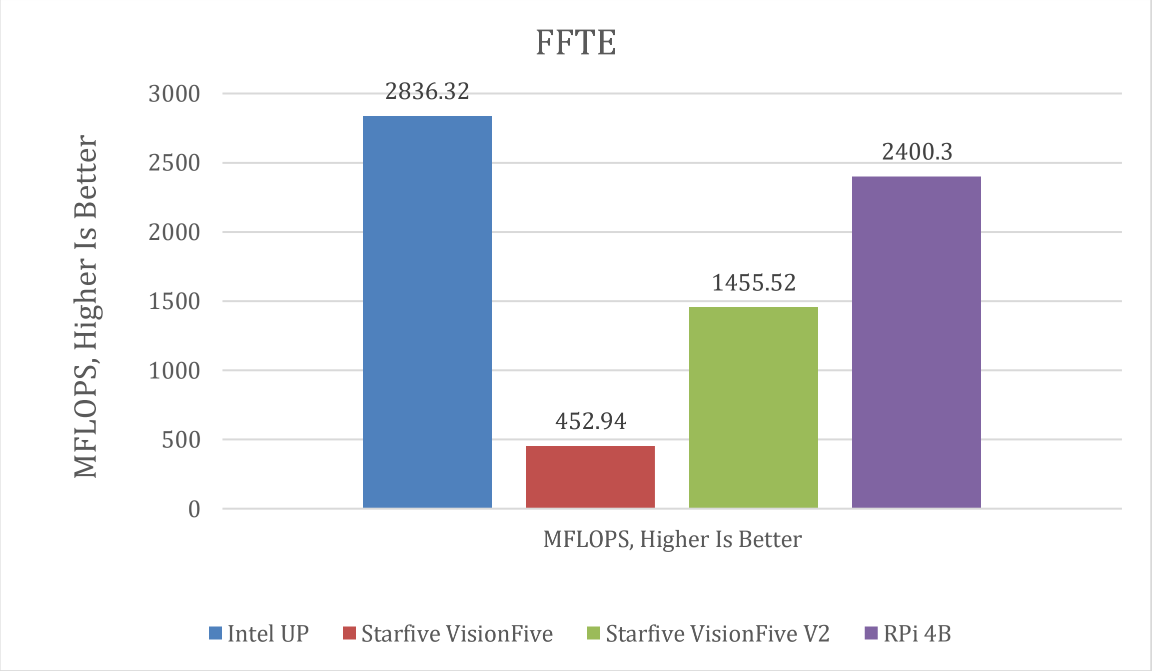


Figure 5. 3 FFTE Results

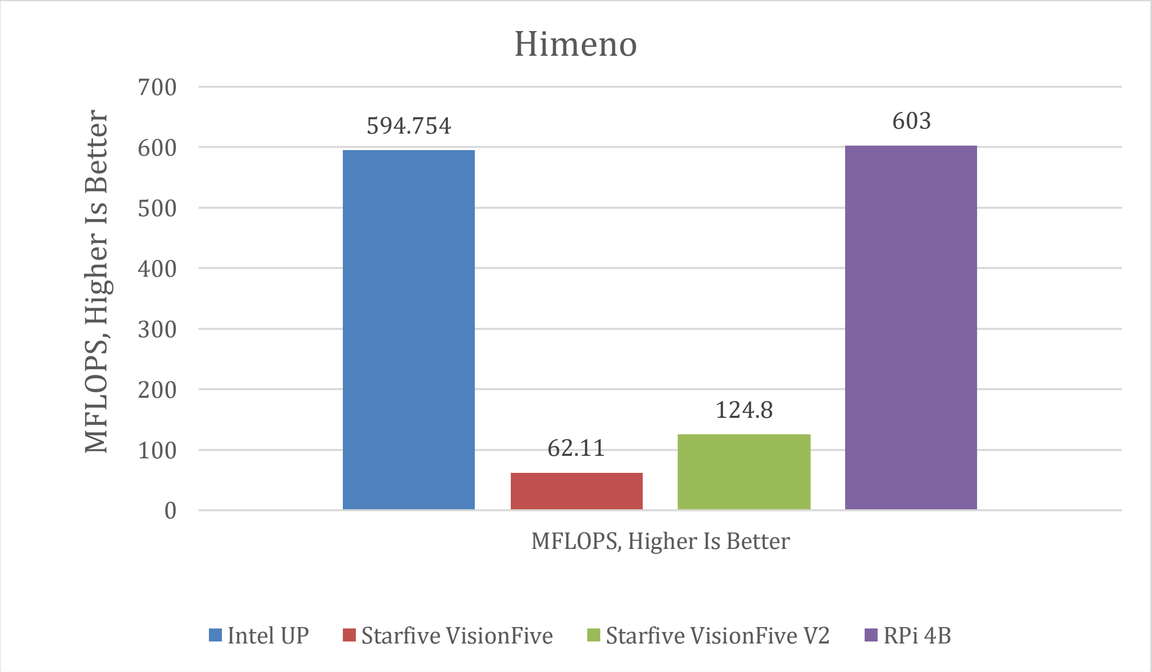


Figure 5. 4 Himeno Results

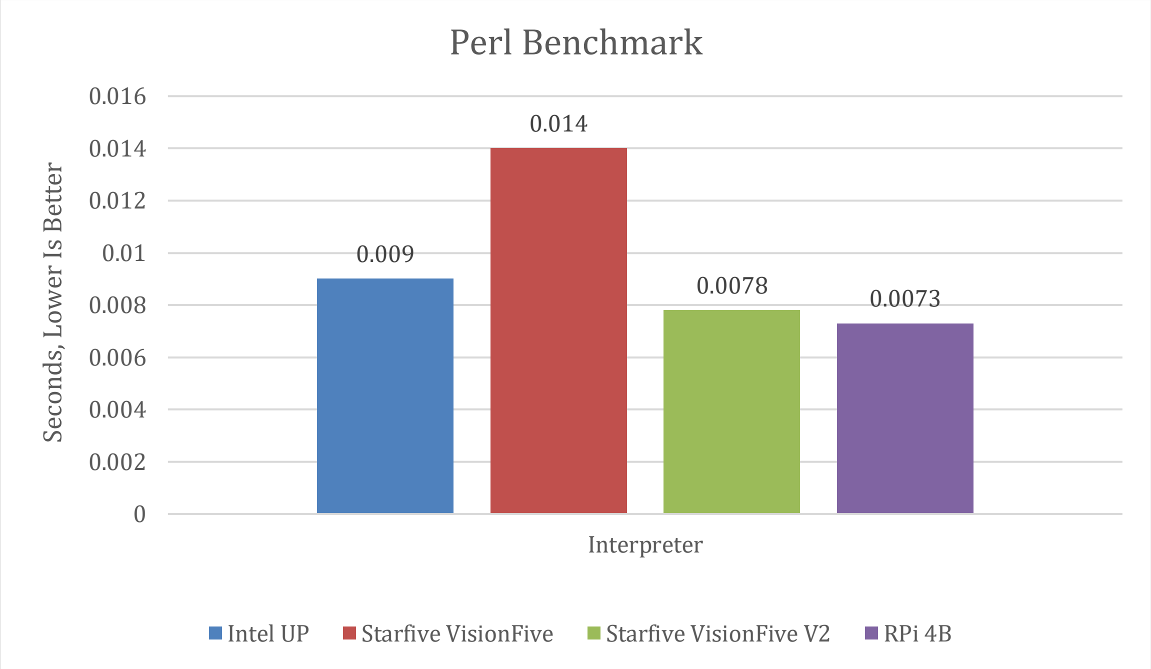


Figure 5. 5 Perl Benchmark Interpreter Results

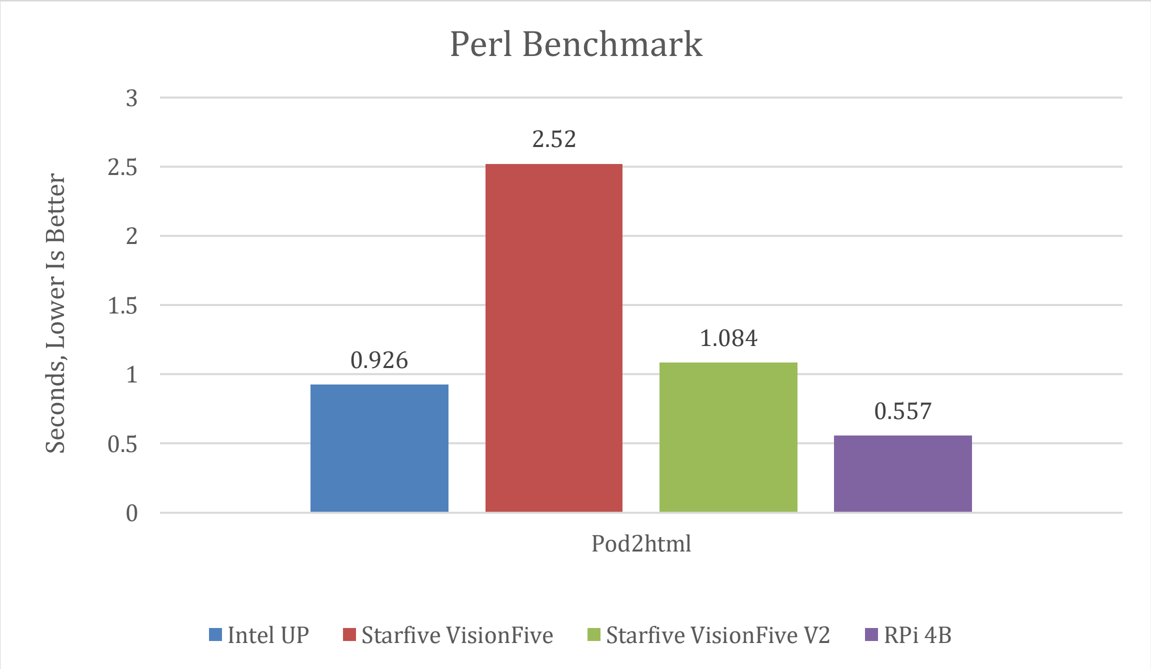


Figure 5. 6 Perl Benchmark Pod2html Results

5.1.2 Memory Benchmark Result

In memory benchmarks, the Intel UP board performs the best, closely followed by the Raspberry Pi 4B. The StarFive v2 board outperforms the StarFive v1 board in all memory benchmarks showing significant improvement to its predecessor. However, it still trails the Intel UP and Raspberry Pi 4B in memory performance.

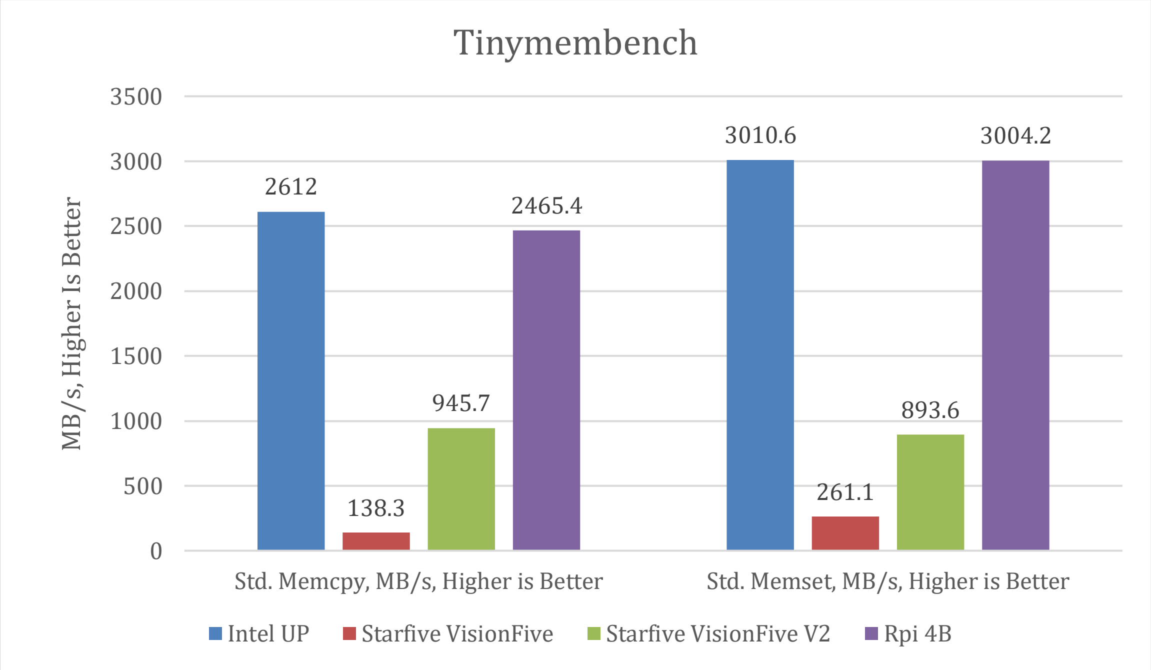


Figure 5. 7 Tinymembench Results

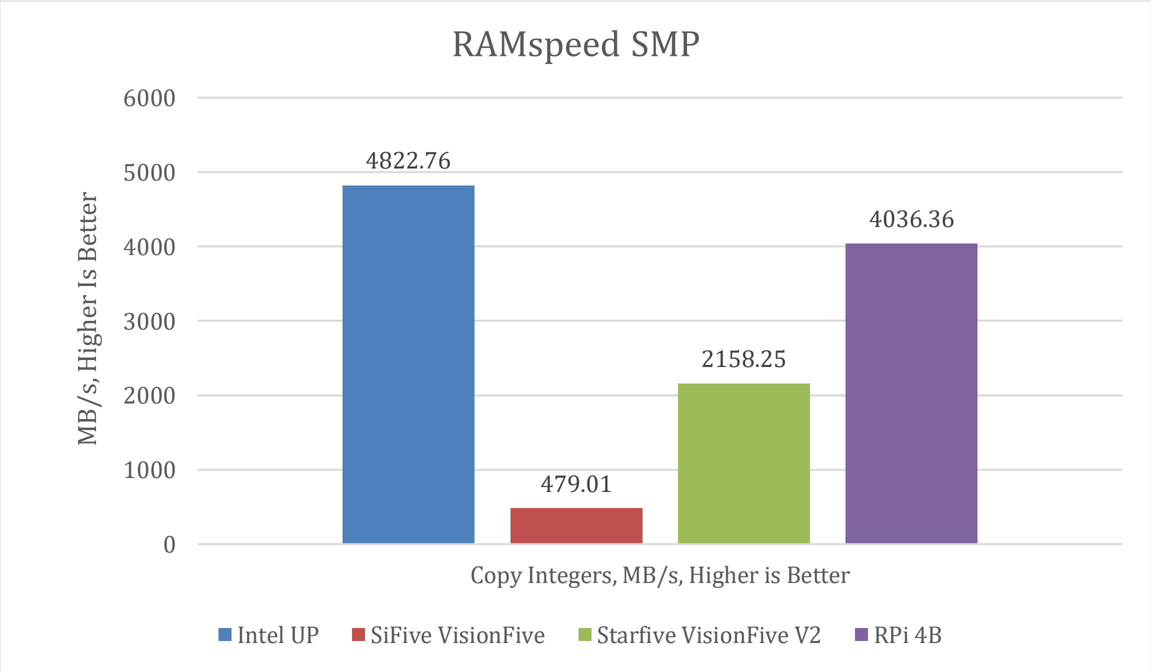


Figure 5. 8 RAMspeed SMP Results

5.1.3 Cache Benchmark Result

Moving on to cache benchmarks, in the read benchmarks, the Raspberry Pi 4B outperforms all other boards significantly. Following closely behind is the StarFive v2, which outperforms its predecessor and achieves competitive read speeds. The Intel UP board also performs well in reading operations being slightly behind StarFive v2. In contrast, the StarFive v1 has lower read performance than the other boards.

The Intel UP board leads the writing benchmarks, closely followed by the Raspberry Pi 4B. The StarFive v2 board also has good write performance when compared to Intel UP board and Raspberry Pi 4B and showing significant improvements over the StarFive v1.

In read-modify-write operations, the Raspberry Pi 4B outperforms Intel UP, by twice the speed. However, the StarFive v2 performs significantly worse in this benchmark when compared to previous benchmarks like read operations and writing operations, with a score that is just half the speed of the Intel UP. Meanwhile, the StarFive v1 board ranks bottom in read-modify-write performance among the four boards.

Overall, the Raspberry Pi 4B shows superior performance in this benchmark, while the performance of Intel UP and StarFive v2 board in read and write operations is equally matched, StarFive v2 board unexpectedly performs worse in read-modify-write operations. However, it outperforms its predecessor and is able to achieve a good performance in read and write operations.

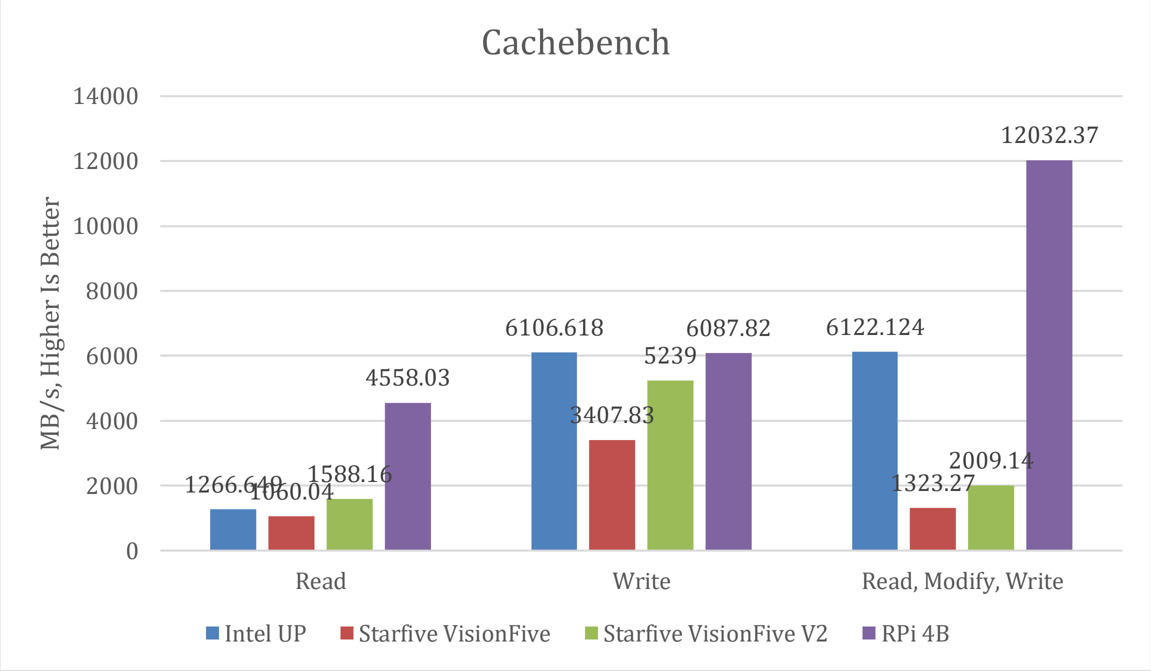


Figure 5. 9 Cachebench Results

5.1.4 GPU Benchmark Result

In the GraphicsMagick benchmark, the Intel UP board constantly outperforms. In the HWB color space operation, the StarFive v2 board outperforms the Raspberry Pi 4B, with 39 iterations per minute versus 19. Furthermore, in the noise-gaussian operation, Intel UP scores 12, with StarFive v2 closely following with a score of 11, while the Raspberry Pi and StarFive v1 trail with a score of 4. Moving on to resizing operations, the Intel UP board scores 41, while the StarFive v2 scores 35, representing a substantial improvement over the StarFive v1. Furthermore, the StarFive v2 maintains a large advantage over the Raspberry Pi, with a score of 35 to 9. In rotating operations, the Intel UP board scores 82, while the StarFive v2 scores 54, once again outperforming both the StarFive v1 and the Raspberry Pi 4B. In terms of sharpening operations, the StarFive v2 has the highest score of 11, followed by the Intel UP with a score of 5. The StarFive v1 scores 4, while the Raspberry Pi 4B scores 2. In swirl operations, the Intel UP scores 20, while the StarFive v2 scores 12, indicating significant performance improvement over the StarFive v1 and Raspberry Pi, both of which score 4.

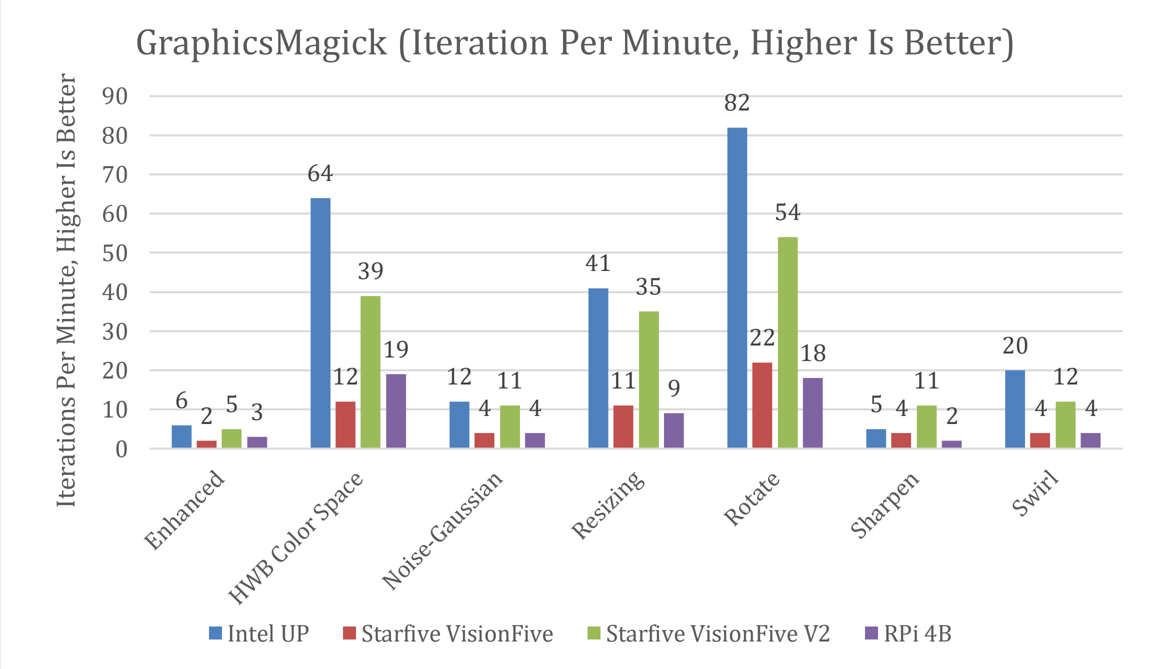


Figure 5. 10 GraphicsMagick Results

In OpenCV Benchmarks, we can observe that the Intel UP board consistently demonstrates superior GPU performance across all benchmarks, with the StarFive v2 board outperforming Raspberry Pi 4B and showing notable improvements over its predecessor. While the core benchmark where StarFive VisionFive V2 performs slightly slower than Raspberry Pi 4B, StarFive VisionFive V2 performs significantly better when compared with Raspberry Pi 4B.

Overall, while the Intel UP board showcases superior performance across various GPU benchmarks, StarFive v2 board demonstrates notable improvements compared to its predecessor and maintains competitive performance against the Raspberry Pi 4B.

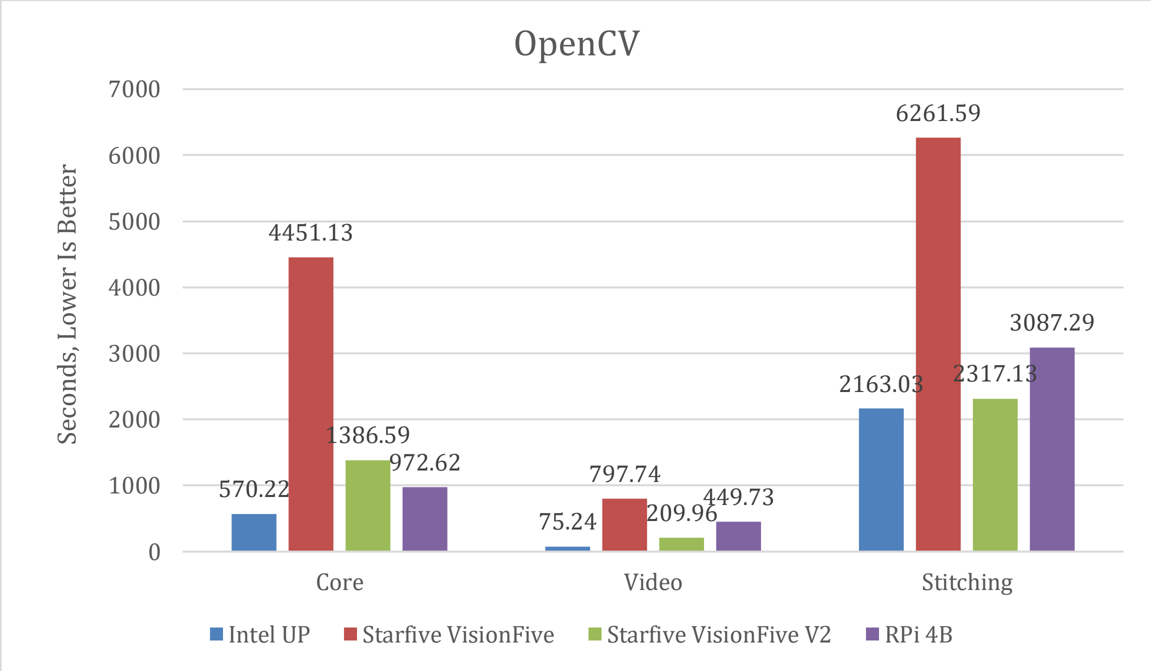


Figure 5. 11 OpenCV Results (1)

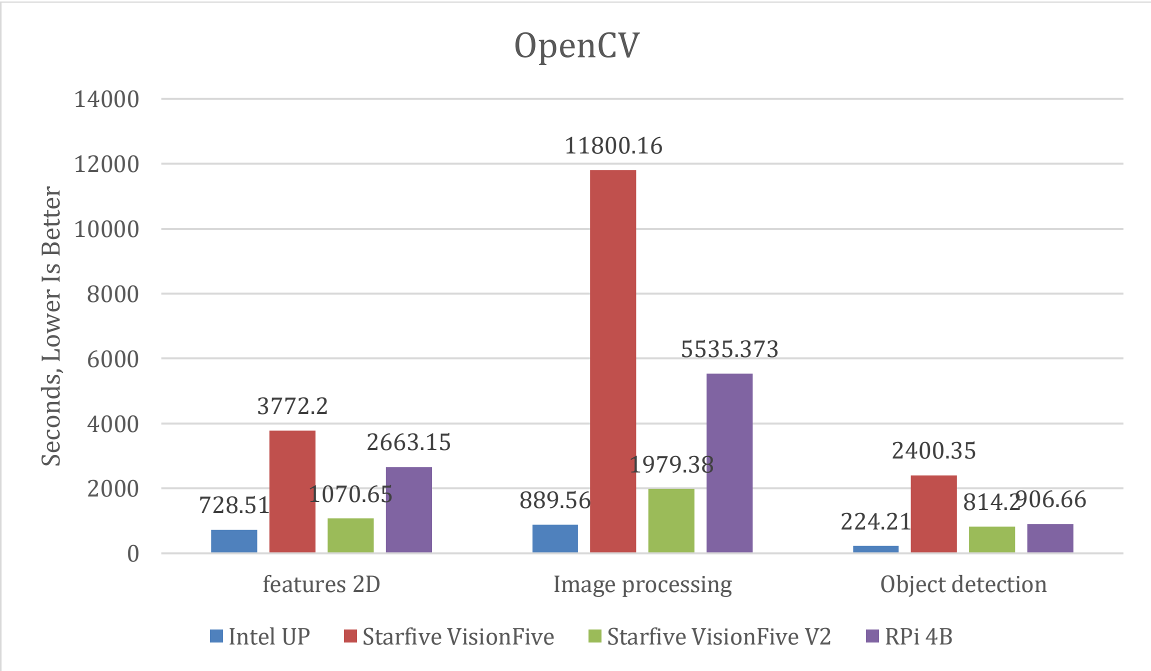


Figure 5. 12 OpenCV Results (2)

5.1.6 File-system Benchmark Result

In filesystem benchmarks using Postmark, the Starfive VisionFive V2 board demonstrates the highest performance with a TPS(Transactions Per Second) of 626 followed by the Intel UP board with a TPS of 508. The StarFive v1 board lags behind both the Intel UP and StarFive v2 in filesystem performance, while the Raspberry Pi 4B demonstrates comparatively lower performance among the boards. This highlights the strength of Starfive VisionFive V2 in small file transactions under simultaneous access. It indicates a filesystem with a high throughput, low latency, efficient concurrency support, efficient space utilization, robustness, and reliability in ensuring data integrity.

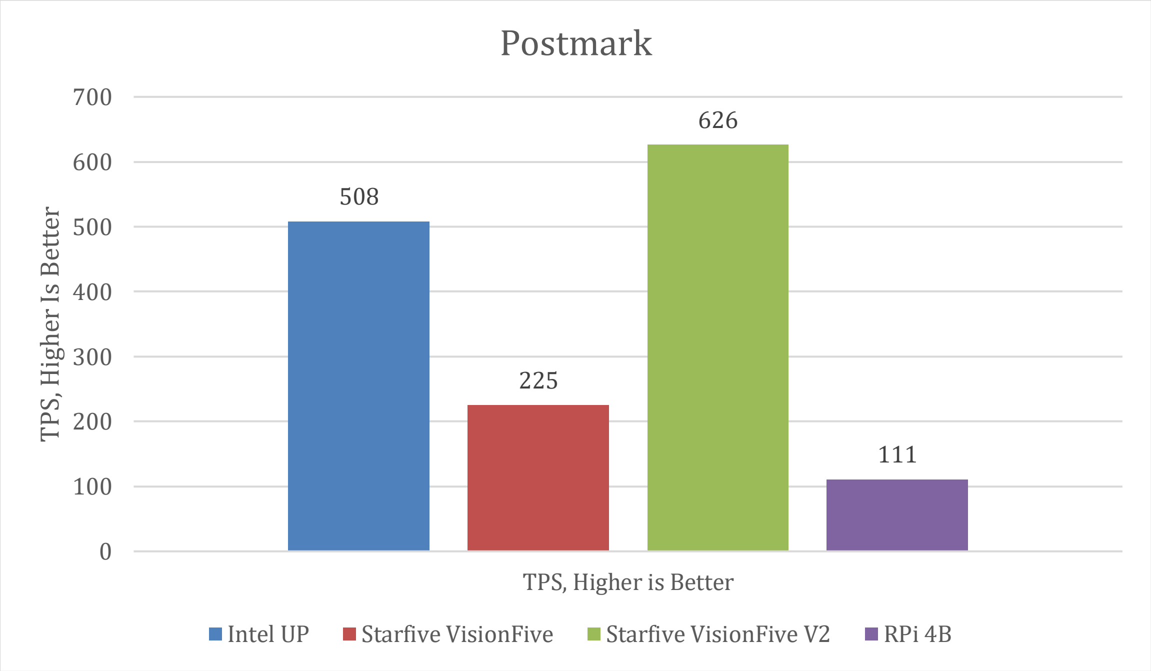


Figure 5. 13 Postmark Results

5.1.7 Multimedia-encoding Benchmark Result

The FLAC (Free Lossless Audio Codec) audio encoding benchmark results suggest that the Raspberry Pi 4B and Intel UP are better suited for efficient FLAC audio encoding tasks. While the StarFive VisionFive v2 shows an improvement over v1, it is still behind Raspberry Pi 4B and Intel UP board in terms of performance.

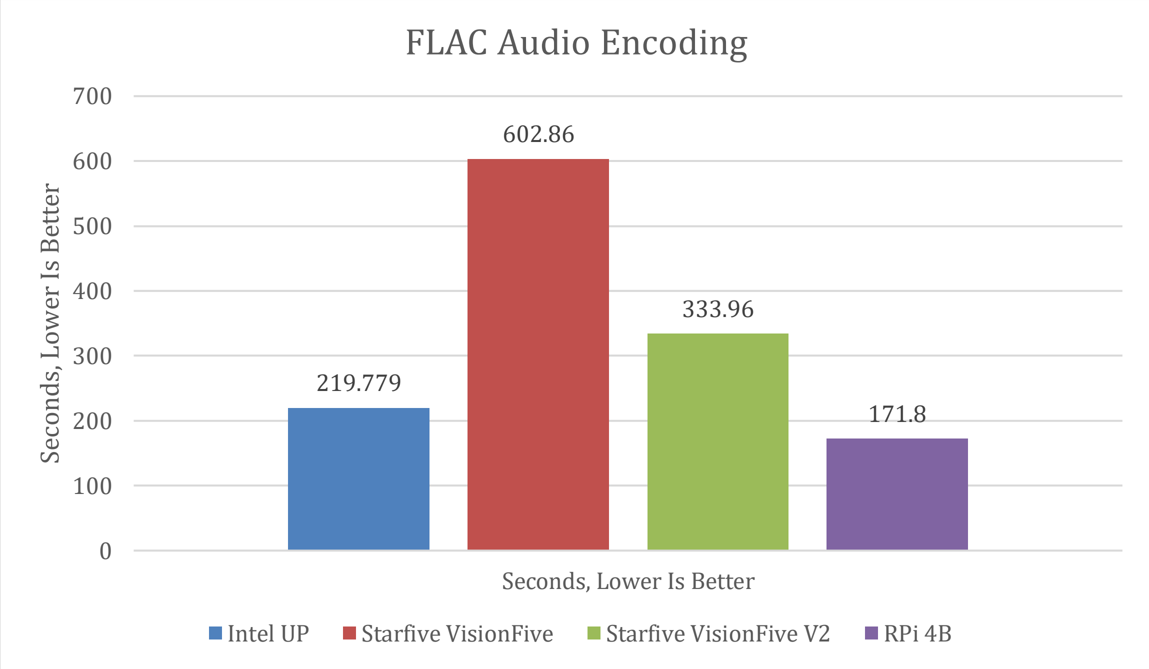


Figure 5. 14 FLAC Audio Encoding Results

In the WebP Image Encoding benchmark, we observed that Raspberry Pi 4B is consistently outperforming the other boards, followed by the Intel UP. While starfive v2 had slight improvements when compared to v1, is still far behind to pose competition in this benchmark.

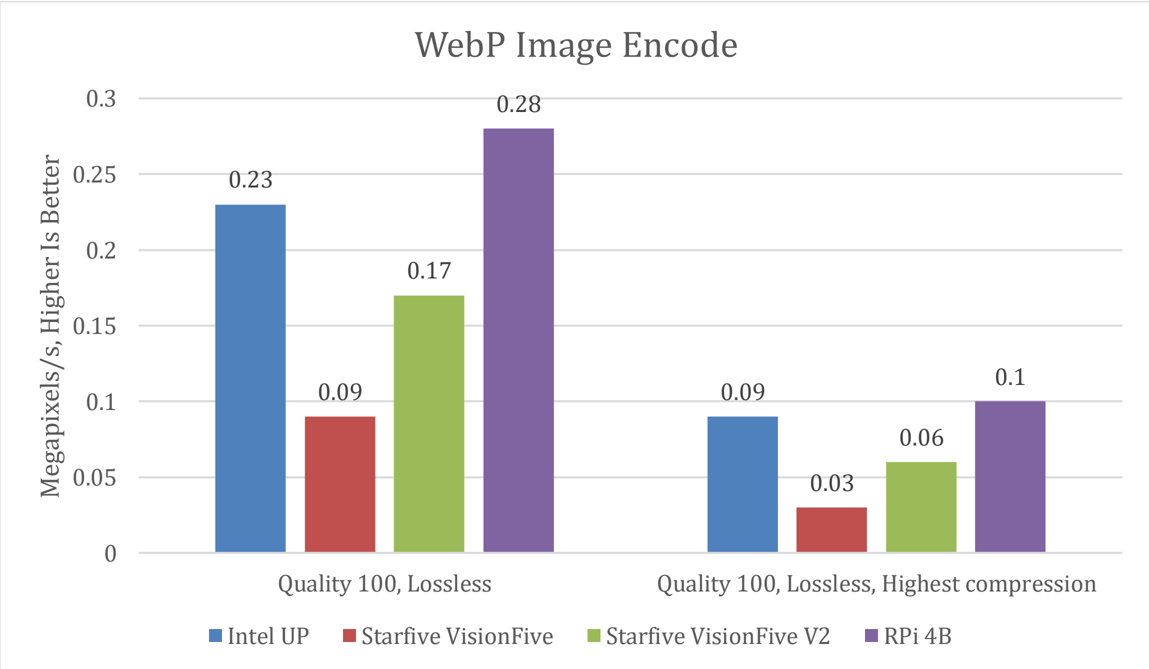


Figure 5. 15 WebP Image Encode Results (1)

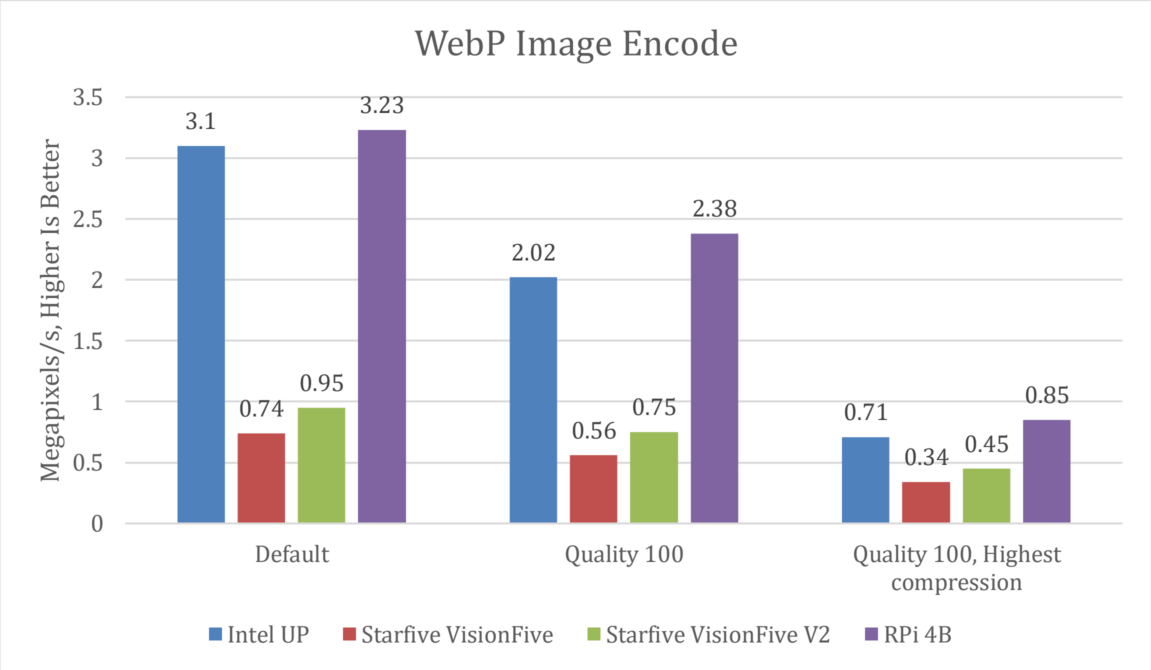


Figure 5. 16 WebP Image Encode Results (2)

Overall, in both FLAC audio encoding and WebP image encoding benchmarks, the Raspberry Pi 4B demonstrates a better performance, followed by the Intel UP, showing their efficiency in executing multimedia encoding tasks. The StarFive VisionFive v2 board shows some improvement compared to v1, but it still lags significantly behind the leading boards, showing its limitation in multimedia encoding. The disparity in performance between the boards may be attributed to differences in CPU performance, memory bandwidth, and optimization of the FLAC encoding algorithm for each platform.

5.1.8 Web Server Benchmark Result

In the PHPBench benchmark, the Raspberry Pi 4B demonstrates the highest performance, followed by the Intel UP, StarFive v2, and StarFive v1, respectively. The Raspberry Pi 4B outperforms the other boards in web server operations, while the Intel UP also exhibits strong performance. The StarFive v2 shows improvement over its predecessor but does not meet standards when compared to Raspberry Pi 4B and Intel UP.

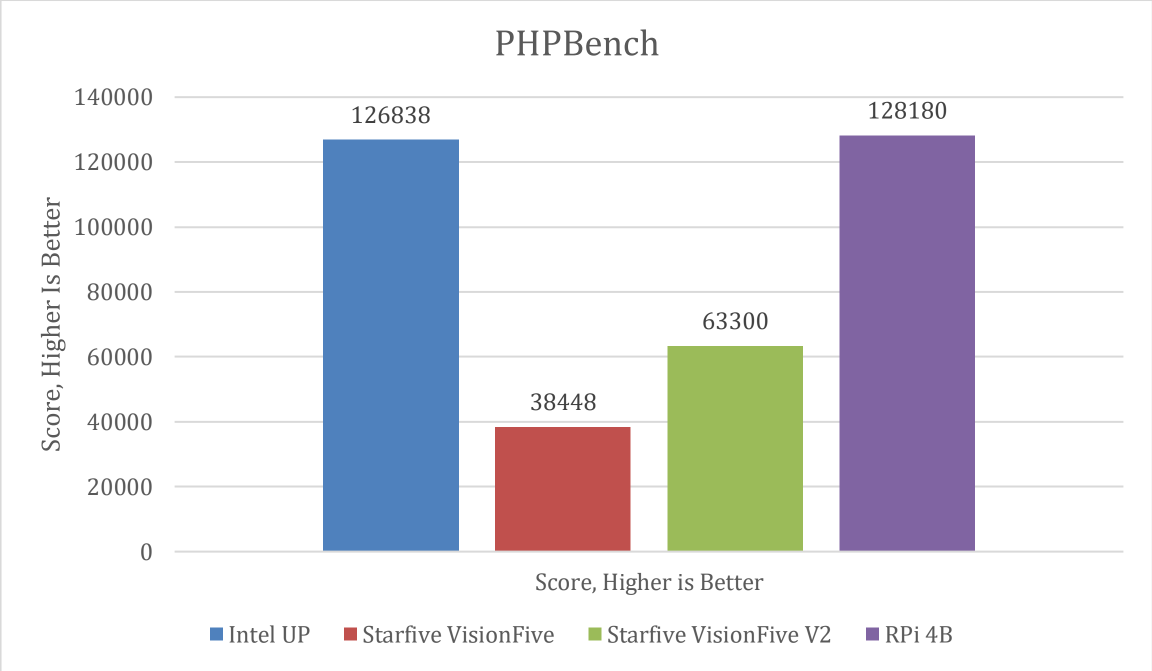


Figure 5. 17 PHPBench Results

5.1.9 Computational Fluid Dynamics Benchmark Result

In the Dolfyn computational fluid dynamics benchmark, the Raspberry Pi 4B demonstrates the highest performance, followed by the Intel UP, StarFive v2, and StarFive v1, respectively. The Raspberry Pi 4B outperforms the other boards in computational fluid dynamics simulations, while the Intel UP also exhibits strong performance. The StarFive v2 shows improvement over its predecessor, but it still falls behind the Raspberry Pi 4B and Intel UP in terms of computational fluid dynamics performance.

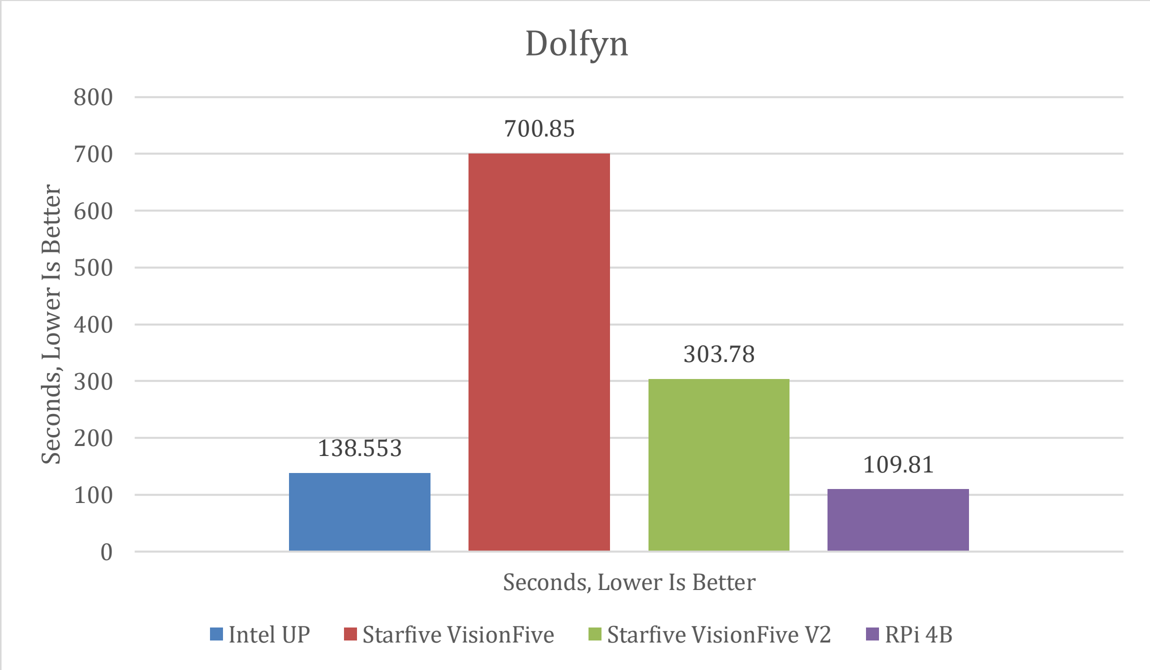


Figure 5. 18 Dolfyn Results

5.1.10 Database Benchmark Result

Overall, in the SQLite speedtest database benchmark, the Intel UP board demonstrates the fastest performance, followed by the StarFive v2, Raspberry Pi 4B, and StarFive v1, respectively. The StarFive v2 board shows notable improvement over its predecessor in database operations, when comparing to other boards, its performance maintains competitive as we can observe its performance is slightly ahead of Raspberry Pi 4B and slightly behind Intel UP board.

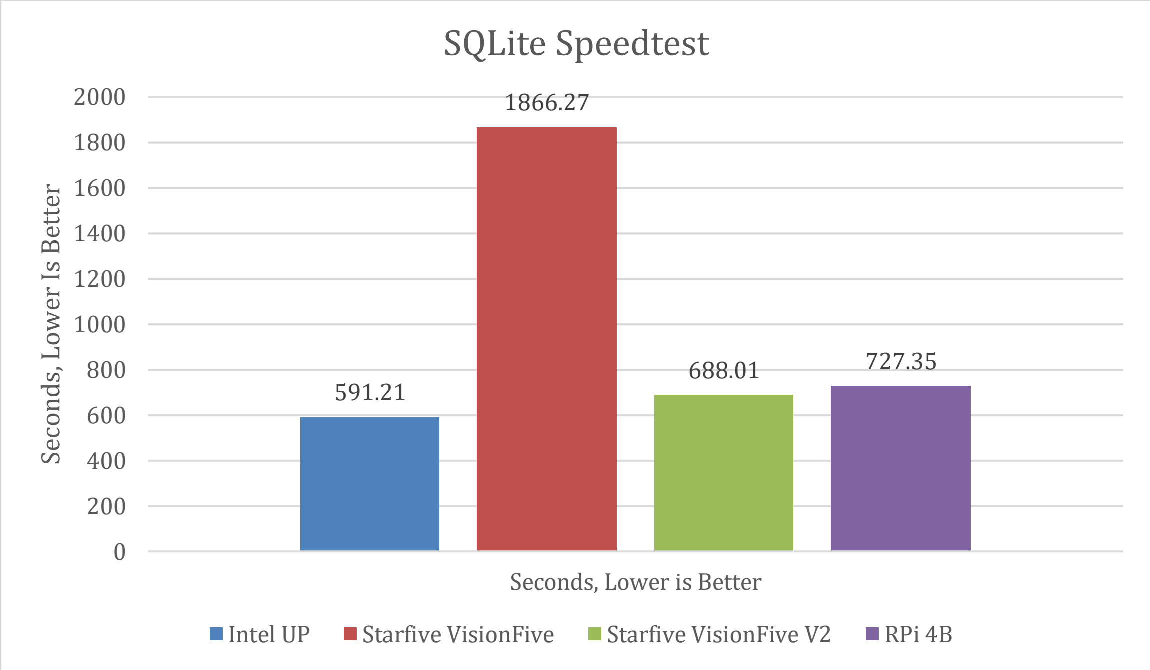


Figure 5. 19 SQLite Speedtest Results

5.1.11 Python Benchmark Result

In the Python benchmark using PyPerformance, where lower time indicates better performance, the results show that starfive v2 achieved significant improvement compared to V1. However, despite this improvement, both versions of the benchmark still lag behind the Intel UP and Raspberry Pi 4B.

This suggests that while there have been advancements and optimizations in Python performance between starfive v2 and v1, they may not be sufficient to surpass the performance of the Intel UP and Raspberry Pi 4B in tasks that include the use of python language.

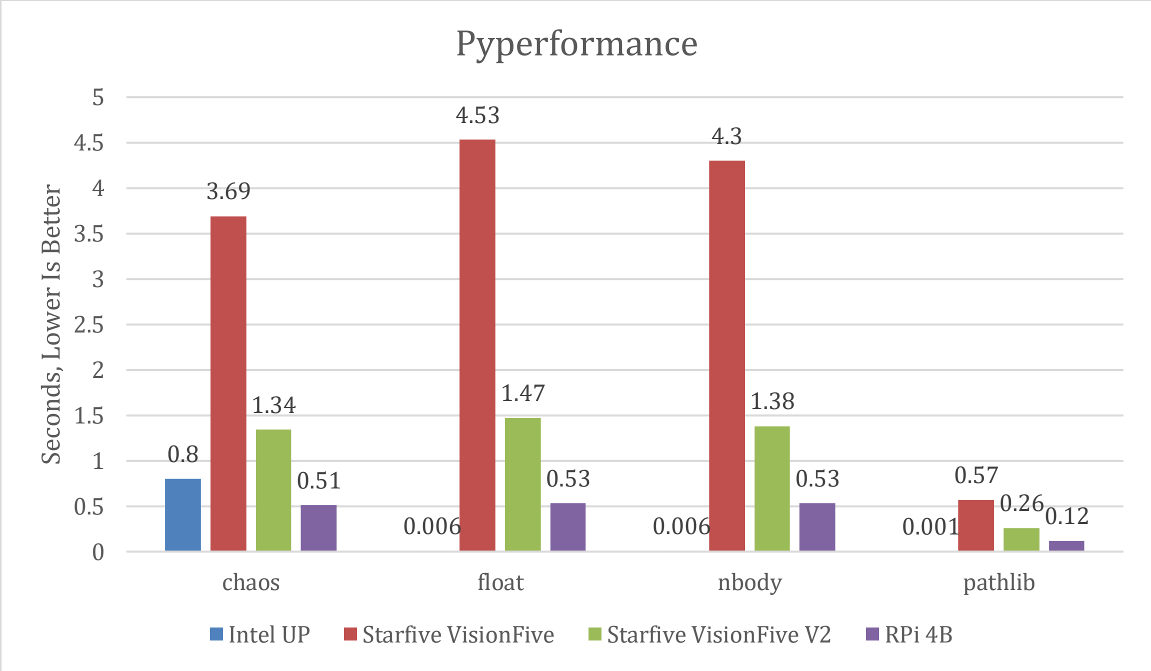


Figure 5. 20 Pyperformance Results (1)

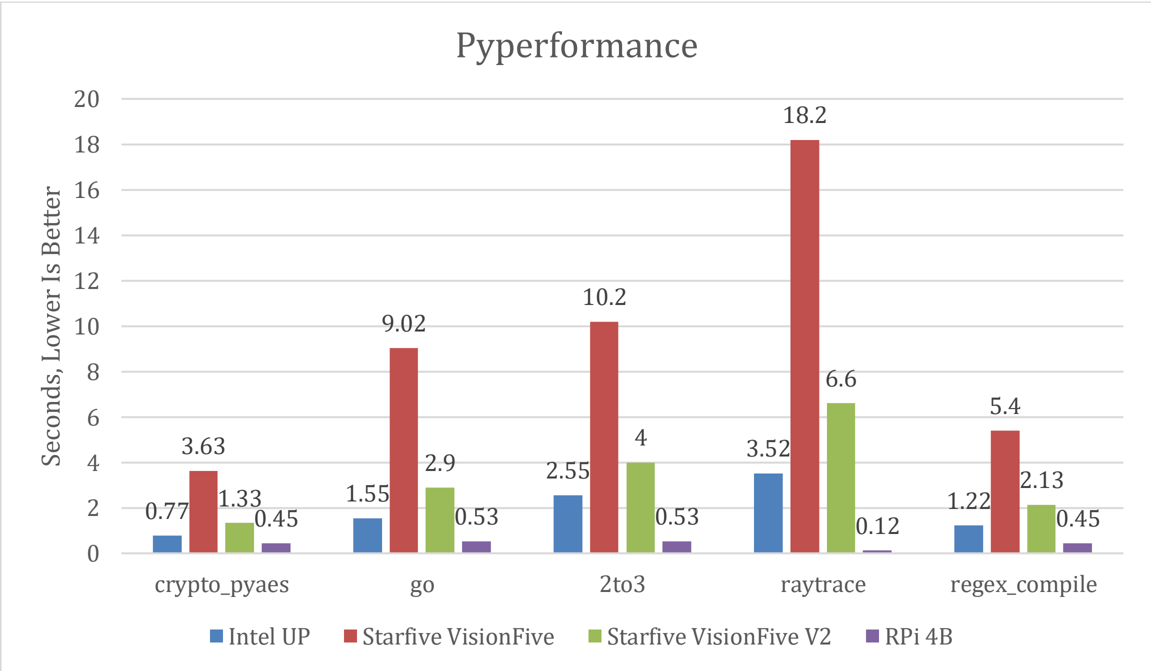


Figure 5. 21 Pyperformance Results (2)

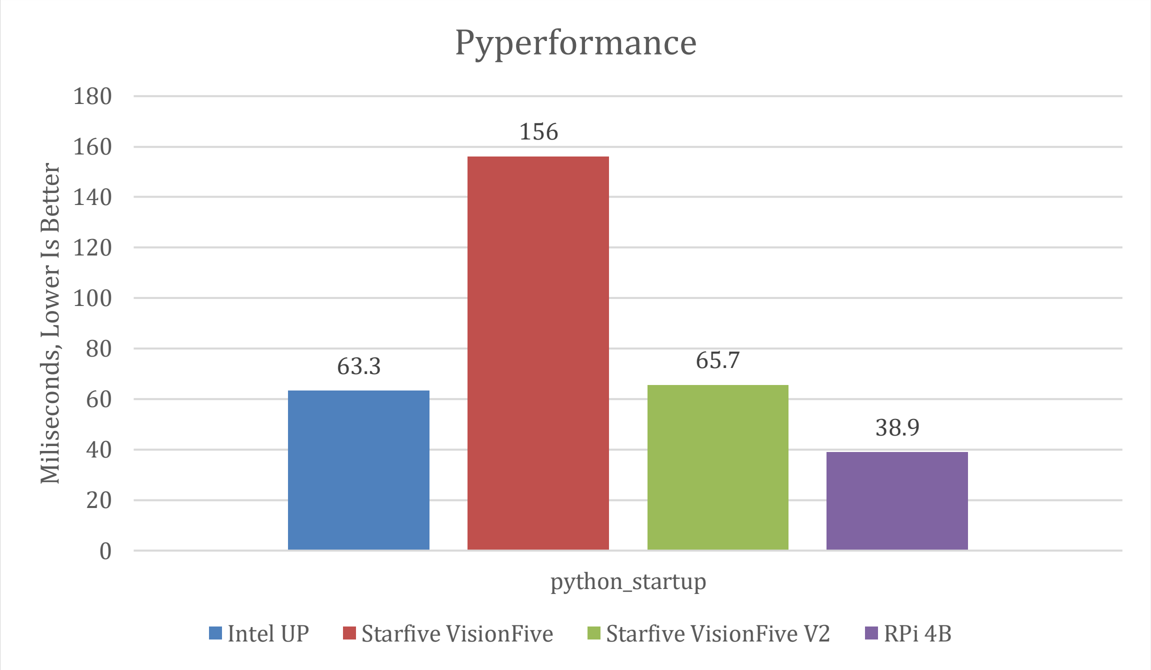


Figure 5. 22 Pyperformance Results (3)

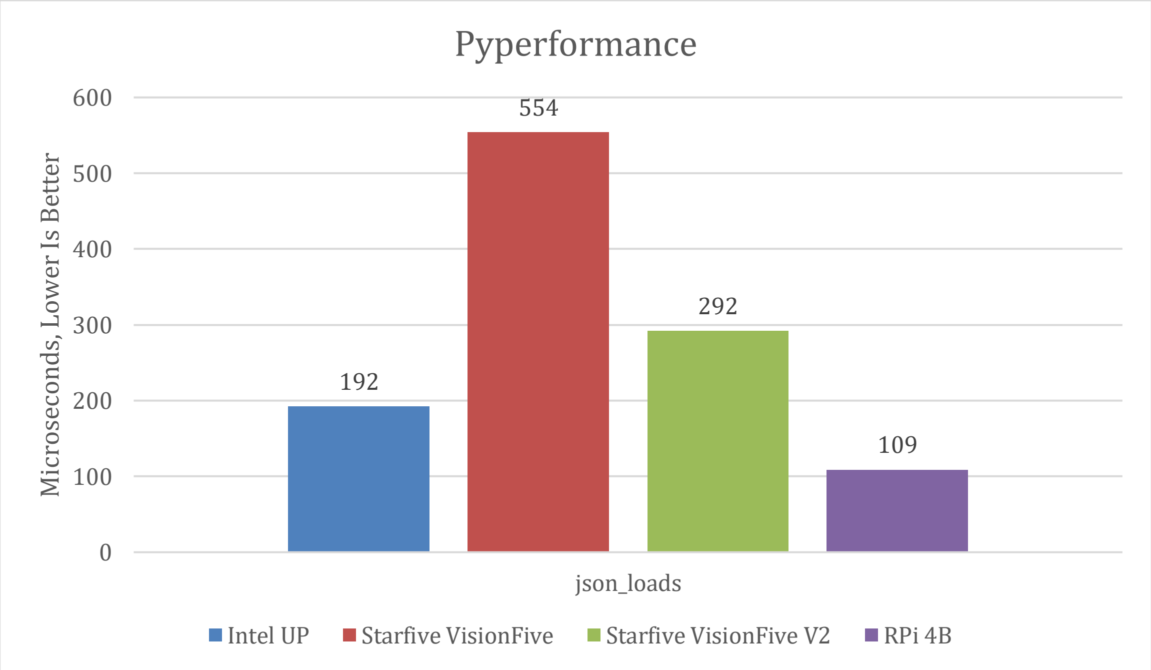


Figure 5. 23 Pyperformance Results (4)

5.1.12 T-test1 Benchmark Result

In the t-test1 benchmark, Intel UP has the fastest performance in both the 1-thread and 2-thread scenarios. In this benchmark, starFive v2 board shows improvement by performing better than its predecessor, but also remains competitive by performing better than Raspberry Pi 4B and being only several seconds behind in performance when compared to Intel UP Board.

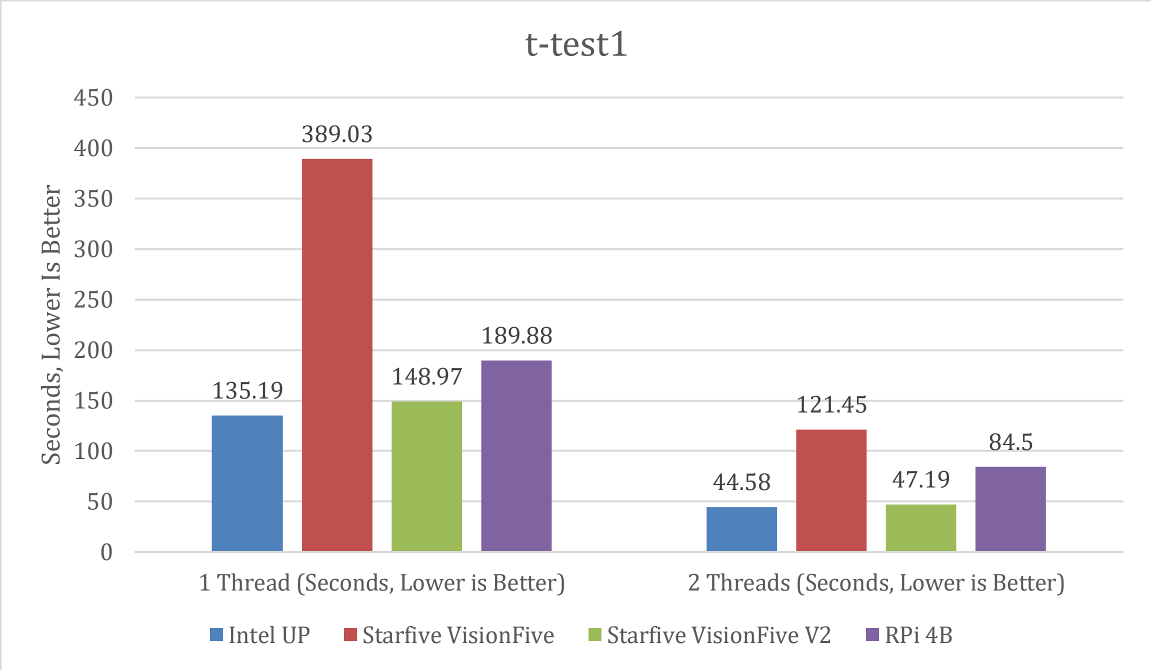


Figure 5. 24 T-Test1 Results

5.2 Power consumption test

The power consumption of the Intel UP board, Raspberry Pi 4B, Starfive Visionfive V1 and Starfive Visionfive V2 was measured. From the power consumption test, we can observe that intel up has the lowest power consumption compared to others.StarFive v2 shows slightly higher power consumption during single-threaded tasks compared to Raspberry Pi 4B. However, as the number of threads increases, StarFive v2 exhibits lower power consumption compared to Raspberry Pi 4B. This suggests that StarFive v2 may be more efficient in multi-threaded workloads.

Despite underperforming in most benchmark tests, StarFive v1 demonstrates lower power consumption compared to both Raspberry Pi 4B and StarFive v2. This indicates that while StarFive v1 may not excel in performance, it offers better energy efficiency, making it suitable for applications where power consumption is a concern.

However, we also observed that in overall benchmarks, the power consumption of StarFive VisionFive V1 between 1 thread and 2 thread is very minimal. We further conducted a test to measure its single thread power consumption and compare it to when one of its cpu is switched off. We observed that even with only one CPU active during the tests, the power consumption is almost consistent when both CPUs are active. This indicates that there is a problem of inefficieny in the board's power consumption.

This phenomenon may be caused by issues such as power management implementation and static power consumption. StarFive v1 may have a power management implementation that does not effectively reduce power consumption when one CPU core is idle. Inefficient power management algorithms or hardware configurations could contribute to this behavior. Certain components of the StarFive v1 platform, such as the memory subsystem, system-on-chip (SoC) components, may consume a significant amount of power even when CPU cores are idle. This static power consumption could contribute to the observed minimal difference in power consumption between one and two active CPU cores.

Overall, the consistent power consumption of StarFive v1 across different CPU configurations suggests potential opportunities for improvement in power management and efficiency. Addressing these issues could lead to better energy efficiency and reduced power consumption, enhancing the overall performance of the platform.

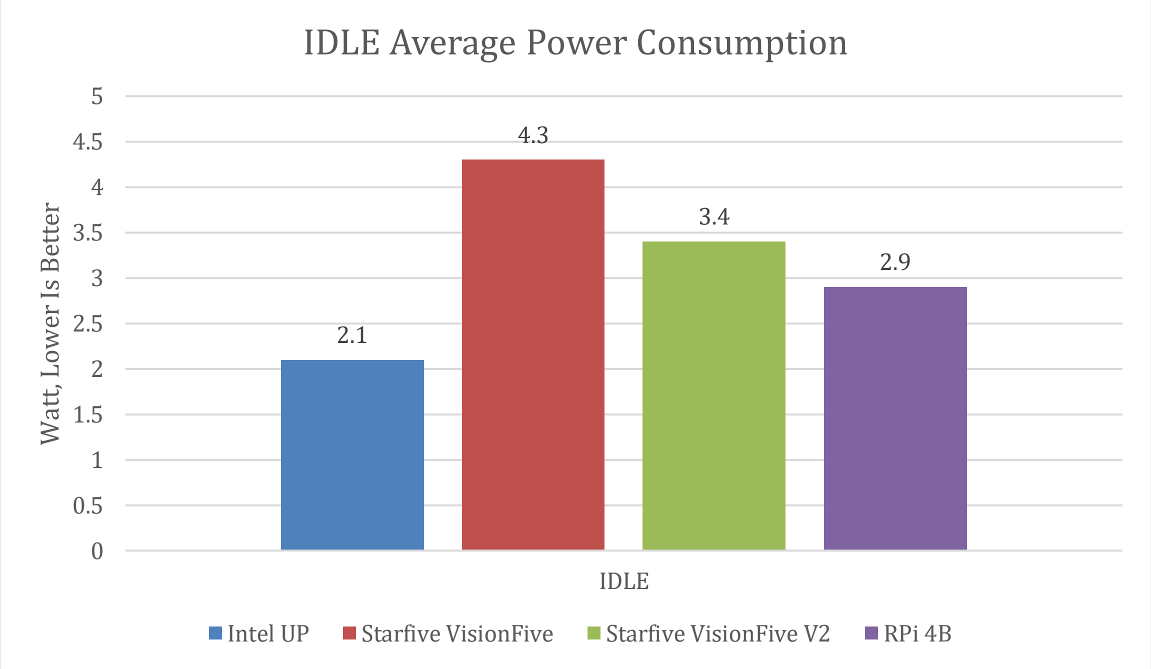


Figure 5. 25 IDLE Average Power Consumption

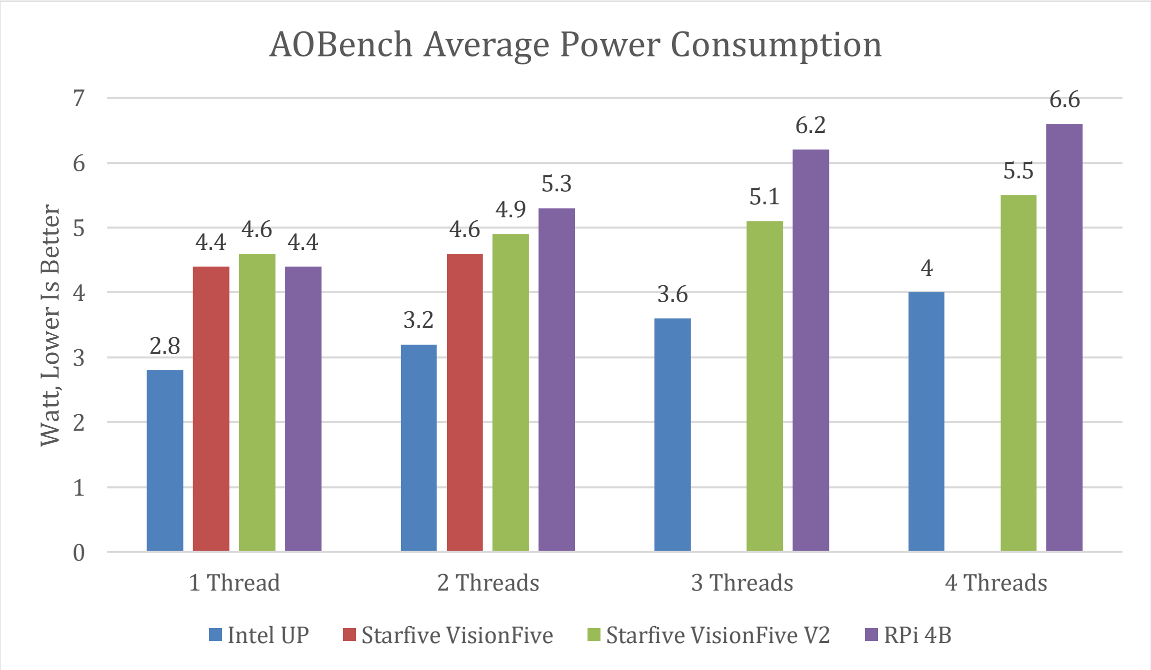


Figure 5. 26 AOBench Average Power Consumption

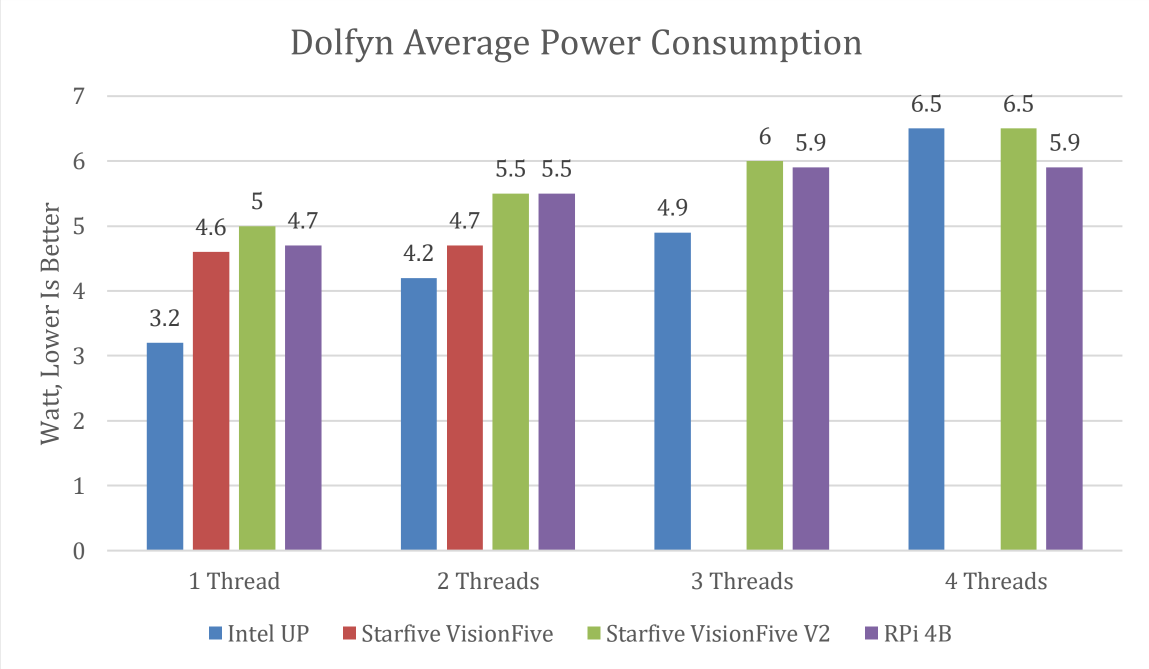


Figure 5. 27 Dolfyn Average Power Consumption

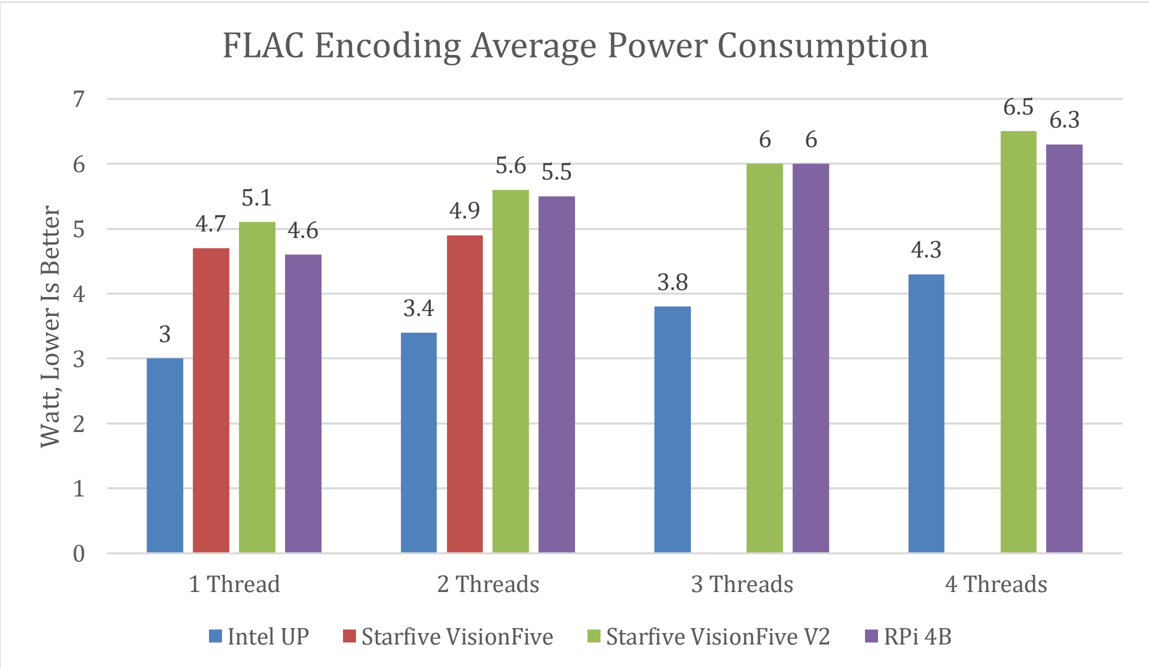


Figure 5. 28 FLAC Audio Encoding Average Power Consumption

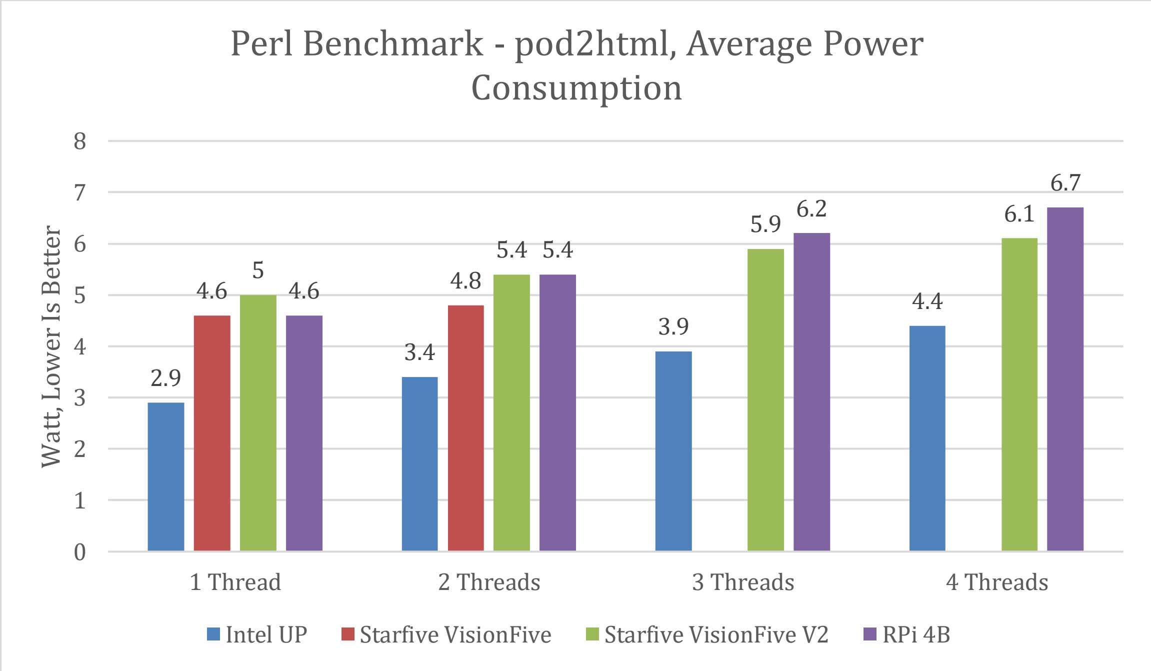


Figure 5. 29 Perl Benchmark – Pod2html Average Power Consumption

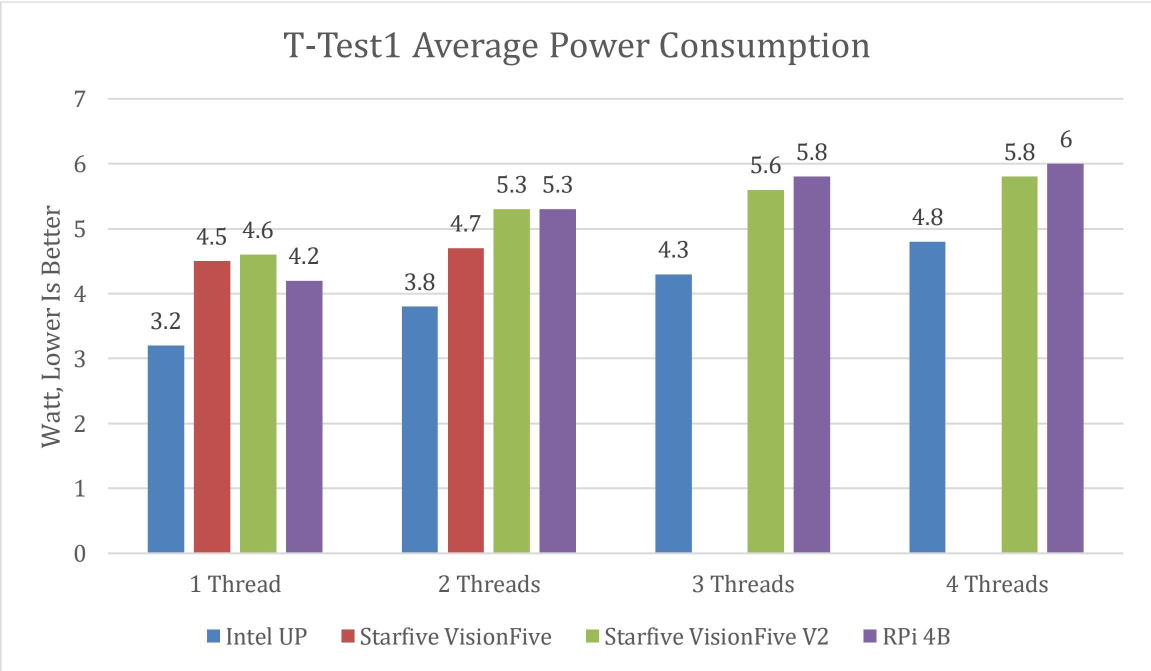


Figure 5. 30 T-Test1 Average Power Consumption

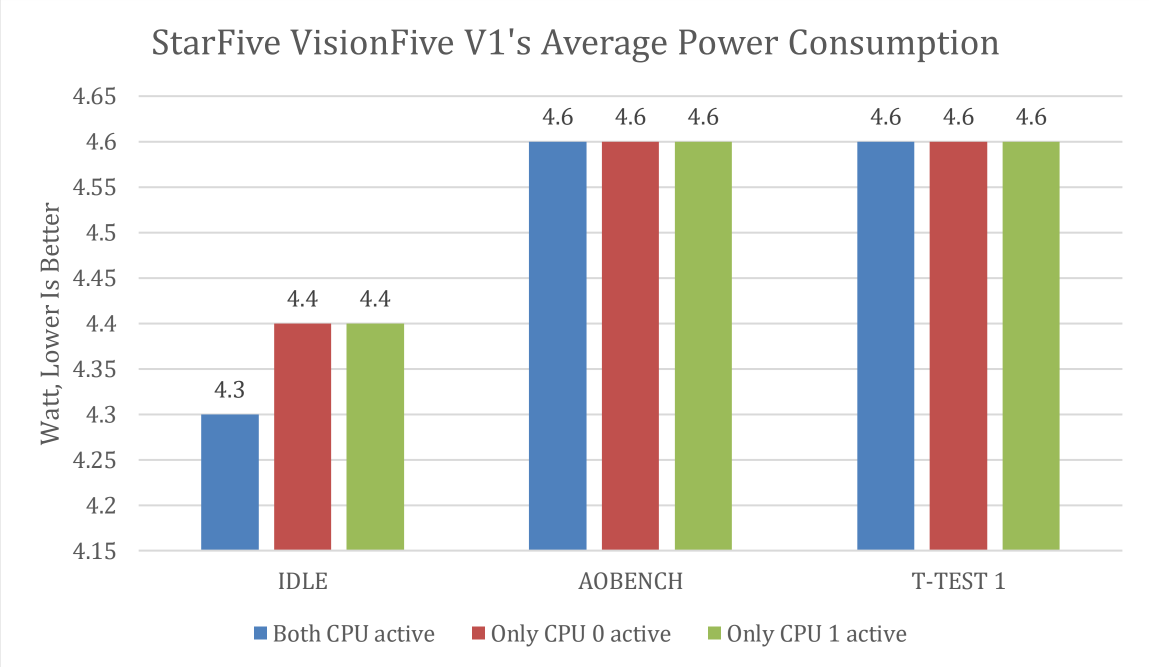


Figure 5. 31 StarFive VisionFive V1's Average Power Consumption

5.3 Machine Learning Benchmarks

NCNN, developed by Tencent, is a high-performance neural network inference framework designed for mobile and various other platforms. The performance metric is the model inference time measured in milliseconds, where lower inference time indicate better performance. NCNN supports two primary inference targets: CPU and Vulkan GPU. Both targets are tested on each model. A total of 17 model options are tested: mobilenet, mobilenet V2, mobilenet V3, shufflenet-v2, mnasnet, efficientnet-b0, blazeface, googlenet, vgg16, resnet18, alexnet, resnet50, yolov4-tiny, squeezenet_ssd, regnety_400m, vision_transformer and FastestDet.

Referring to the figures below, it is observed that the Intel UP board performs significantly better than the other boards. However, when compared to the Raspberry Pi 4B, the StarFive v2 board matches the competition closely in terms of performance, consistently edging ahead in all models except for efficientnet-b0 and vision_transformer. This displays a degree of competitiveness from the StarFive v2, especially when considering its improved performance over its predecessor.

It's worth noting that the StarFive v2 comes with a graphics processor and supports Vulkan, a graphics API noted for its speed and efficiency. Despite these benefits, the results indicate that the performance difference between the target CPU and the target Vulkan GPU is relatively small. This shows that the StarFive v2's CPU performance has the potential to compete with or even outperform its GPU counterpart in certain cases.

In summary, while the Intel UP board performs better overall, the StarFive v2 board is a formidable competitor, particularly when compared to the Raspberry Pi 4B. Its huge improvement over its predecessor, as well as its CPU performance, makes it a potential alternative. Further exploration may be required to understand the details of performance between the CPU and GPU components in various settings.

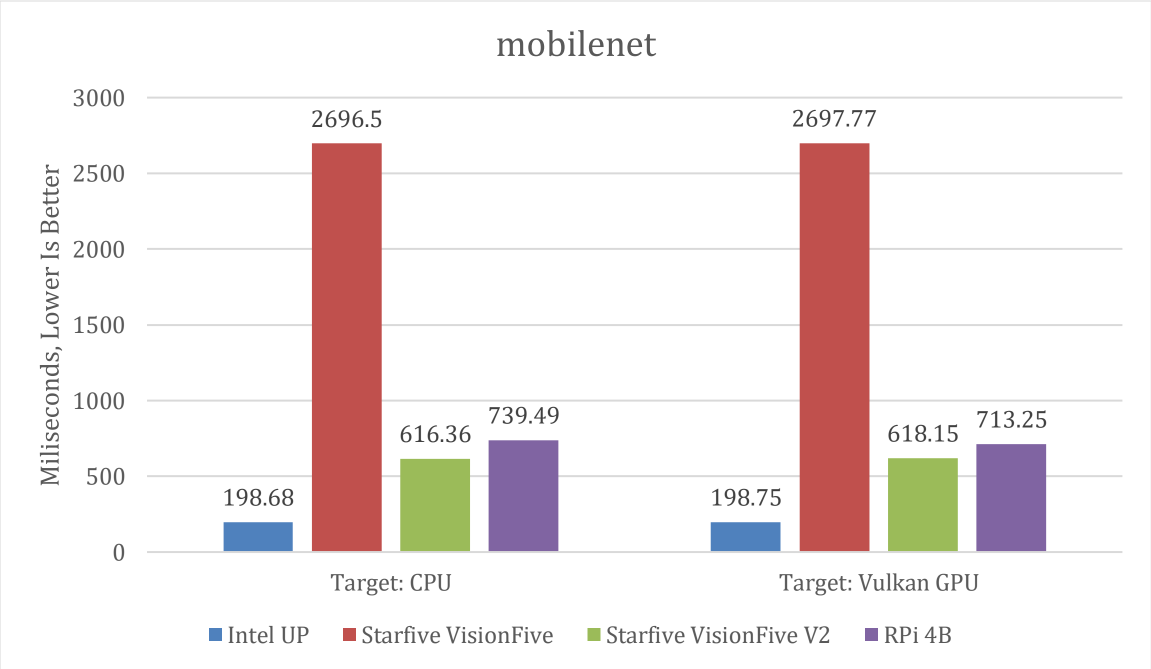


Figure 5. 32 Mobilenet Results

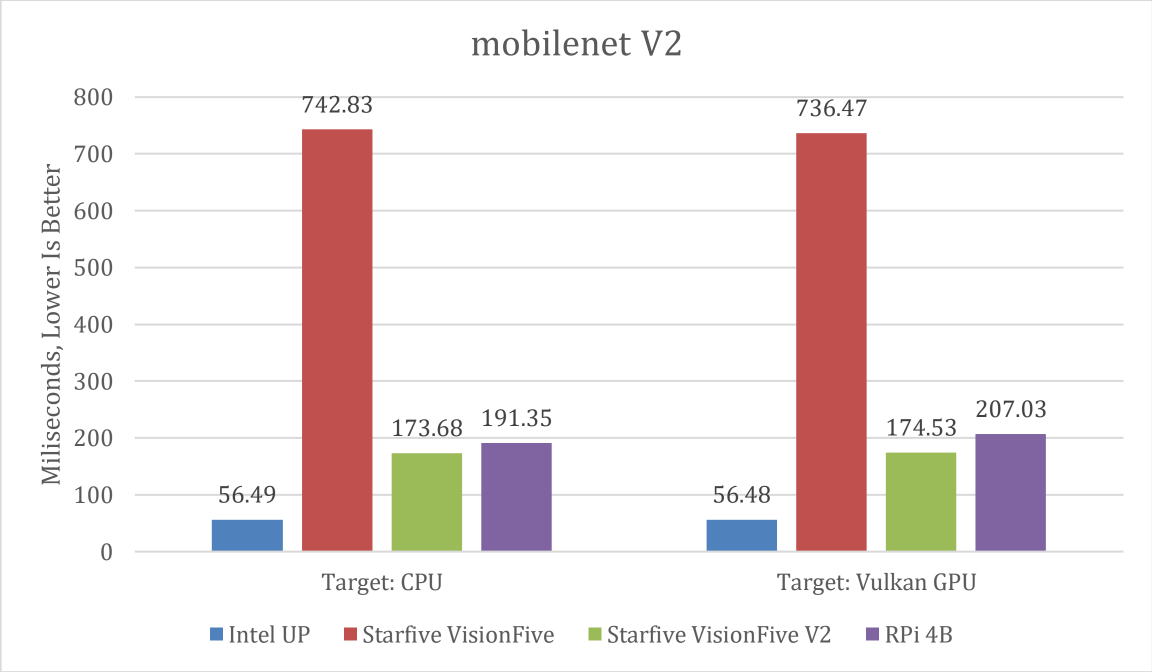


Figure 5. 33 Mobilenet V2 Results

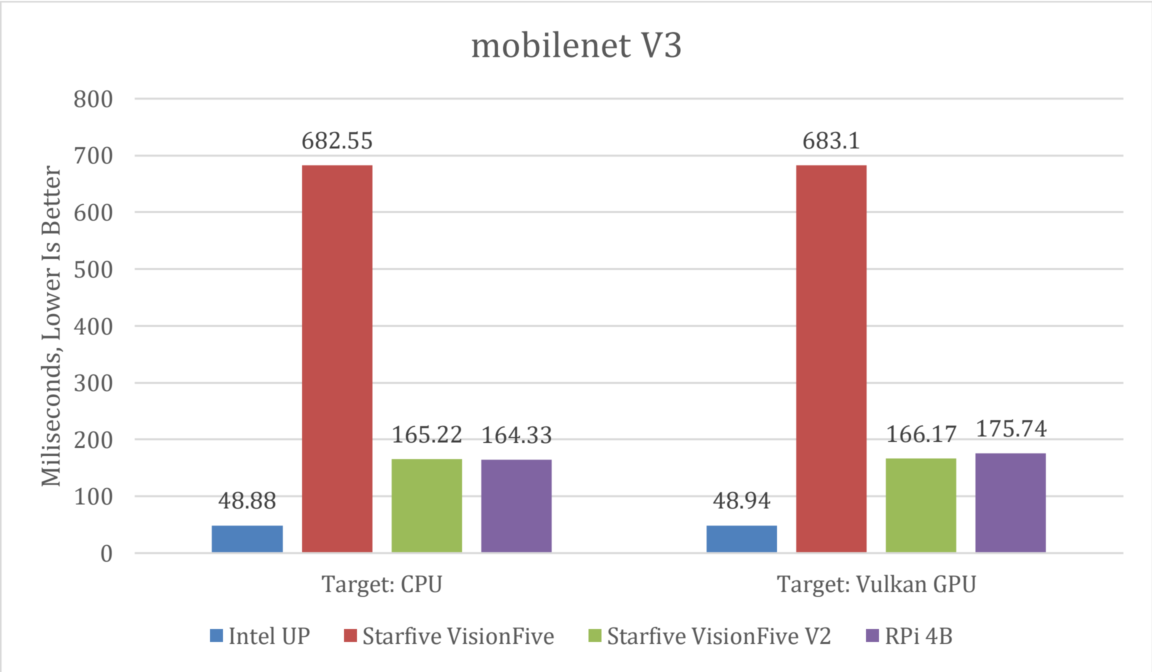


Figure 5. 34 Mobilenet V3 Results

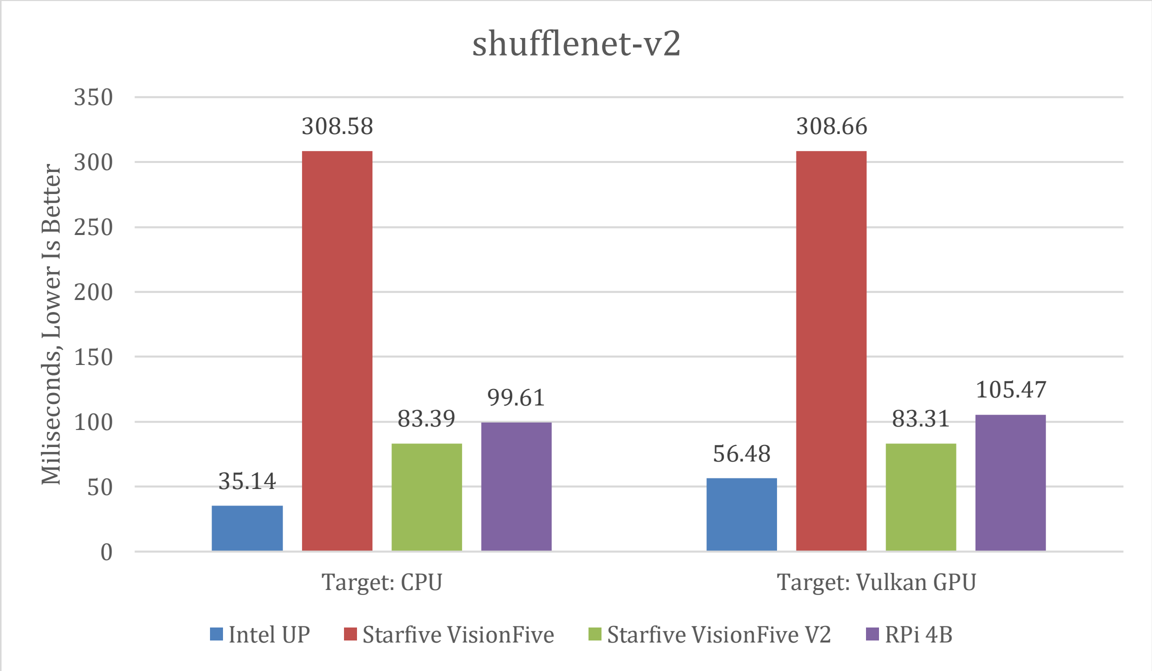


Figure 5. 35 Shufflenet-V2 Results

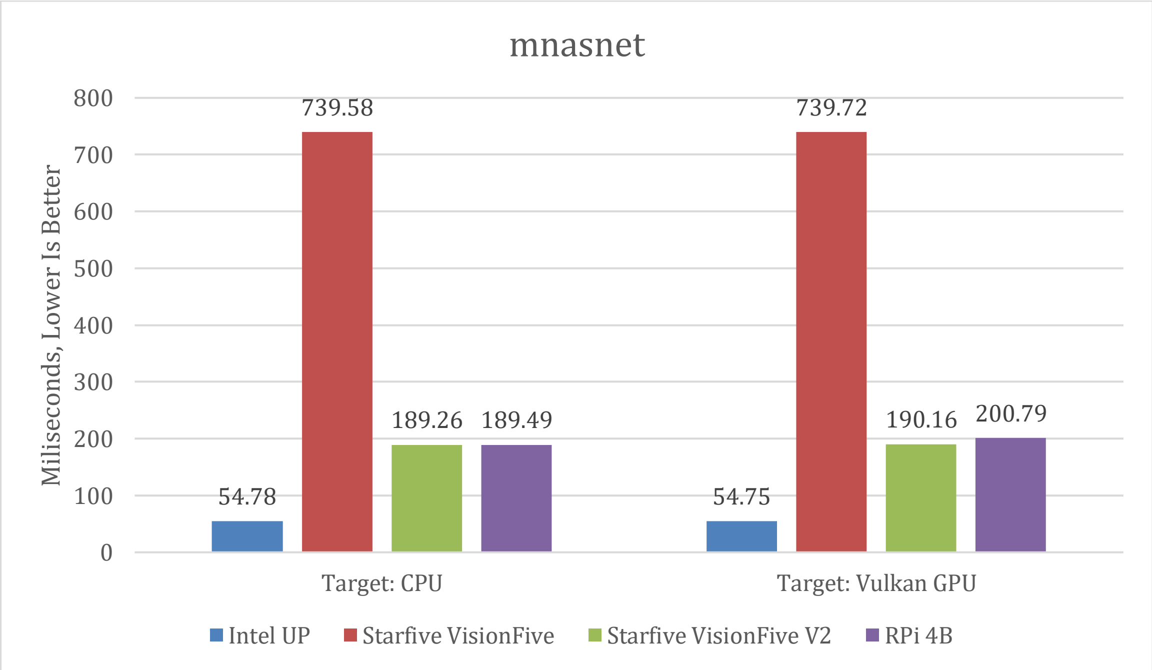


Figure 5. 36 mnasnet Results

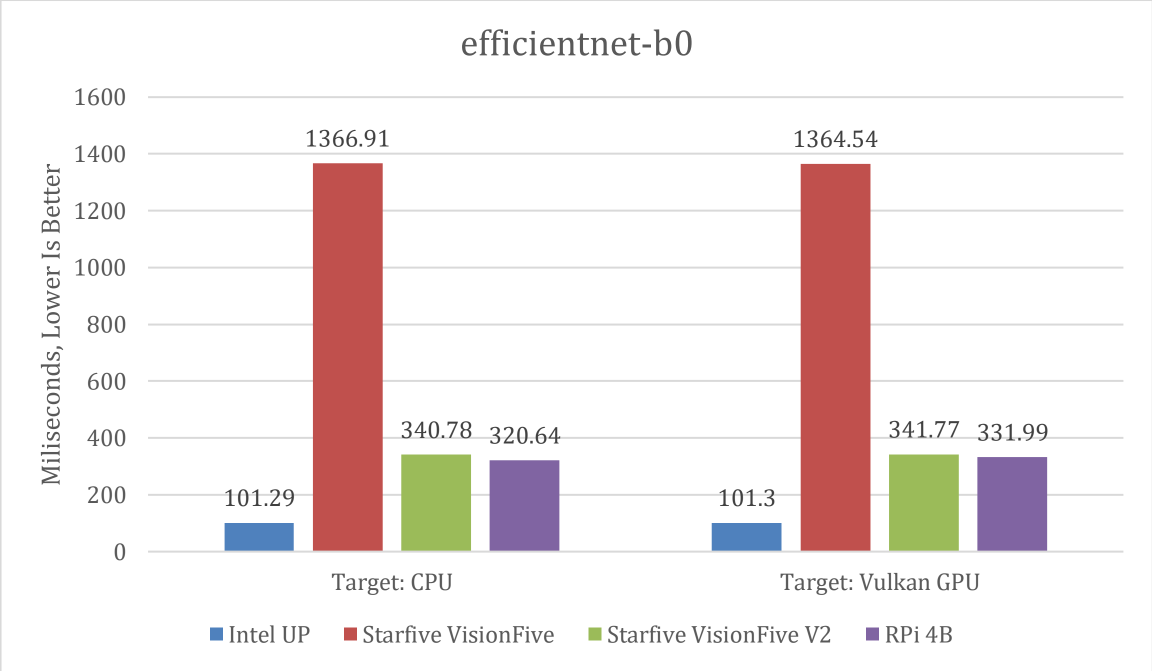


Figure 5. 37 Efficientnet-b0 Results

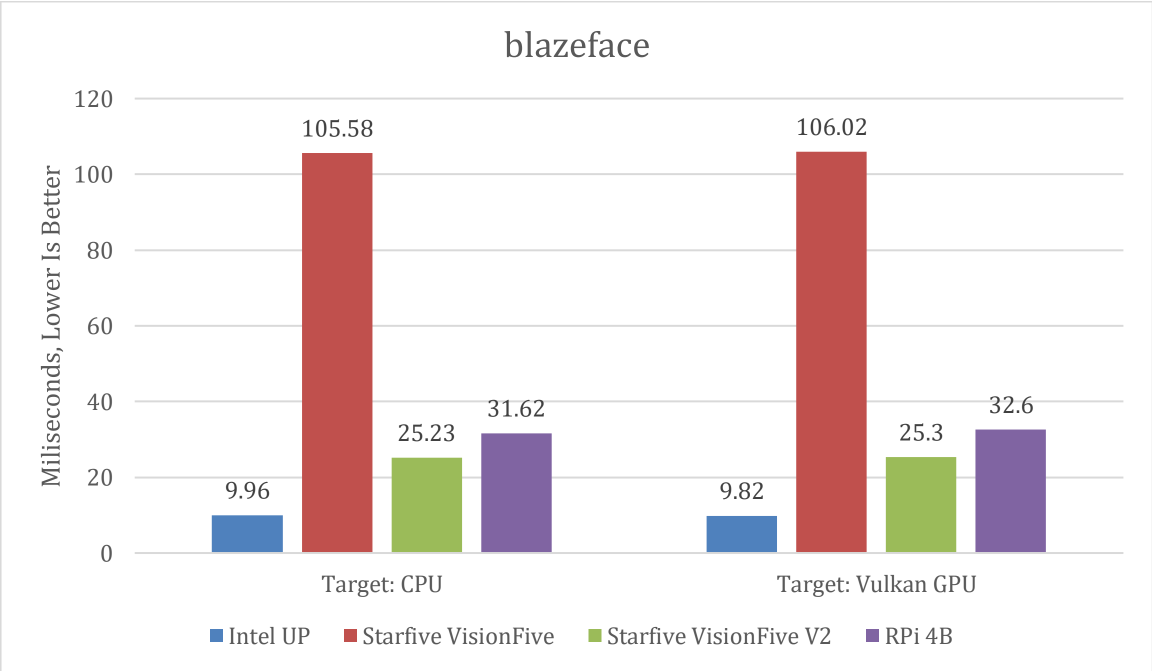


Figure 5. 38 Blazeface Results

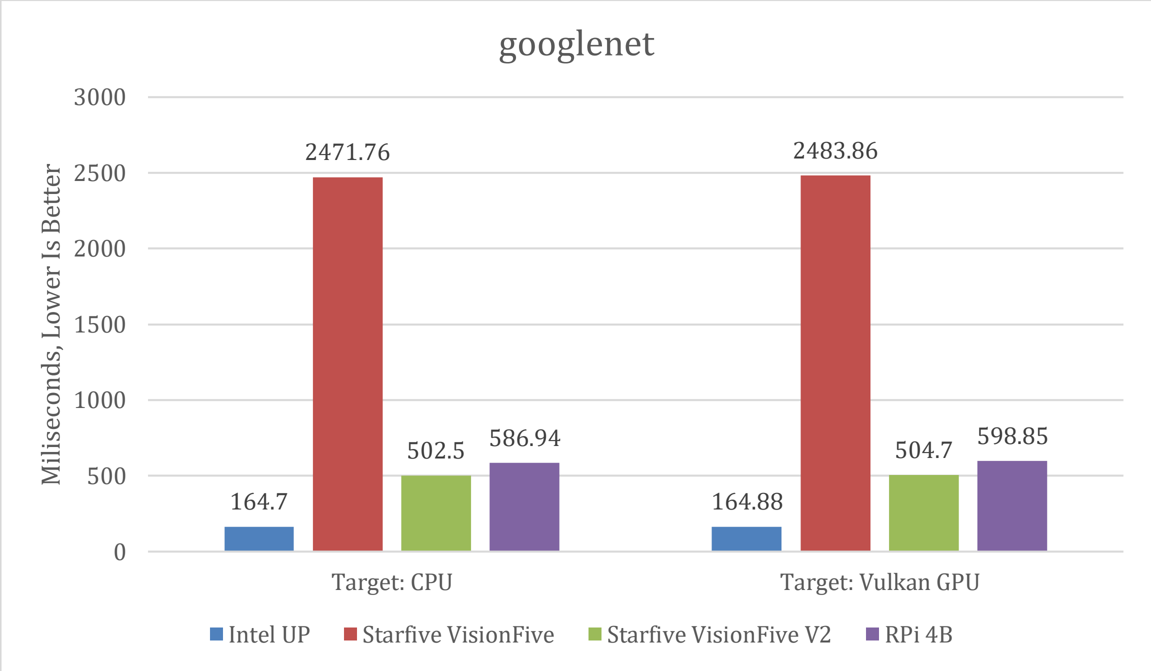


Figure 5. 39 Googlenet Results

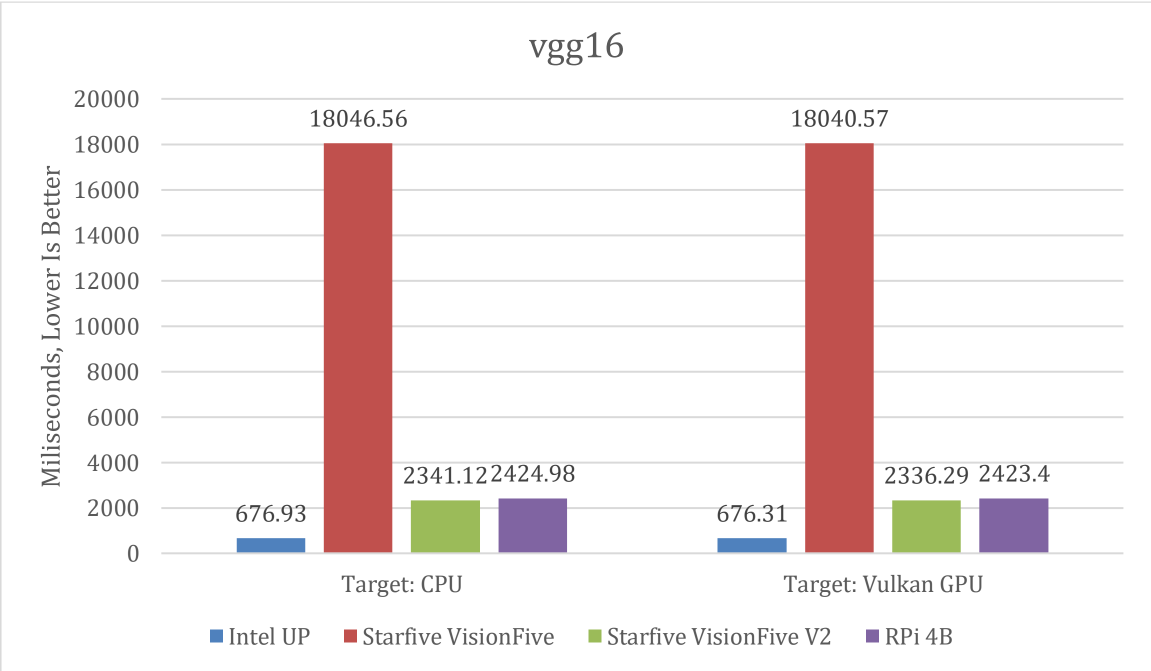


Figure 5. 40 VGG16 Results

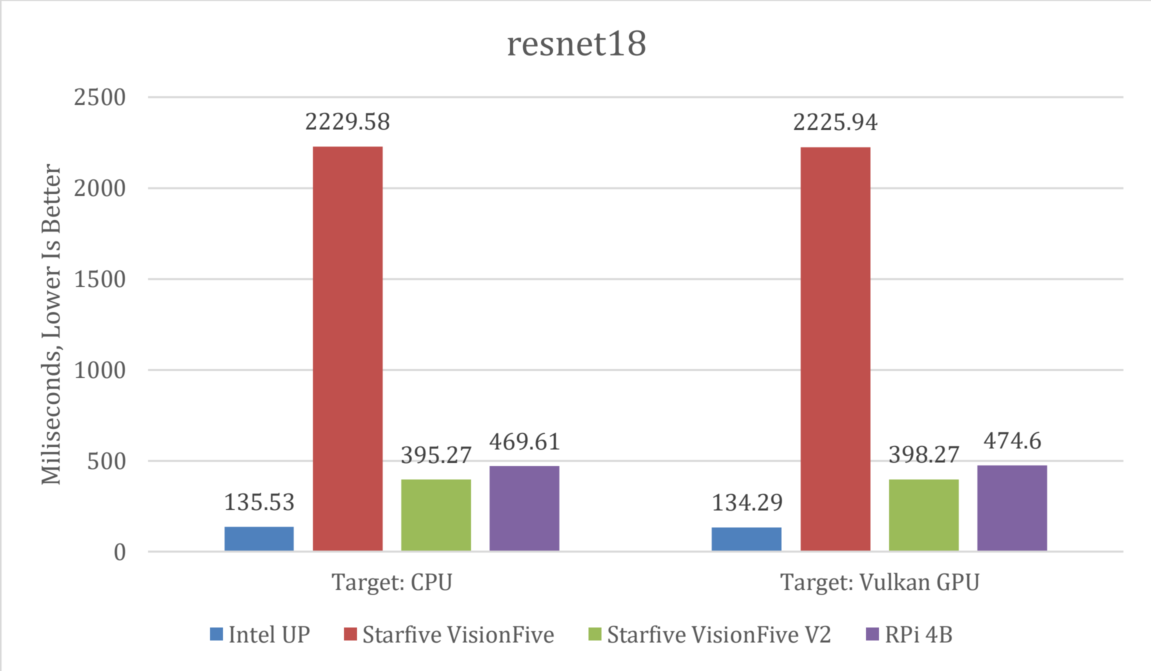


Figure 5. 41 Resnet18 Results

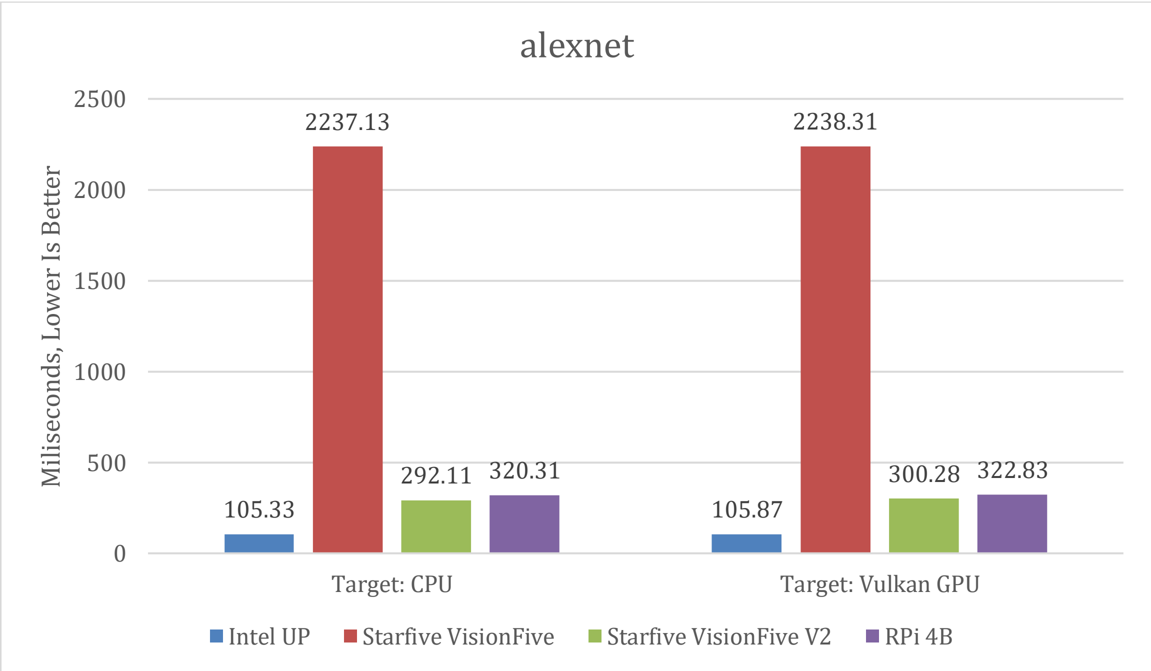


Figure 5. 42 Alexnet Results

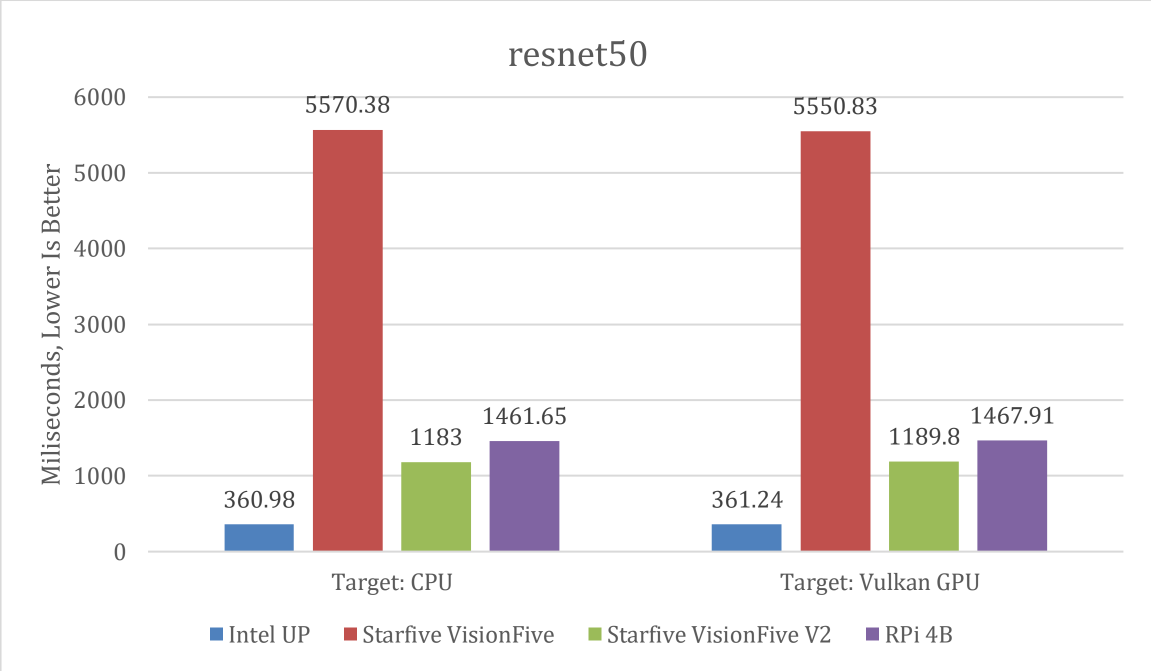


Figure 5. 43 Resnet50 Results

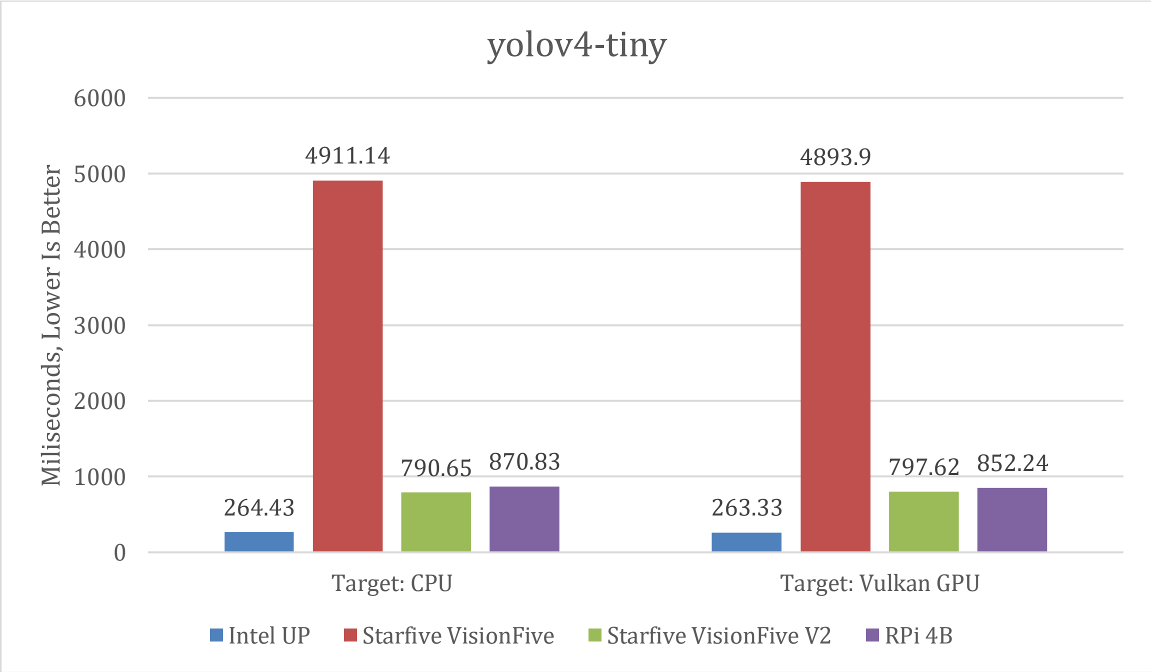


Figure 5. 44 Yolov4-tiny Results

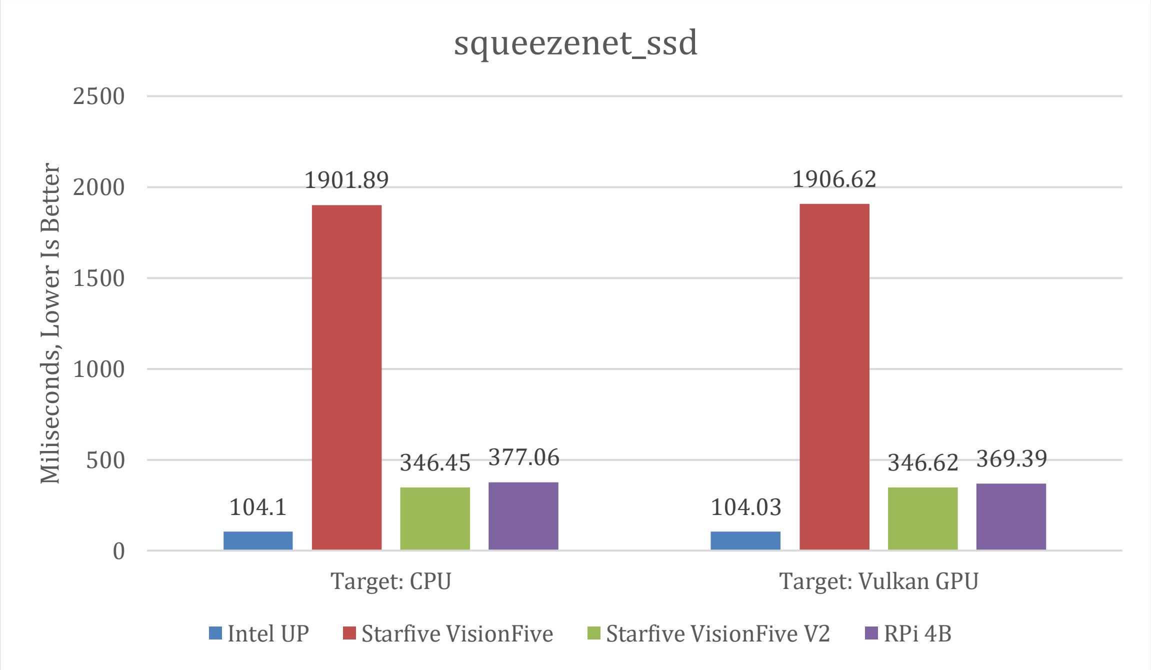


Figure 5. 45 Squeezenet_ssd Results

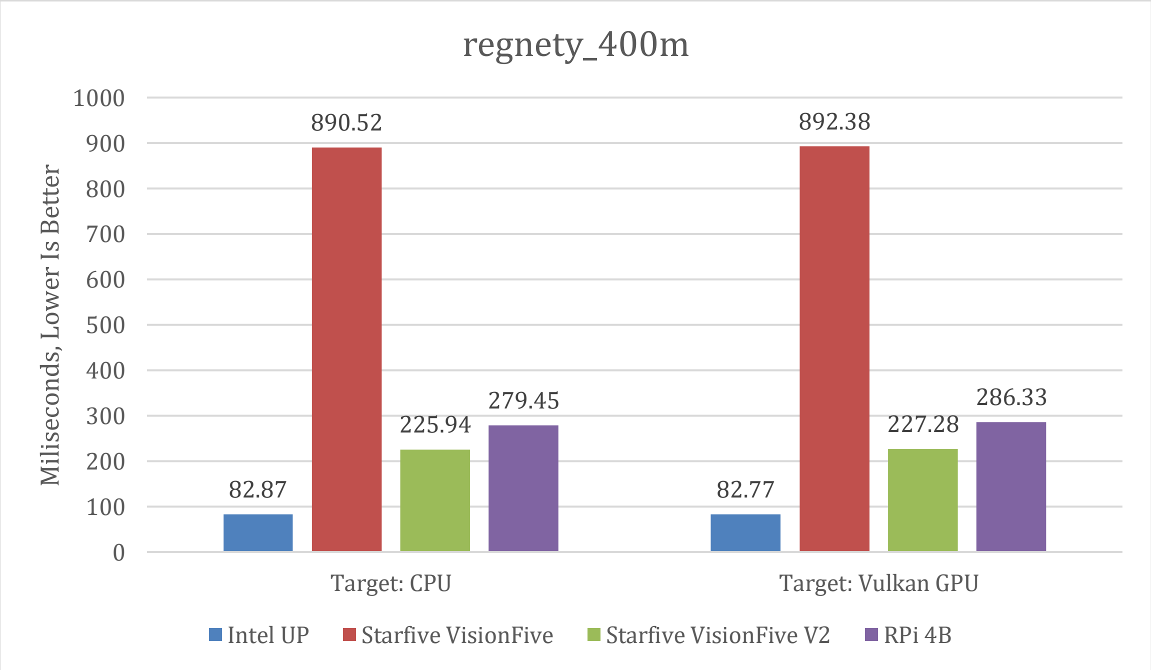


Figure 5. 46 Regnety_400m Results

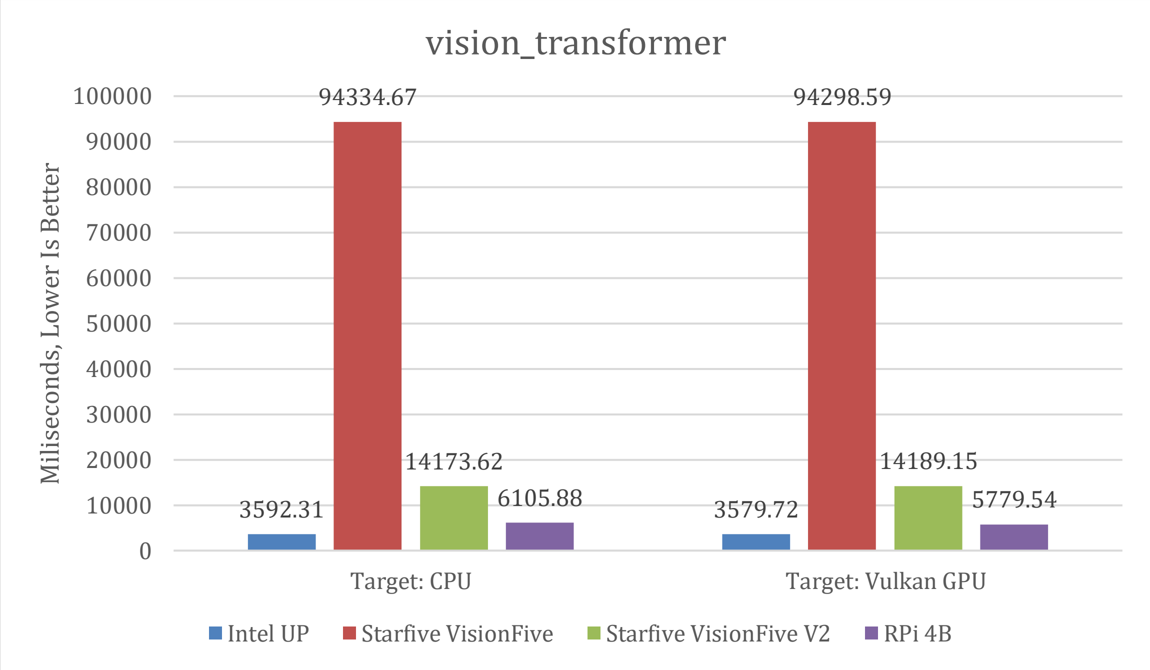


Figure 5. 47 Vision_transformer Results

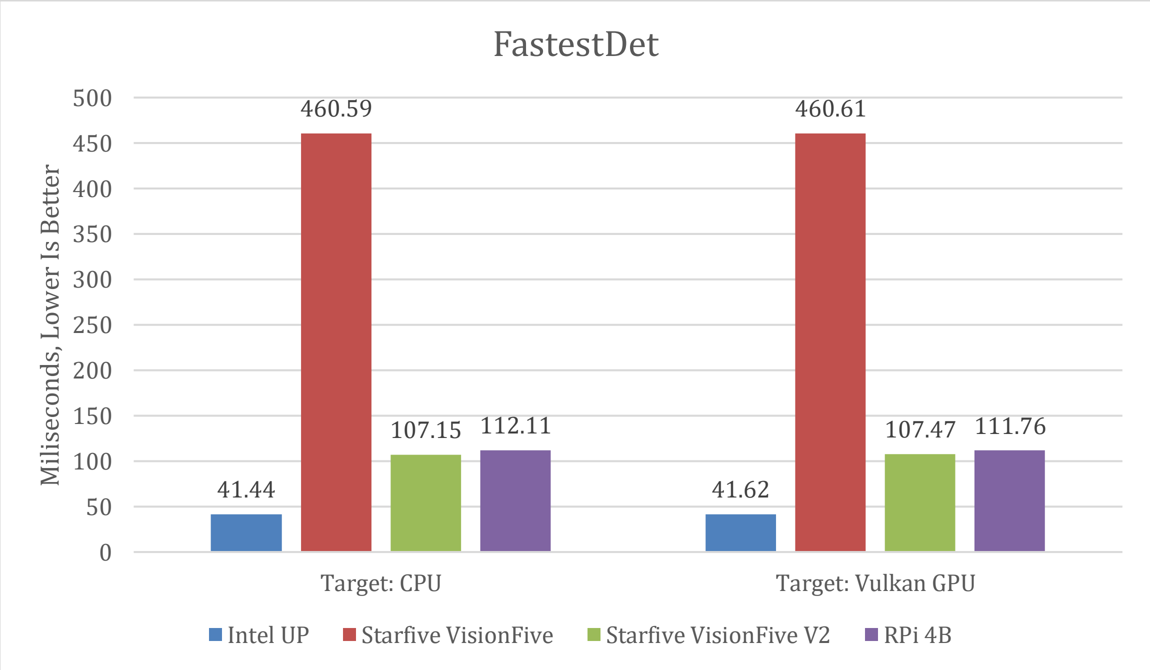


Figure 5. 48 FastestDet Results

5.4 Project Challenges and Limitations

The project's primary goal is to benchmark the performance of hardware boards across a variety of computational tasks and workloads. It has a wide range of benchmarks, including CPU performance evaluation, memory operations, graphics processing, file-system performance, multimedia encoding, web server operations, database operations, Python code execution, and machine learning activities. However, there are potential overlooked use cases that can provide new insights into the boards' capabilities across a wide range of application scenarios. These include IoT and embedded systems applications, high-performance computing (HPC), edge computing, real-time multimedia processing, and big data analytics. These use cases have specific performance needs that may not have been fully addressed during the project.

Furthermore, the machine learning benchmarks is limited due to the lack of PyTorch functionality on the StarFive v2 board. While PyTorch is a widely used and popular framework for machine learning applications, its absence from the StarFive v2 board restricts potential insights and comparisons. A possible solution to this issue is to port PyTorch to the board. PyTorch would be ported by altering the framework to run efficiently on the StarFive v2 board's hardware architecture, allowing researchers to use its capabilities for machine learning. However, porting PyTorch can be a time-consuming and complex process. Therefore, while porting PyTorch could address the compatibility issue, it may introduce additional challenges and resource requirements to the project.

Moreover, it is important to note that benchmarking tests often provide an overview of performance at a specific point in time. Long-term testing to analyze aspects like reliability, stability, and performance degradation over time may have fallen outside the scope of the project. This limitation should be considered when analyzing the project's results and conclusions.

Another challenge encountered during the project was a lack of information about compilation settings such as -O1, -O2, -O3, -mcpu=sifive-u74, and -march=rv64gc. Despite using these flags, the results did not improve, indicating the need for additional research into the optimization settings or potential limitations of the hardware and software configurations.

Overall, while the project provides significant insights into the performance of hardware boards across multiple workloads, addressing these challenges and considering other use cases could improve the benchmarking study's comprehensiveness and applicability.

Chapter 6
Conclusion

6.1 Conclusion

The benchmarking study conducted on the Intel UP Board, StarFive VisionFive V1 and V2 Boards, and Raspberry Pi 4B offers a detailed assessment of their performance across a spectrum of computational tasks and workloads. Here's a deeper dive into the key findings:

StarFive VisionFive V2 shown significant improvements in CPU performance, making it a formidable competition alongside the Raspberry Pi 4B and Intel UP Board. While it lags slightly in memory and cache performance, its competitive GPU capabilities make it a good choice for graphics-intensive applications. However, limitations in multimedia encoding indicate areas for development. StarFive VisionFive V2 excels in filesystem performance, especially for short file transactions, demonstrating efficient concurrency support and data integrity. In database operations, it competes closely with the Raspberry Pi 4B but falls slightly short of the Intel UP board. In the Python benchmark using PyPerformance, the StarFive V2 board outperforms V1, but both versions still lag behind the Intel UP and Raspberry Pi 4B, showing their limitations in Python-related workloads. Nonetheless, in benchmarks such as t-test1, StarFive VisionFive V2 shows tremendous improvement, placing slightly behind the Intel UP board in performance.

The power consumption tests performed on the Intel UP board, Raspberry Pi 4B, StarFive VisionFive V1, and StarFive VisionFive V2 provided interesting findings. The Intel UP board has the lowest overall power usage. While StarFive V2 consumes significantly more power during single-threaded jobs than the Raspberry Pi 4B, it has lower power consumption in multi-threaded applications. Despite underperforming in benchmarks, StarFive V1 has reduced power consumption. However, we noticed small power consumption differences between single-threaded and multi-threaded operations in StarFive V1, additional testing shows consistent power use regardless of CPU configuration, indicating potential power management inefficiencies. This shows that StarFive V1's power management implementation and efficiency could be improved, perhaps improving overall performance and energy efficiency.

During machine learning benchmarks, while the Intel UP board outperforms others significantly, the StarFive v2 was able to outperform Raspberry Pi 4B. Despite StarFive v2's GPU support, its CPU performance remains competitive, with the potential to outperform GPU performance in some cases. Overall, StarFive v2's significant improvement and CPU capability in this field make it a viable option, necessitating additional research into CPU-GPU performance dynamics.

In conclusion, the benchmarking study underscores StarFive VisionFive V2's emergence as a formidable computing platform, offering competitive performance across various tasks. Besides, it showcased a significant improvement compared to its predecessor, StarFive VisionFive V1. Its continued evolution and optimization efforts hold promise for a wide range of applications, making it a compelling choice for developers and researchers seeking robust and efficient computing solutions.

6.2 Recommendations

Based on the challenges and limitations faced during the project, as well as potential overlooked use cases, some recommendations can be considered for future research in this field.

To acquire a more comprehensive understanding of the boards' capabilities, benchmarking in application scenarios such as IoT and embedded systems, high-performance computing (HPC), edge computing, real-time multimedia processing, and big data analytics can be considered. This will provide insights into how the boards work in real-world use cases, allowing users to make more informed decisions based on their specific requirements.

Given the lack of PyTorch capabilities on the StarFive V2 board, porting PyTorch to the hardware architecture is a considerable option. This would allow researchers to use PyTorch's capabilities for machine learning projects while also enabling comparisons with other PyTorch-compatible boards. However, it is important to keep in mind that porting PyTorch may introduce additional challenges and resource requirements into the project.

Future works may conduct long-term testing to examine factors such as reliability, stability, and performance degradation over time. This will provide information about the boards' long-term performance and assist users in determining if they are suitable for long-term use in a wide range of applications.

Graphics processing benchmarks in OpenGL, OpenCL, and Vulkan to test the StarFive V2 board's graphics processor capabilities have potential as the board has accessibility to these utilities. Benchmarking graphics processing will provide useful information about its performance for graphics-intensive activities.

Conduct additional research on optimization parameters such as compilation flags (-O1, -O2, -O3, -mcpu=sifive-u74, -march=rv64gc) and experiment on other compilation flags allows further comprehension of their impact on performance. Investigate the potential limitations of the hardware and software configurations and consider alternate strategies for optimization to improve benchmark results.

By addressing these recommendations, the benchmarking study can become more comprehensive, providing valuable insights into the performance and capabilities of the hardware boards across a wider range of applications and workloads.

REFERENCES

[1] StarFive Tech, "Development Board", StarFive Tech [Online]. Available: https://www.starfivetech.com/en/site/boards

[2] R. Smith, J. Gordon, Oracle.com, 2023. [Online]. Available: https://blogs.oracle.com/linux/post/best-practices-for-adapting-phoronix-test-suite-to-benchmark-linux-performance

[3] "Homepage," Rvspace [Online]. Available: https://rvspace.org/en/home

[4] C. Lattner "RISC-V is the future of computing | Chris Lattner and Lex Fridman." [Online]. Available: https://youtube.com/watch?v=lXdx0X2WHfY

[5] A. Castaneda,"Benchmarking the Wally RISC-V Processor" [Online]. Available: http://smartcities.sites.unlv.edu/wp-content/uploads/2022/10/AbrahamC_REU_FINAL_Poster_PDF.pdf

[6] M. Kimura, "World's First High-Performance RISC-V SBC Integrated with 3D GPU for mass Production--VisionFive 2.", [Online]. Available:https://riscv.or.jp/wp-content/uploads/day3_StarFive_risc-v_tokyo_day2022Autumn.pdf

[7] G. Du "Degree Project in Embedded System Evaluating a RISC-V processor running Benchmarks using the QEMU Virtual Platform tool. GENGWU DU.", [Online]. Available: http://kth.diva-portal.org/smash/get/diva2: 1714139/FULLTEXT01.pdf

[8] J. Geerling "RISC-V Business: Testing StarFive's VisionFive 2 SBC | Jeff Geerling,", [Online]. Available: https://www.jeffgeerling.com/blog/2023/risc-v-business- testing-starfives-visionfive-2-sbc

[9] J. Geerling "StarFive VisionFive 2 - Geekbench Browser," browser.geekbench.com. [Online]. Available: https://browser.geekbench.com/v5/cpu/20379272

[10] J. Chamber "StarFive VisionFive 2 High Performance RISC-V SBC Review,", James A. Chambers [Online]. Available: https://jamesachambers.com/starfive-visionfive2- review/

[11] J. Chamber "Benchmark #66456 - pibenchmarks.com," pibenchmarks.com. [Online]. Available: https://pibenchmarks.com/benchmark/66456/

[12] J. Bennett "The Future Of RISC-V And the VisionFive 2 Single Board Computer,"

Hackaday. [Online]. Available: https://hackaday.com/2023/03/06/the-future-of-risc-v-and-the-visionfive-2-single-board-computer/

[13] B. Johan, "http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1114115," *DIVA*, 2017. [Online]. Available: http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1114115&dswid=-6461

[14] R. P. (Trading) Ltd, "Raspberry Pi 4 Model B specifications,", Raspberry Pi. [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/

[15] UP, "UP Specifications," UP Bridge the Gap. [Online]. Available: https://upboard.org/up/specifications/

[16] InnovatorsGuru, "PZEM-004T V3 Module | Arduino & NodeMCU Code, Circuit, Pinout And Library," innovatorsguru.com. [Online]. Available: https://innovatorsguru.com/pzem-004t-v3/

[17] "pts / Phoronix Test Suite Test Suites - OpenBenchmarking.org," openbenchmarking.org. [Online]. Available: https://openbenchmarking.org/suites/pts

APPENDIX

Appendix A – Benchmark Results of Intel UP Board

```
Perl Benchmarks:
    pts/perl-benchmark-1.0.1 [Test: Interpreter]
    Test 2 of 2
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 4 Minutes [16:57 UTC]
        Started Run 1 @ 16:53:48
        Started Run 2 @ 16:54:51
        Started Run 3 @ 16:56:10
        Started Run 4 @ 16:57:12 *
        Started Run 5 @ 16:58:16 *
        Started Run 6 @ 16:59:36 *
        Started Run 7 @ 17:00:39 *
        Started Run 8 @ 17:01:59 *
        Started Run 9 @ 17:03:16 *
        Started Run 10 @ 17:04:19 *
        Started Run 11 @ 17:05:34 *
        Started Run 12 @ 17:06:40 *
        Started Run 13 @ 17:07:42 *
        Started Run 14 @ 17:08:46 *
        Started Run 15 @ 17:10:00 *

    Test: Interpreter:
        0.0093011889708586
        0.0091348646622475
        0.0083154732455618
        0.0093793000637755
        0.0089576807714855
        0.0093860574675325
        0.0090756748511905
        0.0094172540155631
        0.009201781619151
        0.0085520810041408
        0.0089482129002463
        0.0096850580428962
        0.0084669825545612
        0.0088048137362637
        0.008652721550697

    Average: 0.00901861 Seconds
    Deviation: 4.40%
    Samples: 15

    Comparison of 5,759 OpenBenchmarking.org samples since 4 March 2018; median result: 0 Seconds. B
ox plot of samples:
    [                     |--------------------------------------*------*-------------##*!*#*|*]
                               This Result (8th Percentile): 0.00901861 ^
                                        ARMv7 Cortex-A7: 0.01139 ^  AMD Ryzen 7 5800X: 0.00058 ^
                                                                  AMD Ryzen 7 1800X: 0.00127 ^
                                                              Intel Core i7-6800K: 0.00221 ^
                                                           AMD Athlon II X4 630: 0.00271 ^
```

```
Perl Benchmarks:
    pts/perl-benchmark-1.0.1 [Test: Pod2html]
    Test 1 of 2
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      9 Minutes
    Estimated Time To Completion: 12 Minutes [16:56 UTC]
        Started Run 1 @ 16:45:22
        Started Run 2 @ 16:48:08
        Started Run 3 @ 16:50:54

    Test: Pod2html:
        0.92638711607143
        0.92506870535714
        0.92674470535714

    Average: 0.92606684 Seconds
    Deviation: 0.10%

    Comparison of 2,681 OpenBenchmarking.org samples since 4 March 2018; median result: 0.17 Seconds
. Box plot of samples:
    [                             |--------------------------*--*-------------------#######*#*|* *]
                   This Result (6th Percentile): 0.92606684 ^
                                  Intel Atom x5-Z8350: 0.85818 ^   Intel Core i9-13900K: 0.0555 ^
                                                                   AMD Ryzen 9 5950X: 0.08864 ^
                                                               Intel Core i7-4770S: 0.13939 ^
                                                        POWER9 altivec supported: 0.19376 ^
```

```
PostMark 1.51:
    pts/postmark-1.1.2
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 25 Minutes [14:18 UTC]
        Started Run 1 @ 13:53:53
        Started Run 2 @ 14:02:10
        Started Run 3 @ 14:10:29

    Disk Transaction Performance:
        508
        507
        508

    Average: 508 TPS
    Deviation: 0.11%

    TPS > Higher Is Better
    postmark intel up . 508 |==================================
```

```
RAMspeed SMP 3.5.0:
    pts/ramspeed-1.4.3 [Type: Copy - Benchmark: Integer]
    Test 2 of 10
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      18 Minutes
    Estimated Time To Completion: 2 Hours, 33 Minutes [13:42 UTC]
        Started Run 1 @ 11:09:48
        Started Run 2 @ 11:15:30
        Started Run 3 @ 11:21:12

    Type: Copy - Benchmark: Integer:
        4824.5
        4820.68
        4823.11

    Average: 4822.76 MB/s
    Deviation: 0.04%

    MB/s > Higher Is Better
    Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 4809.42 |==========================================
    ramspeed smp intel up ............................ 4822.76 |==========================================

    Comparison of 5,108 OpenBenchmarking.org samples since 26 February 2011 to 13 February 2022; median result: 17593 MB/s. Box plot of samples:
    [          |--------*-------------------------#############################!###########################------------------------------------------
                         ^ This Result (7th Percentile): 4823
```

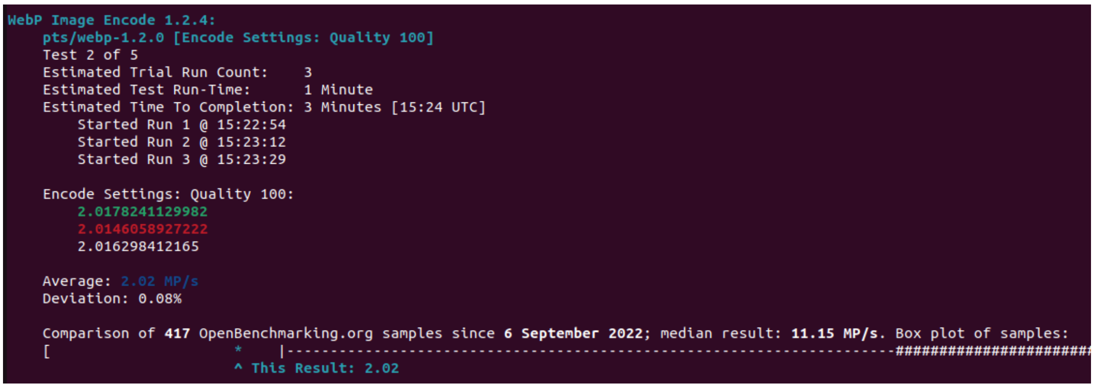
```
Tinymembench 2018-05-28:
    pts/tinymembench-1.0.2
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 19 Minutes [10:04 UTC]
        Started Run 1 @ 09:45:49
        Started Run 2 @ 09:52:41
        Started Run 3 @ 09:59:29

    Standard Memcpy:
        2614.2
        2611
        2610.9

    Average: 2612.0 MB/s
    Deviation: 0.07%

    MB/s > Higher Is Better
    tinymembench intel up ............................ 2612.0 |================================================================================
    Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 2621.1 |================================================================================


    Standard Memset:
        3012.7
        3006.4
        3012.8

    Average: 3010.6 MB/s
    Deviation: 0.12%

    MB/s > Higher Is Better
    tinymembench intel up ............................ 3010.6 |================================================================================
    Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 3025.0 |================================================================================

    Comparison of 472 OpenBenchmarking.org samples since 4 February 2019 to 20 January 2022; median result: 13304 MB/s. Box plot of samples:
    [          *-------------------################!###################----------------------------------------------------------------------|
               ^ This Result (2nd Percentile): 3011
```

```
WebP Image Encode 1.2.4:
    pts/webp-1.2.0 [Encode Settings: Default]
    Test 1 of 5
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      1 Minute
    Estimated Time To Completion: 3 Minutes [15:24 UTC]
        Started Run 1 @ 15:22:09
        Started Run 2 @ 15:22:22
        Started Run 3 @ 15:22:35

    Encode Settings: Default:
        3.0951766830023
        3.0967741935484
        3.0935808197989

    Average: 3.10 MP/s
    Deviation: 0.05%

    Comparison of 467 OpenBenchmarking.org samples since 6 September 2022; median result: 17.52 MP/s. Box plot of samples:
    [                  *  |----------------------------------------------------------------#########################!#####
                       ^ This Result: 3.10
```

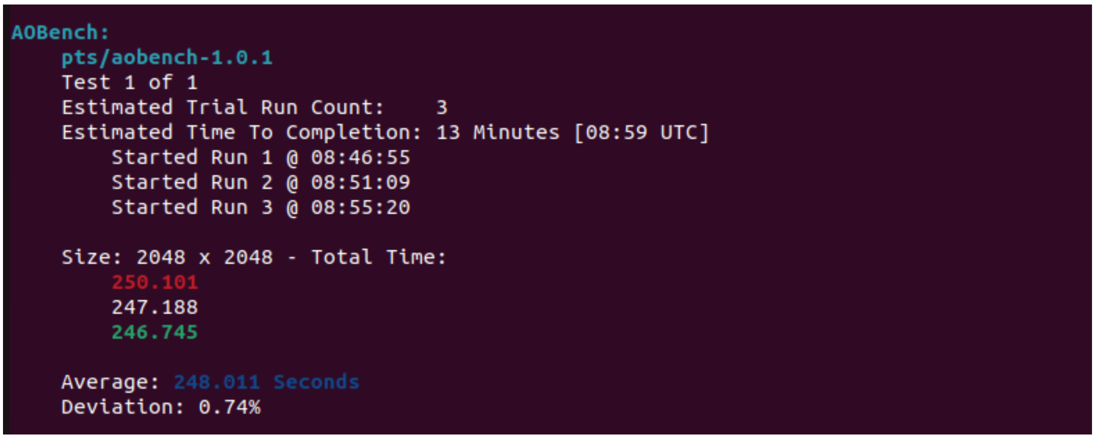
```
WebP Image Encode 1.2.4:
    pts/webp-1.2.0 [Encode Settings: Quality 100, Highest Compression]
    Test 4 of 5
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      2 Minutes
    Estimated Time To Completion: 4 Minutes [15:32 UTC]
        Started Run 1 @ 15:29:25
        Started Run 2 @ 15:30:04
        Started Run 3 @ 15:30:43

    Encode Settings: Quality 100, Highest Compression:
        0.71155385573246
        0.71269487750557
        0.71206052514464

    Average: 0.71 MP/s
    Deviation: 0.08%

    Comparison of 407 OpenBenchmarking.org samples since 6 September 2022; median result: 3.48 MP/s. Box plot of samples:
    [                    *            |-----------------------------------------------------------###############!###########
                         ^ This Result: 0.71
```

```
WebP Image Encode 1.2.4:
    pts/webp-1.2.0 [Encode Settings: Quality 100, Lossless, Highest Compression]
    Test 5 of 5
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 2 Minutes [15:33 UTC]
        Started Run 1 @ 15:31:28
        Started Run 2 @ 15:35:57
        Started Run 3 @ 15:40:25

    Encode Settings: Quality 100, Lossless, Highest Compression:
        0.091079865657198
        0.091359660142064
        0.091305786884685

    Average: 0.09 MP/s
    Deviation: 0.16%

    Comparison of 312 OpenBenchmarking.org samples since 9 September 2022; median result: 0.61 MP/s. Box plot of samples:
    [|!|
```

```
WebP Image Encode 1.2.4:
    pts/webp-1.2.0 [Encode Settings: Quality 100, Lossless]
    Test 3 of 5
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      1 Minute
    Estimated Time To Completion: 2 Minutes [15:25 UTC]
        Started Run 1 @ 15:23:52
        Started Run 2 @ 15:25:41
        Started Run 3 @ 15:27:30

    Encode Settings: Quality 100, Lossless:
        0.23129186141763
        0.23181014748921
        0.23147924884984

    Average: 0.23 MP/s
    Deviation: 0.11%

    Comparison of 386 OpenBenchmarking.org samples since 6 September 2022; median result: 1.5 MP/s. Box plot of samples:
    [                  *    |-------------------------------------------------------------------------#####################!##
                       ^ This Result: 0.23
```



```
WebP Image Encode 1.2.4:
    pts/webp-1.2.0 [Encode Settings: Quality 100]
    Test 2 of 5
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      1 Minute
    Estimated Time To Completion: 3 Minutes [15:24 UTC]
        Started Run 1 @ 15:22:54
        Started Run 2 @ 15:23:12
        Started Run 3 @ 15:23:29

    Encode Settings: Quality 100:
        2.0178241129982
        2.0146058927222
        2.016298412165

    Average: 2.02 MP/s
    Deviation: 0.08%

    Comparison of 417 OpenBenchmarking.org samples since 6 September 2022; median result: 11.15 MP/s. Box plot of samples:
    [                      *    |----------------------------------------------------------------------#######################
                           ^ This Result: 2.02
```



```
AOBench:
    pts/aobench-1.0.1
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 13 Minutes [08:59 UTC]
        Started Run 1 @ 08:46:55
        Started Run 2 @ 08:51:09
        Started Run 3 @ 08:55:20

    Size: 2048 x 2048 - Total Time:
        250.101
        247.188
        246.745

    Average: 248.011 Seconds
    Deviation: 0.74%
```

```
CacheBench:
    pts/cachebench-1.1.2 [Test: Read / Modify / Write]
    Test 3 of 3
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 7 Minutes [10:42 UTC]
        Started Run 1 @ 10:36:30
        Started Run 2 @ 10:38:39
        Started Run 3 @ 10:40:48

    Test: Read / Modify / Write:
        6122.883894
        6122.267007
        6121.222451

    Average: 6122.124451 MB/s
    Minimum: 5593.44
    Maximum: 6380.75
    Deviation: 0.01%

    Comparison of 3,693 OpenBenchmarking.org samples since 26 February 2011 to 12 February 2022; median result: 27812 MB/s. Box plot of samples:
    [     |---------*-----------------------------------##########################!##############################---------------------------------
                    ^ This Result (4th Percentile): 6122
```

```
CacheBench:
    pts/cachebench-1.1.2 [Test: Read]
    Test 1 of 3
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      7 Minutes
    Estimated Time To Completion: 19 Minutes [10:41 UTC]
        Started Run 1 @ 10:22:22
        Started Run 2 @ 10:24:32
        Started Run 3 @ 10:26:41

    Test: Read:
        1266.950226
        1266.154618
        1266.842383

    Average: 1266.649076 MB/s
    Minimum: 1252.69
    Maximum: 1278.02
    Deviation: 0.03%

    MB/s > Higher Is Better
    cachebench intel up . 1266.65 |=====================================================================================
    Intel Atom x5-Z8350 . 1267.83 |=====================================================================================
```

```
CacheBench:
    pts/cachebench-1.1.2 [Test: Write]
    Test 2 of 3
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      7 Minutes
    Estimated Time To Completion: 13 Minutes [10:42 UTC]
        Started Run 1 @ 10:29:36
        Started Run 2 @ 10:31:45
        Started Run 3 @ 10:33:54

    Test: Write:
        6107.125752
        6107.925725
        6104.803843

    Average: 6106.618440 MB/s
    Minimum: 5364.38
    Maximum: 6376.43
    Deviation: 0.03%

    Comparison of 2,532 OpenBenchmarking.org samples since 26 February 2011 to 10 February 2022; median result: 19806 MB/s. Box plot of samples:
    [          |-------------*---------------------##########################################!##############################-------------------------------------|
                             ^ This Result (10th Percentile): 6107
```

```
Coremark 1.0:
    pts/coremark-1.0.1
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 2 Minutes [16:39 UTC]
        Started Run 1 @ 16:38:06
        Started Run 2 @ 16:38:31
        Started Run 3 @ 16:38:55

    CoreMark Size 666 - Iterations Per Second:
        23677.554754
        23756.816587
        23639.391823

    Average: 23691.254388 Iterations/Sec
    Deviation: 0.25%

    Iterations/Sec > Higher Is Better
    coremark intel up ... 23691.25 |==============================================================
=====
    Intel Atom x5-Z8350 . 23703.93 |==============================================================
=====

    Comparison of 4,578 OpenBenchmarking.org samples since 26 February 2011; median result: 320765
 Iterations/Sec. Box plot of samples:
    [-*####!############-----------*---------------------------------------------------*-*---|    *
    ]
      ^ AMD FX-8350: 114720          ^ AMD Ryzen Threadripper 3970X: 1436230
                                                      2 x Intel Xeon Platinum 8490H: 4342494 ^
                                                             AMD EPYC 9654: 3912868 ^
                                              Ampere Altra ARMv8 Neoverse-N1: 3832874 ^
```

```
Dolfyn 0.527:
    pts/dolfyn-1.0.3
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 7 Minutes [14:55 UTC]
        Started Run 1 @ 14:48:30
        Started Run 2 @ 14:50:53
        Started Run 3 @ 14:53:15

    Computational Fluid Dynamics:
        138.772
        138.215
        138.672

    Average: 138.553 Seconds
    Deviation: 0.21%

    Seconds < Lower Is Better
    dolfyn intel up ..... 138.55 |=========================
    Intel Atom x5-Z8350 . 137.66 |=========================
```

```
FFTE 7.0:
    pts/ffte-1.2.1
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 1 Minute [15:11 UTC]
        Started Run 1 @ 15:11:08
        Started Run 2 @ 15:11:16
        Started Run 3 @ 15:11:23

    N=256, 3D Complex FFT Routine:
        2837.4425133012
        2835.162081866
        2836.3501812151

    Average: 2836.32 MFLOPS
    Deviation: 0.04%

    MFLOPS > Higher Is Better
    Intel Atom x5-Z8350 . 2823.16 |==============================================================================================================
    ffte intel up ....... 2836.32 |==============================================================================================================

    Comparison of 1,019 OpenBenchmarking.org samples since 27 September 2020 to 7 February 2022; median result: 49970 MFLOPS. Box plot of samples:
    [ *--------------####################!#############################################################------------------------------------------------
      ^ This Result (2nd Percentile): 2836
```

```
FLAC Audio Encoding 1.4:
    pts/encode-flac-1.8.1
    Test 1 of 1
    Estimated Trial Run Count:    5
    Estimated Time To Completion: 19 Minutes [09:29 UTC]
        Started Run 1 @ 09:10:44
        Started Run 2 @ 09:14:28
        Started Run 3 @ 09:18:12
        Started Run 4 @ 09:21:56
        Started Run 5 @ 09:25:40

    WAV To FLAC:
        219.822
        219.802
        220.105
        219.469
        219.695

    Average: 219.779 Seconds
    Deviation: 0.10%

    Seconds < Lower Is Better
    Intel Atom x5-Z8350 . 219.78 |=============================================
======================
    Intel Atom x5-Z8350 . 219.25 |=============================================
======================
    Intel Atom x5-Z8350 . 219.21 |=============================================
======================
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Enhanced]
    Test 4 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 13 Minutes [16:09 UTC]
        Started Run 1 @ 15:57:37
        Started Run 2 @ 15:58:44
        Started Run 3 @ 15:59:51

    Operation: Enhanced:
        6
        6
        6

    Average: 6 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 734 OpenBenchmarking.org samples since 22 August 2022; median result: 300 Iterations
Per Minute. Box plot of samples:
    [ |---------###############!###############*#*###*##*###########--------------------------*-------
---------------------*---*-----------*-------*------------------*------------|           *       ]
                       AMD Ryzen Threadripper 3960X: 569 ^   AMD Ryzen Threadripper 3990X: 966 ^   2 x
AMD EPYC 9374F: 1283 ^                 2 x AMD EPYC 9554: 1748 ^ 2 x AMD EPYC 9654: 2029 ^
                   AMD Ryzen Threadripper 2990WX: 535 ^
                         AMD EPYC 9654: 1534 ^
                         AMD Ryzen 9 7950X3D: 488 ^
            2 x AMD EPYC 7773X: 1453 ^
                         AMD Ryzen 9 7900X: 474 ^
 2 x AMD EPYC 7713: 1324 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: HWB Color Space]
    Test 7 of 7
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 4 Minutes [16:11 UTC]
        Started Run 1 @ 16:08:01
        Started Run 2 @ 16:09:06
        Started Run 3 @ 16:10:11

    Operation: HWB Color Space:
        64
        64
        64

    Average: 64 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 544 OpenBenchmarking.org samples since 22 August 2022; median result: 941 Iterations
Per Minute. Box plot of samples:
    [      *  |-----------------------------*-----------*--------*---####*#####################*#####
#!#######*####*#######*#####*################-----------*-----------*-------*--*-*-----------|  ]
           ^ This Result (1st Percentile): 64
                               Intel Core i7-1065G7: 513 ^          Intel Core i9-10980XE: 878 ^
 Intel Core i9-10900K: 1203 ^ Intel Core i5-13600K: 1476 ^ AMD Ryzen 9 7950X: 1724 ^
                        AMD Athlon 3000G: 396 ^                                                     AM
D Ryzen 7 5700G: 1145 ^                                 AMD Ryzen 9 7900X: 1705 ^
                                                   AMD Ryzen 5 4500U: 674 ^     AMD Ryzen Threadrippe
r 3960X: 1066 ^                                    AMD Ryzen 9 7950X3D: 1672 ^
                                           AMD Ryzen 7 4800U: 596 ^      AMD Ryzen Threadripper 2990
WX: 1025 ^                                 Intel Core i9-13900K: 1602 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Noise-Gaussian]
    Test 6 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 7 Minutes [16:10 UTC]
        Started Run 1 @ 16:04:30
        Started Run 2 @ 16:05:38
        Started Run 3 @ 16:06:46

    Operation: Noise-Gaussian:
        12
        12
        12

    Average: 12 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 542 OpenBenchmarking.org samples since 22 August 2022; median result: 229 Iterations
Per Minute. Box plot of samples:
    [ *  |*---------------#################*##*#!##*########*###########################*-------------
-----------------------*--*-*---*-----------------------------------|                          *   ]
      ^ This Result (1st Percentile): 12
          ^ ARMv8 Cortex-A72: 30    Intel Core i5-13400: 291 ^ Intel Core i9-12900K: 435 ^          AM
D Ryzen Threadripper 3970X: 666 ^                                         2 x AMD EPYC 9654: 990 ^
                             AMD Ryzen 5 5600X: 247 ^
     AMD Ryzen 9 7950X: 649 ^
                      Intel Core i5-12400: 221 ^                                                AMD Ryze
n Threadripper 3990X: 637 ^
                   Intel Core i7-5960X: 206 ^                                           AMD Ryzen Thread
ripper PRO 5965WX: 624 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Resizing]
    Test 5 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 10 Minutes [16:10 UTC]
        Started Run 1 @ 16:01:05
        Started Run 2 @ 16:02:10
        Started Run 3 @ 16:03:16

    Operation: Resizing:
        41
        41
        41

    Average: 41 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 547 OpenBenchmarking.org samples since 22 August 2022; median result: 852 Iterations
Per Minute. Box plot of samples:
    [  *------*-----------------##########*####*####*#*#####!######################*#################
*#*###--*--*-----------------------------------*---*---*-------------*----------*-----|            ]
       ^ This Result (1st Percentile): 41
              ^ 2 x AMD EPYC 7601: 151   Apple M1: 755 ^  Intel Core i9-10900K: 1212 ^    AMD Ryzen 7 7
700X: 1637 ^         AMD Ryzen 9 7950X3D: 2179 ^        AMD Ryzen 9 7950X: 2662 ^
                            Intel Core i7-5960X: 731 ^                           Intel Core i5-13600
K: 1585 ^                         AMD Ryzen Threadripper 3970X: 2495 ^
                       Intel Core i7-8700K: 660 ^                               AMD Ryzen 9 3900X: 149
8 ^            AMD Ryzen Threadripper PRO 5965WX: 2298 ^
                  Intel Core i7-9850H: 575 ^                                Intel Core i9-10980XE: 1477
^                          AMD Ryzen 9 7900X: 2229 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Rotate]
    Test 2 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 19 Minutes [16:08 UTC]
        Started Run 1 @ 15:49:47
        Started Run 2 @ 15:50:51
        Started Run 3 @ 15:51:56
        Started Run 4 @ 15:53:00 *

    Operation: Rotate:
        84
        83
        80
        81

    Average: 82 Iterations Per Minute
    Deviation: 2.23%
    Samples: 4

    Comparison of 481 OpenBenchmarking.org samples since 22 August 2022; median result: 711 Iterations
Per Minute. Box plot of samples:
    [         * |-------------------------*----------*------*---###################!#*##########*##
*###*-------*-----------------------*-------*----------------*---------*-------|   *              ]
          ^ This Result: 82
                                        AMD Ryzen 7 4800U: 511 ^  AMD Ryzen 9 3900X: 730 ^  AMD Ryzen 7
 5800X: 951 ^ AMD Ryzen 5 5600: 1173 ^                  Intel Core i9-13900K: 1574 ^
                                 2 x AMD EPYC 7601: 446 ^                     Intel Core i9-10900K:
883 ^                                                  Apple M1: 1469 ^
                 Intel Xeon E5-2609 v4: 340 ^                                  AMD Ryzen 9 5900HX: 847
^                                  Intel Core i9-12900K: 1386 ^
                                                          AMD Ryzen Threadripper PRO 5965WX: 827 ^
                  Intel Core i5-12600K: 1238 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Sharpen]
    Test 3 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 16 Minutes [16:09 UTC]
        Started Run 1 @ 15:54:13
        Started Run 2 @ 15:55:19
        Started Run 3 @ 15:56:24

    Operation: Sharpen:
        5
        5
        5

    Average: 5 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 635 OpenBenchmarking.org samples since 22 August 2022; median result: 197 Iterations
Per Minute. Box plot of samples:
    [ |---------###*#############!############*#######*######*##*############--------------------------
*-----------------------------*-----*---------------------------*-------*---------|              *   ]
                    ^ Apple M2: 104    Intel Core i9-13900K: 407 ^    AMD Ryzen Threadripper 3990X: 669
^     2 x AMD EPYC 7773X: 875 ^                                 2 x AMD EPYC 9654: 1314 ^
                          AMD Ryzen Threadripper 2990WX: 387 ^
                                    2 x Intel Xeon Platinum 8490H: 1159 ^
               AMD Ryzen Threadripper PRO 5965WX: 341 ^
                                        2 x AMD EPYC 9554: 1107 ^
                    Intel Core i9-12900K: 284 ^
                 AMD EPYC 9654: 912 ^
```

```
GraphicsMagick 1.3.38:
    pts/graphics-magick-2.1.0 [Operation: Swirl]
    Test 1 of 7
    Estimated Trial Run Count:    3
    Estimated Test Run-Time:      4 Minutes
    Estimated Time To Completion: 22 Minutes [16:07 UTC]
        Started Run 1 @ 15:46:24
        Started Run 2 @ 15:47:30
        Started Run 3 @ 15:48:34

    Operation: Swirl:
        20
        20
        20

    Average: 20 Iterations Per Minute
    Deviation: 0.00%

    Comparison of 461 OpenBenchmarking.org samples since 22 August 2022; median result: 354 Iterations
Per Minute. Box plot of samples:
    [ |----------############!#######################*----------------*--------*--------*-*-----------
---------------------------*-------------------------*--*--*-------------------------|    *    *]
                          Intel Core i9-10980XE: 674 ^ AMD Ryzen Threadripper 3960X: 1186 ^     AMD Ryz
en Threadripper 3990X: 1747 ^  2 x AMD EPYC 7713: 2110 ^                 2 x AMD EPYC 9654: 2706 ^
                                                                 AMD Ryzen 9 7950X3D: 1151 ^
                                                            2 x Intel Xeon Platinum 8490H: 2645 ^
                                              AMD Ryzen Threadripper 2990WX: 1036 ^
                          2 x Intel Xeon Platinum 8380: 2200 ^
                                            Intel Core i9-12900K: 907 ^
                          Intel Xeon Platinum 8490H: 2161 ^
```

```
Himeno Benchmark 3.0:
    pts/himeno-1.3.0
    Test 1 of 1
    Estimated Trial Run Count:    3
    Estimated Time To Completion: 4 Minutes [09:40 UTC]
        Started Run 1 @ 09:37:31
        Started Run 2 @ 09:38:35
        Started Run 3 @ 09:39:39

    Poisson Pressure Solver:
        595.416596
        594.584029
        594.261464

    Average: 594.754030 MFLOPS
    Deviation: 0.10%

    MFLOPS > Higher Is Better
    Intel Atom x5-Z8350 ....... 594.64 |==================================================
    Himeno Benchmark intel up . 594.75 |==================================================
    Intel Atom x5-Z8350 ....... 595.17 |==================================================

    Comparison of 2,351 OpenBenchmarking.org samples since 26 February 2011 to 13 February 2022; median result: 3646 MFLOPS. Box plot of samples:
    [   |-------*---------------------------------------##################!##############------------------------|
                ^ This Result (5th Percentile): 595
```

```
t-test1 2017-01-13
Threads: 1
Seconds < Lower Is Better
Intel Atom x5-Z8350 . 135.19 |===================


t-test1 2017-01-13
Threads: 2
Seconds < Lower Is Better
Intel Atom x5-Z8350 . 44.58 |====================
```

```
SQLite Speedtest 3.30
Timed Time - Size 1,000
Seconds < Lower Is Better
Intel Atom x5-Z8350 . 591.21 |=============
```

```
PHPBench 0.8.1
PHP Benchmark Suite
Score > Higher Is Better
Intel Atom x5-Z8350 . 126838 |==============
```

loh@loh-UP-CHT01: ~

```
Performance version: 1.11.0
Report on Linux-6.5.0-27-generic-x86_64-with-glibc2.35
Number of logical CPUs: 4
Start date: 2024-04-23 20:38:38.822898
End date: 2024-04-23 20:59:02.305550

### 2to3 ###
Mean +- std dev: 2.55 sec +- 0.01 sec

### chaos ###
Mean +- std dev: 792 ms +- 3 ms

### crypto_pyaes ###
Mean +- std dev: 773 ms +- 7 ms

### float ###
Mean +- std dev: 705 ms +- 6 ms

### go ###
Mean +- std dev: 1.55 sec +- 0.01 sec

### json_loads ###
Mean +- std dev: 196 us +- 2 us

### nbody ###
Mean +- std dev: 812 ms +- 6 ms

### pathlib ###
Mean +- std dev: 253 ms +- 1 ms

### python_startup ###
Mean +- std dev: 63.3 ms +- 1.5 ms

### raytrace ###
Mean +- std dev: 3.52 sec +- 0.01 sec

### regex_compile ###
Mean +- std dev: 1.22 sec +- 0.01 sec
```

```
NCNN 20230517
Target: Vulkan GPU - Model: googlenet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 164.88 |===

NCNN 20230517
Target: Vulkan GPU - Model: vgg16
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 676.31 |===

NCNN 20230517
Target: Vulkan GPU - Model: resnet18
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 134.29 |===

NCNN 20230517
Target: Vulkan GPU - Model: alexnet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 105.87 |===

NCNN 20230517
Target: Vulkan GPU - Model: resnet50
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 361.24 |===

NCNN 20230517
Target: Vulkan GPU - Model: yolov4-tiny
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 263.33 |===

NCNN 20230517
Target: Vulkan GPU - Model: squeezenet_ssd
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 104.03 |===

NCNN 20230517
Target: Vulkan GPU - Model: regnety_400m
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 82.77 |====
```

```
NCNN 20230517
Target: Vulkan GPU - Model: vision_transformer
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 3579.72 |=====

NCNN 20230517
Target: Vulkan GPU - Model: FastestDet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 41.62 |=======
```

```
OpenCV 4.7
Test: Core
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 570215 |==

OpenCV 4.7
Test: Video
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 75238 |===
```

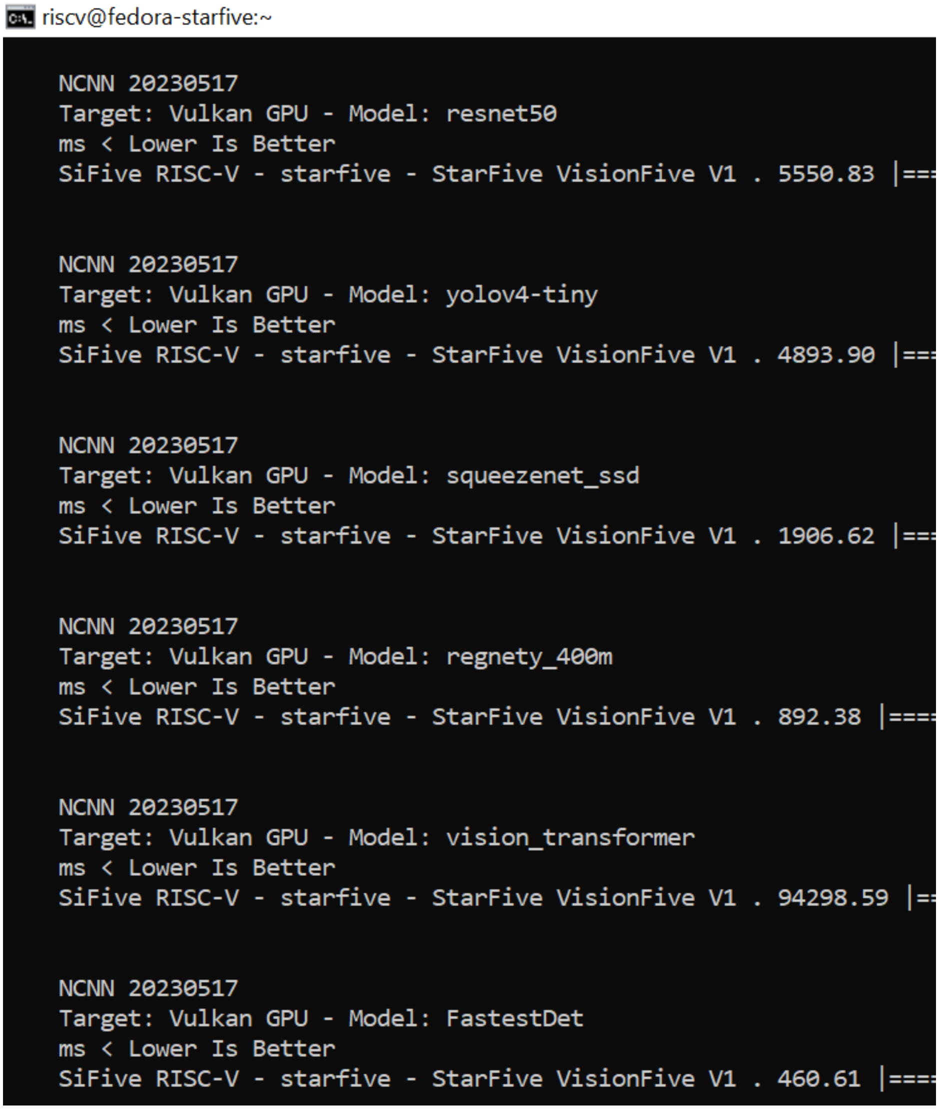
```
OpenCV 4.7
Test: Stitching
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 2163028 |==

OpenCV 4.7
Test: Features 2D
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 728507 |===

OpenCV 4.7
Test: Image Processing
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 889556 |===

OpenCV 4.7
Test: Object Detection
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel HD CHV 2GB - AAEON . 224212 |===
```

```
NCNN 20230517
Target: CPU - Model: mobilenet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 198.68 |===

NCNN 20230517
Target: CPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 56.49 |====

NCNN 20230517
Target: CPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 48.88 |====

NCNN 20230517
Target: CPU - Model: shufflenet-v2
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 35.14 |====

NCNN 20230517
Target: CPU - Model: mnasnet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 54.78 |====

NCNN 20230517
Target: CPU - Model: efficientnet-b0
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 101.29 |===

NCNN 20230517
Target: CPU - Model: blazeface
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 9.96 |=====

NCNN 20230517
Target: CPU - Model: googlenet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 164.70 |===
```

```
NCNN 20230517
Target: CPU - Model: vgg16
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 676.93 |===

NCNN 20230517
Target: CPU - Model: resnet18
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 135.53 |===

NCNN 20230517
Target: CPU - Model: alexnet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 105.53 |===

NCNN 20230517
Target: CPU - Model: resnet50
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 360.98 |===

NCNN 20230517
Target: CPU - Model: yolov4-tiny
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 264.43 |===

NCNN 20230517
Target: CPU - Model: squeezenet_ssd
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 104.10 |===

NCNN 20230517
Target: CPU - Model: regnety_400m
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 82.87 |====

NCNN 20230517
Target: CPU - Model: vision_transformer
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 3592.31 |==
```

```
NCNN 20230517
Target: CPU - Model: FastestDet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 41.44 |====

NCNN 20230517
Target: Vulkan GPU - Model: mobilenet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 198.75 |===

NCNN 20230517
Target: Vulkan GPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 56.48 |====

NCNN 20230517
Target: Vulkan GPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 48.94 |====

NCNN 20230517
Target: Vulkan GPU - Model: shufflenet-v2
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 35.26 |====

NCNN 20230517
Target: Vulkan GPU - Model: mnasnet
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 54.75 |====

NCNN 20230517
Target: Vulkan GPU - Model: efficientnet-b0
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 101.30 |===

NCNN 20230517
Target: Vulkan GPU - Model: blazeface
ms < Lower Is Better
Intel Atom x5-Z8350 - Intel Atom/Celeron/Pentium . 9.82 |=====
```

Appendix B – Benchmark Results of StarFive VisionFive V1



```
riscv@fedora-starfive:~

NCNN 20230517
Target: Vulkan GPU - Model: resnet50
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 5550.83 |==

NCNN 20230517
Target: Vulkan GPU - Model: yolov4-tiny
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 4893.90 |==

NCNN 20230517
Target: Vulkan GPU - Model: squeezenet_ssd
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 1906.62 |==

NCNN 20230517
Target: Vulkan GPU - Model: regnety_400m
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 892.38 |===

NCNN 20230517
Target: Vulkan GPU - Model: vision_transformer
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 94298.59 |=

NCNN 20230517
Target: Vulkan GPU - Model: FastestDet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 460.61 |===
```

riscv@fedora-starfive:~

```
NCNN 20230517
Target: Vulkan GPU - Model: mnasnet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 739.72 |=======


NCNN 20230517
Target: Vulkan GPU - Model: efficientnet-b0
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 1364.54 |======


NCNN 20230517
Target: Vulkan GPU - Model: blazeface
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 106.02 |=======


NCNN 20230517
Target: Vulkan GPU - Model: googlenet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2483.86 |======


NCNN 20230517
Target: Vulkan GPU - Model: vgg16
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 18040.57 |=====


NCNN 20230517
Target: Vulkan GPU - Model: resnet18
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2225.94 |======


NCNN 20230517
Target: Vulkan GPU - Model: alexnet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2238.31 |======
```

riscv@fedora-starfive:~

```
NCNN 20230517
Target: CPU - Model: regnety_400m
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 890.52 |====


NCNN 20230517
Target: CPU - Model: vision_transformer
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 94334.67 |==


NCNN 20230517
Target: CPU - Model: FastestDet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 460.59 |====


NCNN 20230517
Target: Vulkan GPU - Model: mobilenet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2697.77 |===


NCNN 20230517
Target: Vulkan GPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 736.47 |====


NCNN 20230517
Target: Vulkan GPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 683.10 |====


NCNN 20230517
Target: Vulkan GPU - Model: shufflenet-v2
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 308.66 |====
```

riscv@fedora-starfive:~

```
NCNN 20230517
Target: CPU - Model: googlenet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2471.76 |====


NCNN 20230517
Target: CPU - Model: vgg16
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 18046.56 |===


NCNN 20230517
Target: CPU - Model: resnet18
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2229.58 |====


NCNN 20230517
Target: CPU - Model: alexnet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2237.13 |====


NCNN 20230517
Target: CPU - Model: resnet50
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 5570.38 |====


NCNN 20230517
Target: CPU - Model: yolov4-tiny
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 4911.14 |====


NCNN 20230517
Target: CPU - Model: squeezenet_ssd
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 1901.89 |====
```

riscv@fedora-starfive:~

```
NCNN 20230517
Target: CPU - Model: mobilenet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2696.50 |====


NCNN 20230517
Target: CPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 742.83 |=====


NCNN 20230517
Target: CPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 682.55 |=====


NCNN 20230517
Target: CPU - Model: shufflenet-v2
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 308.58 |=====


NCNN 20230517
Target: CPU - Model: mnasnet
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 739.58 |=====


NCNN 20230517
Target: CPU - Model: efficientnet-b0
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 1366.91 |====


NCNN 20230517
Target: CPU - Model: blazeface
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 105.58 |=====
```

```
OpenCV 4.7
Test: Core
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 4451129 |=====

OpenCV 4.7
Test: Video
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 797741 |======

OpenCV 4.7
Test: Stitching
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 6261587 |=====

OpenCV 4.7
Test: Features 2D
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 3772199 |=====

OpenCV 4.7
Test: Image Processing
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 11800160 |====

OpenCV 4.7
Test: Object Detection
ms < Lower Is Better
SiFive RISC-V - starfive - StarFive VisionFive V1 . 2400352 |=====
```

```
SQLite Speedtest 3.30
Timed Time - Size 1,000
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 1877.96 |==================================================================================

PHPBench 0.8.1
PHP Benchmark Suite
Score > Higher Is Better
SC32G - SiFive RISC-V . 38488 |====================================================================================
```

Select riscv@fedora-starfive:~

```
FLAC Audio Encoding 1.4
WAV To FLAC
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 603.99 |=================================================================================

Perl Benchmarks
Test: Pod2html
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 2.59592653 |=============================================================================

Perl Benchmarks
Test: Interpreter
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 0.01391912 |=============================================================================
```

Select riscv@fedora-starfive:~

```
GraphicsMagick 1.3.38
Operation: Sharpen
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 4 |==========================================================================================

GraphicsMagick 1.3.38
Operation: Enhanced
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 2 |==========================================================================================

GraphicsMagick 1.3.38
Operation: Resizing
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 11 |=========================================================================================

GraphicsMagick 1.3.38
Operation: Noise-Gaussian
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 4 |==========================================================================================

GraphicsMagick 1.3.38
Operation: HWB Color Space
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 12 |=========================================================================================

Coremark 1.0
CoreMark Size 666 - Iterations Per Second
Iterations/Sec > Higher Is Better
SC32G - SiFive RISC-V . 6758.17 |====================================================================================

Himeno Benchmark 3.0
Poisson Pressure Solver
MFLOPS > Higher Is Better
SC32G - SiFive RISC-V . 12.46 |======================================================================================

AOBench
Size: 2048 x 2048 - Total Time
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 1254.55 |====================================================================================
```

Select riscv@fedora-starfive:~

```
CacheBench
Test: Read / Modify / Write
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 1323.27 |====================================================================================

WebP Image Encode 1.2.4
Encode Settings: Default
MP/s > Higher Is Better
SC32G - SiFive RISC-V . 0.74 |=======================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100
MP/s > Higher Is Better
SC32G - SiFive RISC-V . 0.56 |=======================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless
MP/s > Higher Is Better
SC32G - SiFive RISC-V . 0.09 |=======================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Highest Compression
MP/s > Higher Is Better
SC32G - SiFive RISC-V . 0.34 |=======================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless, Highest Compression
MP/s > Higher Is Better
SC32G - SiFive RISC-V . 0.03 |=======================================================================================

GraphicsMagick 1.3.38
Operation: Swirl
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 4 |==========================================================================================

GraphicsMagick 1.3.38
Operation: Rotate
Iterations Per Minute > Higher Is Better
SC32G - SiFive RISC-V . 22 |=========================================================================================
```

Select riscv@fedora-starfive:~

```
Tinymembench 2018-05-28
Standard Memcpy
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 138.1 |==========================================================================

Tinymembench 2018-05-28
Standard Memset
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 261.1 |==========================================================================

t-test1 2017-01-13
Threads: 1
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 389.97 |=========================================================================

t-test1 2017-01-13
Threads: 2
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 121.45 |=========================================================================

Dolfyn 0.527
Computational Fluid Dynamics
Seconds < Lower Is Better
SC32G - SiFive RISC-V . 700.85 |=========================================================================

FFTE 7.0
N=256, 3D Complex FFT Routine
MFLOPS > Higher Is Better
SC32G - SiFive RISC-V . 452.94 |=========================================================================

CacheBench
Test: Read
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 1060.04 |========================================================================

CacheBench
Test: Write
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 3406.84 |========================================================================
```

```
RAMspeed SMP 3.5.0
Type: Copy - Benchmark: Integer
MB/s > Higher Is Better
SC32G - SiFive RISC-V . 479.01 |=========================================================================
```

riscv@fedora-starfive:~

```
PostMark 1.51
Disk Transaction Performance
TPS > Higher Is Better
SC32G - SiFive RISC-V . 225 |============================================================================
```

riscv@fedora-starfive:~

```
Performance version: 1.11.0
Report on Linux-5.15.10+-riscv64-with-glibc2.36
Number of logical CPUs: 2
Start date: 2024-04-23 16:05:05.106274
End date: 2024-04-23 17:36:37.714252

### 2to3 ###
Mean +- std dev: 10.2 sec +- 0.2 sec

### chaos ###
Mean +- std dev: 3.69 sec +- 0.16 sec

### crypto_pyaes ###
Mean +- std dev: 3.63 sec +- 0.11 sec

### float ###
Mean +- std dev: 4.53 sec +- 0.16 sec

### go ###
Mean +- std dev: 9.02 sec +- 0.42 sec

### json_loads ###
Mean +- std dev: 554 us +- 23 us

### nbody ###
Mean +- std dev: 4.30 sec +- 0.28 sec

### pathlib ###
Mean +- std dev: 565 ms +- 17 ms

### python_startup ###
Mean +- std dev: 156 ms +- 7 ms

### raytrace ###
Mean +- std dev: 18.2 sec +- 0.7 sec

### regex_compile ###
Mean +- std dev: 5.40 sec +- 0.24 sec
```

Appendix C – Benchmark Results of StarFive VisionFive V2

user@starfive: ~

```
NCNN 20230517
Target: CPU - Model: mobilenet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 617.90 |==============

NCNN 20230517
Target: CPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 174.22 |==============

NCNN 20230517
Target: CPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 166.13 |==============

NCNN 20230517
Target: CPU - Model: shufflenet-v2
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 85.16 |===============

NCNN 20230517
Target: CPU - Model: mnasnet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 189.98 |==============

NCNN 20230517
Target: CPU - Model: efficientnet-b0
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 341.92 |==============

NCNN 20230517
Target: CPU - Model: blazeface
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 25.32 |===============
```

user@starfive: ~

```
NCNN 20230517
Target: CPU - Model: googlenet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 505.86 |======


NCNN 20230517
Target: CPU - Model: vgg16
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 2341.12 |=====


NCNN 20230517
Target: CPU - Model: resnet18
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 397.49 |======


NCNN 20230517
Target: CPU - Model: alexnet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 292.11 |======


NCNN 20230517
Target: CPU - Model: resnet50
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 1189.86 |=====


NCNN 20230517
Target: CPU - Model: yolov4-tiny
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 793.36 |======


NCNN 20230517
Target: CPU - Model: squeezenet_ssd
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 347.41 |======
```

user@starfive: ~

```
NCNN 20230517
Target: CPU - Model: regnety_400m
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 227.17 |====


NCNN 20230517
Target: CPU - Model: vision_transformer
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 14223.13 |==


NCNN 20230517
Target: CPU - Model: FastestDet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 111.04 |====


NCNN 20230517
Target: Vulkan GPU - Model: mobilenet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 617.98 |====


NCNN 20230517
Target: Vulkan GPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 174.22 |====


NCNN 20230517
Target: Vulkan GPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 165.79 |====


NCNN 20230517
Target: Vulkan GPU - Model: shufflenet-v2
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 83.96 |=====
```

user@starfive: ~

```
NCNN 20230517
Target: Vulkan GPU - Model: mnasnet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 190.16 |=======


NCNN 20230517
Target: Vulkan GPU - Model: efficientnet-b0
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 341.81 |=======


NCNN 20230517
Target: Vulkan GPU - Model: blazeface
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 25.36 |========


NCNN 20230517
Target: Vulkan GPU - Model: googlenet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 505.45 |=======


NCNN 20230517
Target: Vulkan GPU - Model: vgg16
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 2336.29 |======


NCNN 20230517
Target: Vulkan GPU - Model: resnet18
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 415.10 |=======


NCNN 20230517
Target: Vulkan GPU - Model: alexnet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 290.72 |=======
```

user@starfive: ~

```
NCNN 20230517
Target: Vulkan GPU - Model: resnet50
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 1189.83 |======

NCNN 20230517
Target: Vulkan GPU - Model: yolov4-tiny
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 796.46 |=======

NCNN 20230517
Target: Vulkan GPU - Model: squeezenet_ssd
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 347.49 |=======

NCNN 20230517
Target: Vulkan GPU - Model: regnety_400m
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 227.63 |=======

NCNN 20230517
Target: Vulkan GPU - Model: vision_transformer
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 14234.59 |=====

NCNN 20230517
Target: Vulkan GPU - Model: FastestDet
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 107.60 |=======
```

```
OpenCV 4.7
Test: Core
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 1386586 |==

OpenCV 4.7
Test: Video
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 209956 |===
```

```
OpenCV 4.7
Test: Stitching
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 2317127 |===

OpenCV 4.7
Test: Features 2D
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 1070653 |===

OpenCV 4.7
Test: Image Processing
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 1979376 |===

OpenCV 4.7
Test: Object Detection
ms < Lower Is Better
SiFive RISC-V - softpipe 8GB - StarFive VisionFive V2 . 814200 |====
```

user@starfive: ~

```
PostMark 1.51
Disk Transaction Performance
TPS > Higher Is Better
SD32G - SiFive RISC-V . 523 |==========================================================================================
```

```
RAMspeed SMP 3.5.0
Type: Copy - Benchmark: Integer
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 2158.25 |=====================================================================================
```

user@starfive: ~

```
Tinymembench 2018-05-28
Standard Memcpy
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 945.7 |========================================================================================

Tinymembench 2018-05-28
Standard Memset
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 833.4 |========================================================================================

t-test1 2017-01-13
Threads: 1
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 151.23 |=======================================================================================

t-test1 2017-01-13
Threads: 2
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 47.19 |========================================================================================

Dolfyn 0.527
Computational Fluid Dynamics
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 303.78 |=======================================================================================

FFTE 7.0
N=256, 3D Complex FFT Routine
MFLOPS > Higher Is Better
SD32G - SiFive RISC-V . 1424.32 |======================================================================================

CacheBench
Test: Read
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 1588.16 |======================================================================================

CacheBench
Test: Write
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 5238.55 |======================================================================================
```

user@starfive: ~

```
CacheBench
Test: Read / Modify / Write
MB/s > Higher Is Better
SD32G - SiFive RISC-V . 2009.14 |==================================================================================

WebP Image Encode 1.2.4
Encode Settings: Default
MP/s > Higher Is Better
SD32G - SiFive RISC-V . 0.95 |=====================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100
MP/s > Higher Is Better
SD32G - SiFive RISC-V . 0.75 |=====================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless
MP/s > Higher Is Better
SD32G - SiFive RISC-V . 0.17 |=====================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Highest Compression
MP/s > Higher Is Better
SD32G - SiFive RISC-V . 0.45 |=====================================================================================

WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless, Highest Compression
MP/s > Higher Is Better
SD32G - SiFive RISC-V . 0.06 |=====================================================================================

GraphicsMagick 1.3.38
Operation: Swirl
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 12 |=======================================================================================

GraphicsMagick 1.3.38
Operation: Rotate
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 54 |=======================================================================================
```

user@starfive: ~

```
GraphicsMagick 1.3.38
Operation: Sharpen
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 11 |=======================================================================================

GraphicsMagick 1.3.38
Operation: Enhanced
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 4 |========================================================================================

GraphicsMagick 1.3.38
Operation: Resizing
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 35 |=======================================================================================

GraphicsMagick 1.3.38
Operation: Noise-Gaussian
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 11 |=======================================================================================

GraphicsMagick 1.3.38
Operation: HWB Color Space
Iterations Per Minute > Higher Is Better
SD32G - SiFive RISC-V . 39 |=======================================================================================

Coremark 1.0
CoreMark Size 666 - Iterations Per Second
Iterations/Sec > Higher Is Better
SD32G - SiFive RISC-V . 21669.90 |=================================================================================

AOBench
Size: 2048 x 2048 - Total Time
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 316.09 |===================================================================================

FLAC Audio Encoding 1.4
WAV To FLAC
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 341.53 |===================================================================================
```

user@starfive: ~

```
Perl Benchmarks
Test: Pod2html
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 1.07501350 |================================================================================

Perl Benchmarks
Test: Interpreter
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 0.00774396 |================================================================================
```

user@starfive: ~

```
SQLite Speedtest 3.30
Timed Time - Size 1,000
Seconds < Lower Is Better
SD32G - SiFive RISC-V . 697.33 |================================================================================
```

```
PHPBench 0.8.1
PHP Benchmark Suite
Score > Higher Is Better
SD32G - SiFive RISC-V . 63300 |=====================================================
```

user@starfive: ~

```
Performance version: 1.11.0
Report on Linux-5.15.0-starfive-riscv64-with-glibc2.36
Number of logical CPUs: 4
Start date: 2024-04-23 09:51:00.168965
End date: 2024-04-23 10:24:53.514837

### 2to3 ###
Mean +- std dev: 4.00 sec +- 0.02 sec

### chaos ###
Mean +- std dev: 1.34 sec +- 0.01 sec

### crypto_pyaes ###
Mean +- std dev: 1.33 sec +- 0.01 sec

### float ###
Mean +- std dev: 1.47 sec +- 0.02 sec

### go ###
Mean +- std dev: 2.90 sec +- 0.03 sec

### json_loads ###
Mean +- std dev: 292 us +- 5 us

### nbody ###
Mean +- std dev: 1.38 sec +- 0.01 sec

### pathlib ###
Mean +- std dev: 259 ms +- 3 ms

### python_startup ###
Mean +- std dev: 65.7 ms +- 3.5 ms

### raytrace ###
Mean +- std dev: 6.60 sec +- 0.08 sec

### regex_compile ###
Mean +- std dev: 2.13 sec +- 0.02 sec
```

Appendix D – Benchmark Results of Raspberry Pi 4B

pi@raspberrypi: ~

```
NCNN 20230517
Target: CPU - Model: mobilenet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 739.49 |=======


NCNN 20230517
Target: CPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 191.35 |=======


NCNN 20230517
Target: CPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 164.33 |=======


NCNN 20230517
Target: CPU - Model: shufflenet-v2
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 99.61 |========


NCNN 20230517
Target: CPU - Model: mnasnet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 189.49 |=======


NCNN 20230517
Target: CPU - Model: efficientnet-b0
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 320.64 |=======


NCNN 20230517
Target: CPU - Model: blazeface
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 31.62 |========
```

pi@raspberrypi: ~

```
NCNN 20230517
Target: CPU - Model: googlenet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 586.94 |====


NCNN 20230517
Target: CPU - Model: vgg16
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 2424.98 |===


NCNN 20230517
Target: CPU - Model: resnet18
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 469.61 |====


NCNN 20230517
Target: CPU - Model: alexnet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 320.31 |====


NCNN 20230517
Target: CPU - Model: resnet50
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 1461.65 |===


NCNN 20230517
Target: CPU - Model: yolov4-tiny
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 870.83 |====


NCNN 20230517
Target: CPU - Model: squeezenet_ssd
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 377.06 |====
```

pi@raspberrypi: ~

```
NCNN 20230517
Target: CPU - Model: regnety_400m
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 279.45 |====


NCNN 20230517
Target: CPU - Model: vision_transformer
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 6105.88 |===


NCNN 20230517
Target: CPU - Model: FastestDet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 112.11 |====


NCNN 20230517
Target: Vulkan GPU - Model: mobilenet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 713.25 |====


NCNN 20230517
Target: Vulkan GPU-v2-v2 - Model: mobilenet-v2
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 207.03 |====


NCNN 20230517
Target: Vulkan GPU-v3-v3 - Model: mobilenet-v3
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 175.74 |====


NCNN 20230517
Target: Vulkan GPU - Model: shufflenet-v2
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 105.47 |====
```

pi@raspberrypi: ~

```
NCNN 20230517
Target: Vulkan GPU - Model: mnasnet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 200.79 |====


NCNN 20230517
Target: Vulkan GPU - Model: efficientnet-b0
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 331.99 |====


NCNN 20230517
Target: Vulkan GPU - Model: blazeface
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 32.60 |=====


NCNN 20230517
Target: Vulkan GPU - Model: googlenet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 598.85 |====


NCNN 20230517
Target: Vulkan GPU - Model: vgg16
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 2423.40 |===


NCNN 20230517
Target: Vulkan GPU - Model: resnet18
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 474.60 |====


NCNN 20230517
Target: Vulkan GPU - Model: alexnet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 322.83 |====
```

pi@raspberrypi: ~

```
NCNN 20230517
Target: Vulkan GPU - Model: resnet50
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 1467.91 |==


NCNN 20230517
Target: Vulkan GPU - Model: yolov4-tiny
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 852.24 |===


NCNN 20230517
Target: Vulkan GPU - Model: squeezenet_ssd
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 369.39 |===


NCNN 20230517
Target: Vulkan GPU - Model: regnety_400m
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 286.33 |===


NCNN 20230517
Target: Vulkan GPU - Model: vision_transformer
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 5779.54 |==


NCNN 20230517
Target: Vulkan GPU - Model: FastestDet
ms < Lower Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 111.76 |===
```

```
Tinymembench 2018-05-28
Standard Memcpy
MB/s > Higher Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 2465.4 |==


Tinymembench 2018-05-28
Standard Memset
MB/s > Higher Is Better
ARMv7 Cortex-A72 - vc4drmfb - BCM2711 Raspberry Pi 4 . 3004.2 |==
```

```
WebP Image Encode 1.2.4
Encode Settings: Default
MP/s > Higher Is Better
ARMv7 Cortex-A72 . 3.23 |====================================


WebP Image Encode 1.2.4
Encode Settings: Quality 100
MP/s > Higher Is Better
ARMv7 Cortex-A72 . 2.38 |====================================


WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless
MP/s > Higher Is Better
ARMv7 Cortex-A72 . 0.28 |====================================


WebP Image Encode 1.2.4
Encode Settings: Quality 100, Highest Compression
MP/s > Higher Is Better
ARMv7 Cortex-A72 . 0.85 |====================================


WebP Image Encode 1.2.4
Encode Settings: Quality 100, Lossless, Highest Compression
MP/s > Higher Is Better
ARMv7 Cortex-A72 . 0.10 |====================================
```

rasp benchmark	
ptsh	SD32G - ARMv7 rev 3
postmark: Disk Transaction Performance	111
ramspeed: Copy - Integer	4036.36
tinymembench: Phoronix Test Suite v5.8.1	2464.63
t-test1: 1	189.88
t-test1: 2	84.50
dolfyn: Computational Fluid Dynamics	109.81
ffte: N=256, 3D Complex FFT Routine	2400.30
cachebench: Read	4558.03
cachebench: Write	6087.82
cachebench: Read / Modify / Write	12032.37
webp: Default	7.59
webp: Quality 100	10.21
webp: Quality 100, Lossless	84.74
webp: Quality 100, Highest Compression	28.38
webp: Quality 100, Lossless, Highest Compression	252.49
graphics-magick: Swirl	4
graphics-magick: Rotate	18
graphics-magick: Sharpen	2
graphics-magick: Enhanced	3
graphics-magick: Resizing	9
graphics-magick: Noise-Gaussian	4
graphics-magick: HWB Color Space	19
coremark: CoreMark Size 666 - Iterations Per Second	29526.48
himeno: Poisson Pressure Solver	603.00
aobench: 2048 x 2048 - Total Time	104.77
encode-flac: WAV To FLAC	171.80
perl-benchmark: Pod2html	0.55683685
perl-benchmark: Interpreter	0.00726535
sqlite-speedtest: Timed Time - Size 1,000	727.35

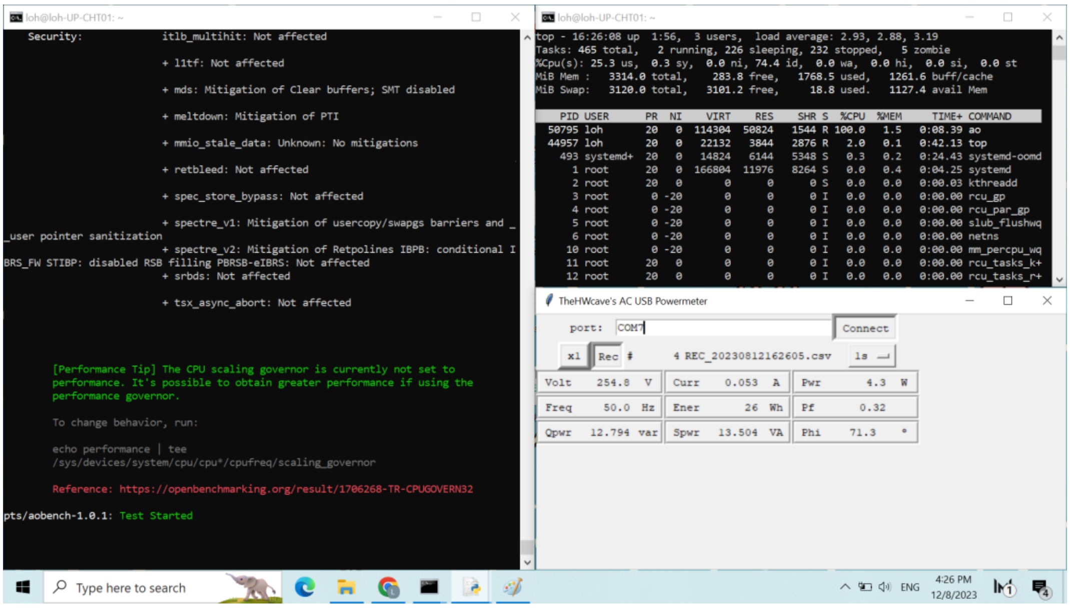
phpbench: PHP Benchmark Suite	128180
opencv: Core	972620
opencv: Video	449730
opencv: Stitching	3087285
opencv: Features 2D	2663150
opencv: Image Processing	5535373
opencv: Object Detection	906660
opencv: DNN - Deep Neural Network	7347609

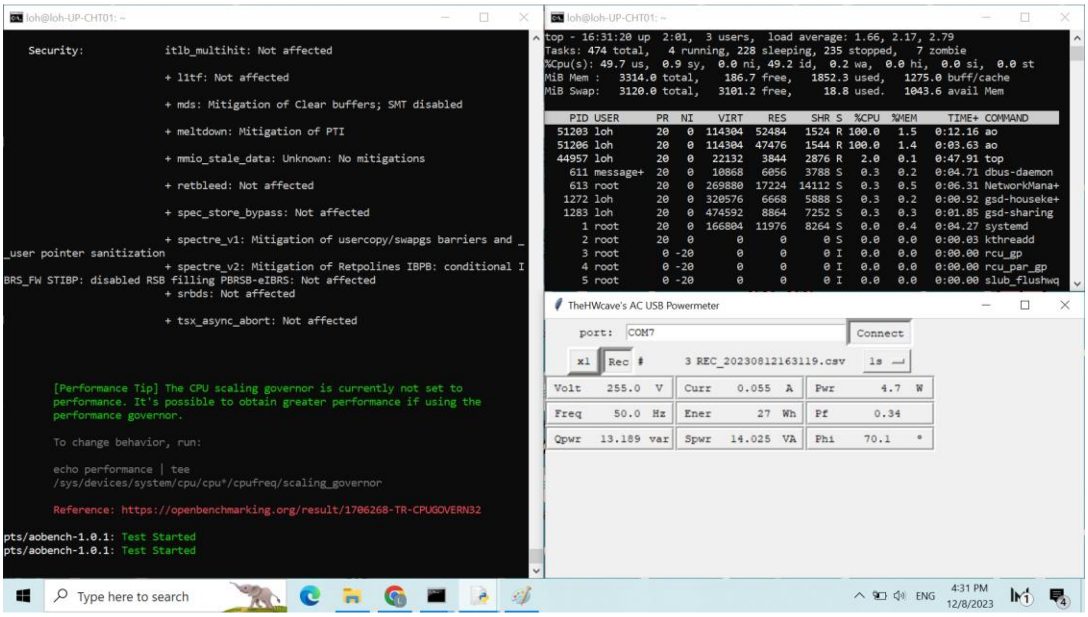
pi@raspberrypi: ~

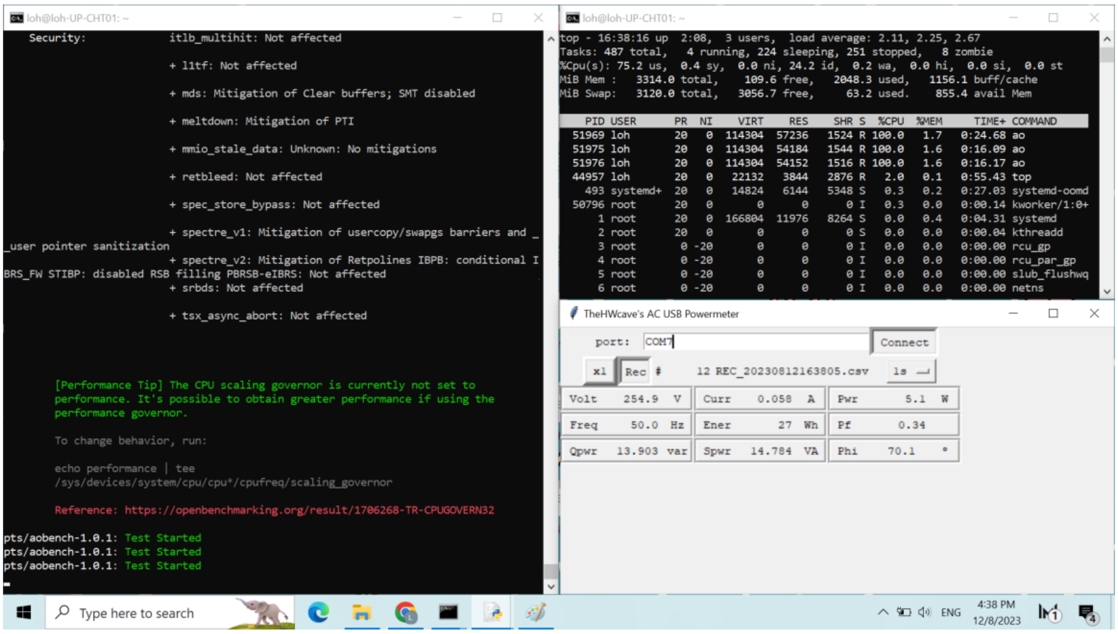
```
Performance version: 1.11.0
Report on Linux-5.15.84-v7l+-armv7l-with-glibc2.31
Number of logical CPUs: 4
Start date: 2024-04-23 08:35:06.734977
End date: 2024-04-23 08:48:43.883666

### 2to3 ###
Mean +- std dev: 1.56 sec +- 0.01 sec

### chaos ###
Mean +- std dev: 507 ms +- 4 ms

### crypto_pyaes ###
Mean +- std dev: 454 ms +- 8 ms

### float ###
Mean +- std dev: 530 ms +- 9 ms

### go ###
Mean +- std dev: 1.16 sec +- 0.01 sec

### json_loads ###
Mean +- std dev: 109 us +- 2 us

### nbody ###
Mean +- std dev: 526 ms +- 12 ms

### pathlib ###
Mean +- std dev: 123 ms +- 1 ms

### python_startup ###
Mean +- std dev: 38.9 ms +- 3.2 ms

### raytrace ###
Mean +- std dev: 2.31 sec +- 0.04 sec

### regex_compile ###
Mean +- std dev: 768 ms +- 12 ms
```

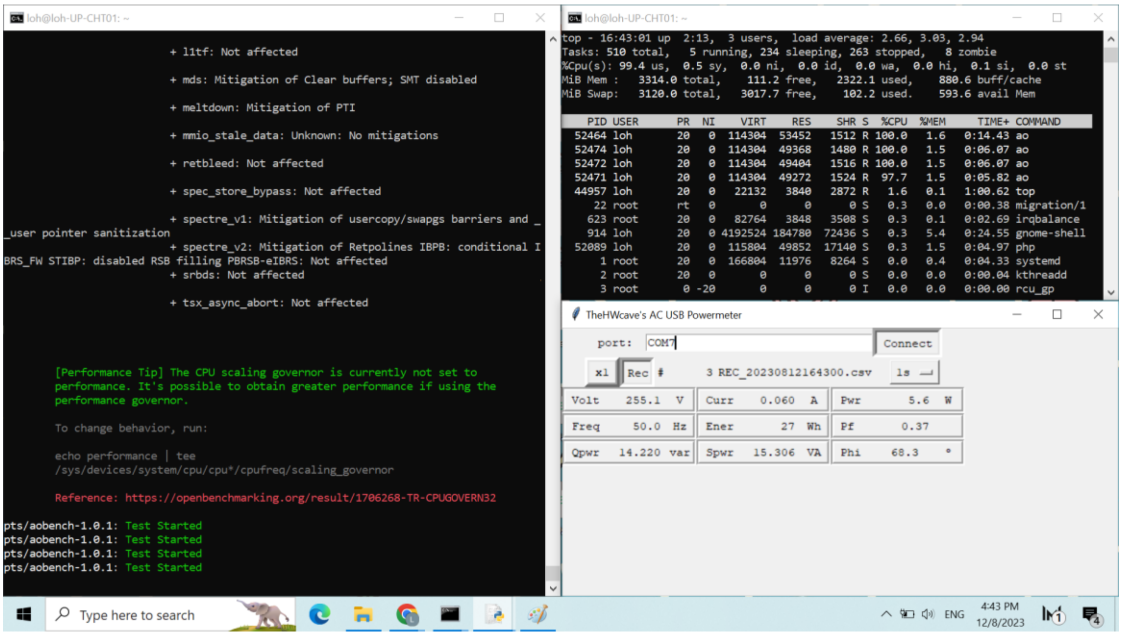
Appendix E – Power Consumption Test

E1 : AOBench

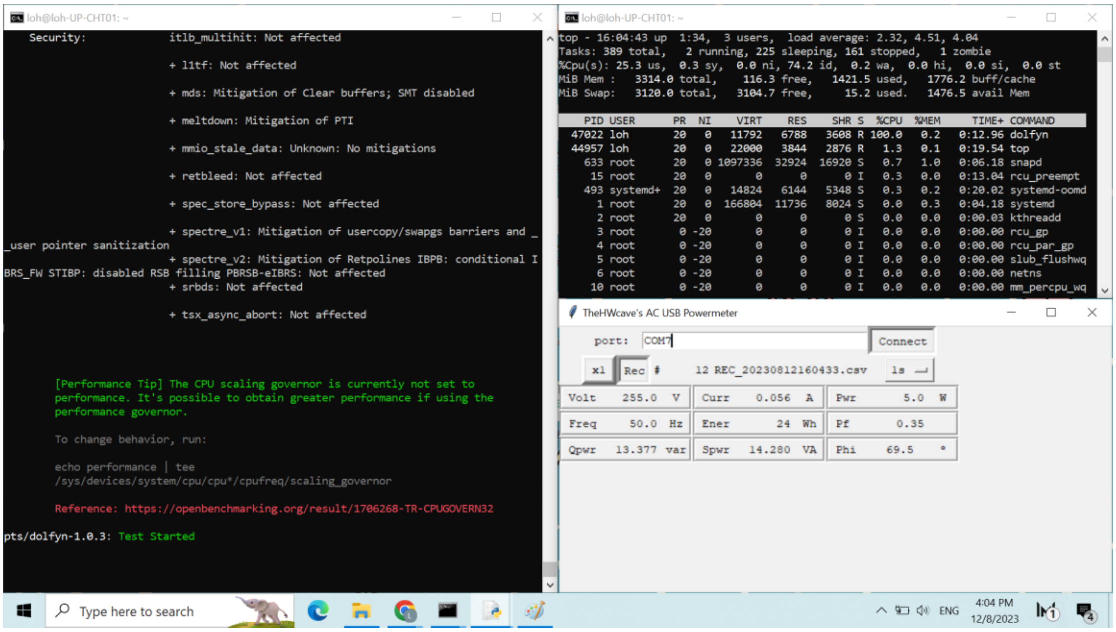


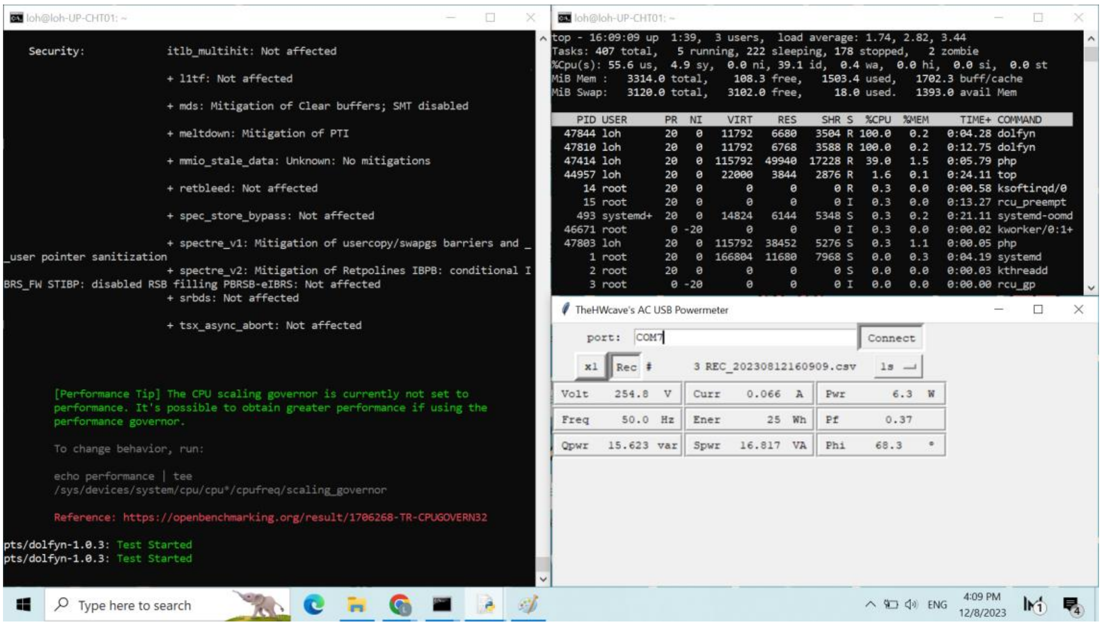


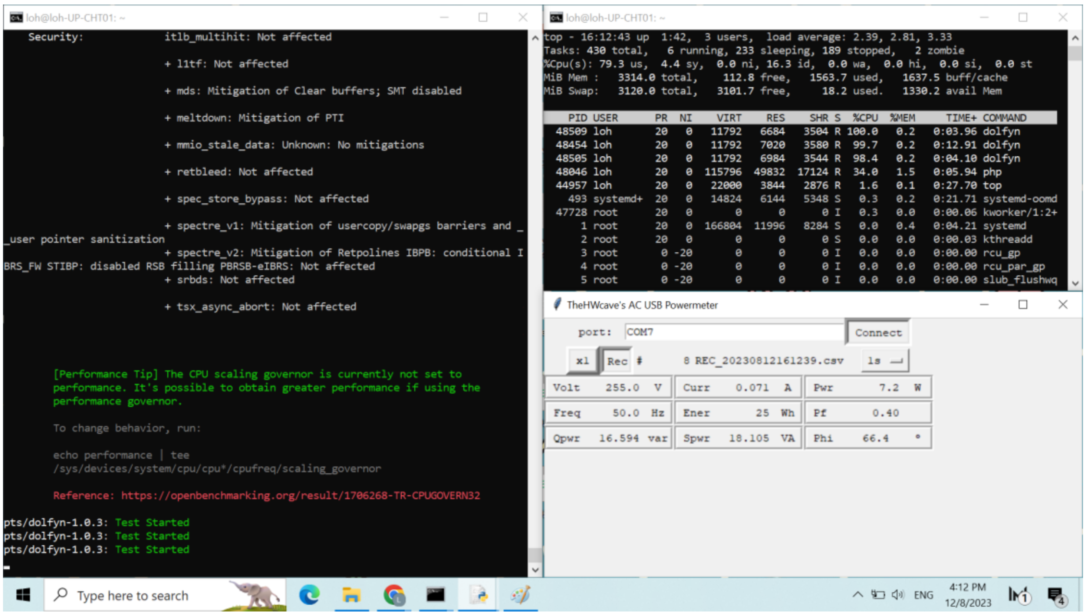

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/aobench-1.0.1: Test Started
pts/aobench-1.0.1: Test Started
pts/aobench-1.0.1: Test Started
top - 16:38:16 up 2:08, 3 users, load average: 2.11, 2.25, 2.67
Tasks: 487 total, 4 running, 224 sleeping, 251 stopped, 8 zombie
%Cpu(s): 75.2 us, 0.4 sy, 0.0 ni, 24.2 id, 0.2 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 109.6 free, 2048.3 used, 1156.1 buff/cache
MiB Swap: 3120.0 total, 3056.7 free, 63.2 used. 855.4 avail Mem
TheHWcave's AC USB Powermeter
port: COM7
Connect
12 REC_20230812163805.csv
Volt 254.9 V
Curr 0.058 A
Pwr 5.1 W
Freq 50.0 Hz
Ener 27 Wh
Pf 0.34
Qpwr 13.903 var
Spwr 14.784 VA
Phi 70.1 °
Type here to search
4:38 PM
12/8/2023

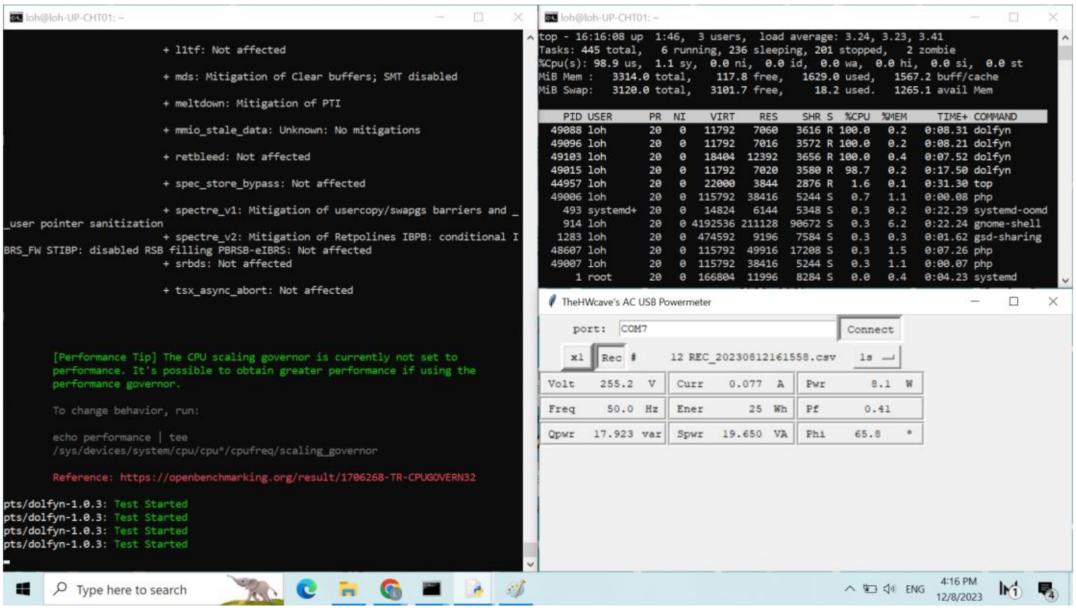

loh@loh-UP-CHT01: ~
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/aobench-1.0.1: Test Started
pts/aobench-1.0.1: Test Started
pts/aobench-1.0.1: Test Started
pts/aobench-1.0.1: Test Started
top - 16:43:01 up 2:13, 3 users, load average: 2.66, 3.03, 2.94
Tasks: 510 total, 5 running, 234 sleeping, 263 stopped, 8 zombie
%Cpu(s): 99.4 us, 0.5 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3314.0 total, 111.2 free, 2322.1 used, 880.6 buff/cache
MiB Swap: 3120.0 total, 3017.7 free, 102.2 used. 593.6 avail Mem
TheHWcave's AC USB Powermeter
port: COM7
Connect
3 REC_20230812164300.csv
Volt 255.1 V
Curr 0.060 A
Pwr 5.6 W
Freq 50.0 Hz
Ener 27 Wh
Pf 0.37
Qpwr 14.220 var
Spwr 15.306 VA
Phi 68.3 °
Type here to search
4:43 PM
12/8/2023

E2: Dolfyn

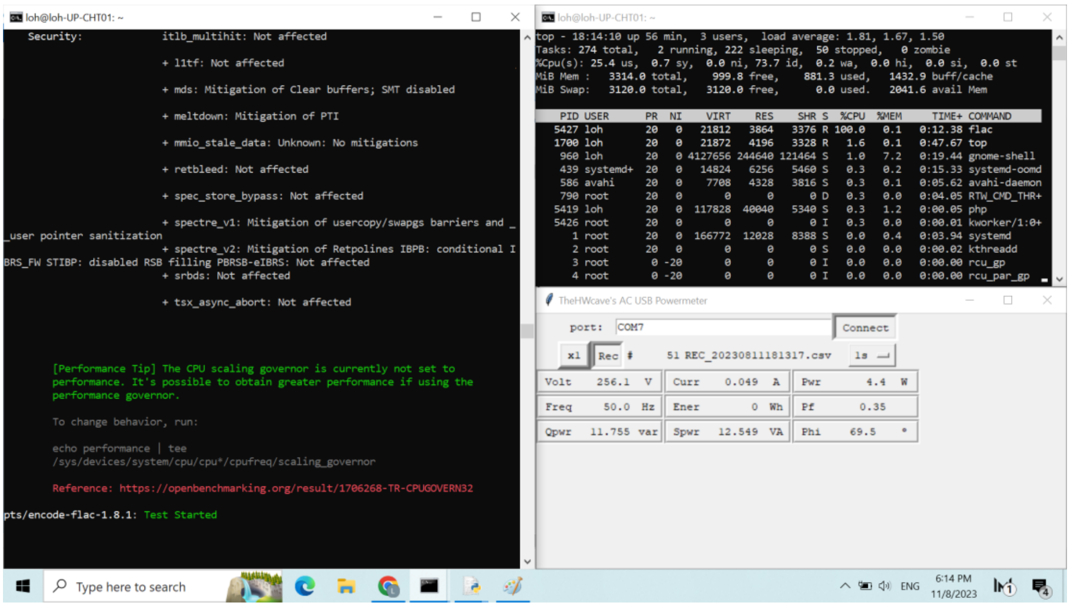

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/dolfyn-1.0.3: Test Started
top - 16:04:43 up 1:34, 3 users, load average: 2.32, 4.51, 4.04
Tasks: 389 total, 2 running, 225 sleeping, 161 stopped, 1 zombie
%Cpu(s): 25.3 us, 0.3 sy, 0.0 ni, 74.2 id, 0.2 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 116.3 free, 1421.5 used, 1776.2 buff/cache
MiB Swap: 3120.0 total, 3104.7 free, 15.2 used. 1476.5 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
47022 loh 20 0 11792 6788 3608 R 100.0 0.2 0:12.96 dolfyn
44957 loh 20 0 22000 3844 2876 R 1.3 0.1 0:19.54 top
633 root 20 0 1097336 32924 16920 S 0.7 1.0 0:06.18 snapd
15 root 20 0 0 0 0 I 0.3 0.0 0:13.04 rcu_preempt
493 systemd+ 20 0 14824 6144 5348 S 0.3 0.2 0:20.02 systemd-oomd
1 root 20 0 166804 11736 8024 S 0.0 0.3 0:04.18 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.03 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 slub_flushwq
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 netns
10 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 mm_percpu_wq
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 12 REC_20230812160433.csv 1s
Volt 255.0 V
Curr 0.056 A
Pwr 5.0 W
Freq 50.0 Hz
Ener 24 Wh
Pf 0.35
Qpwr 13.377 var
Spwr 14.280 VA
Phi 69.5 °
Type here to search
ENG
4:04 PM
12/8/2023

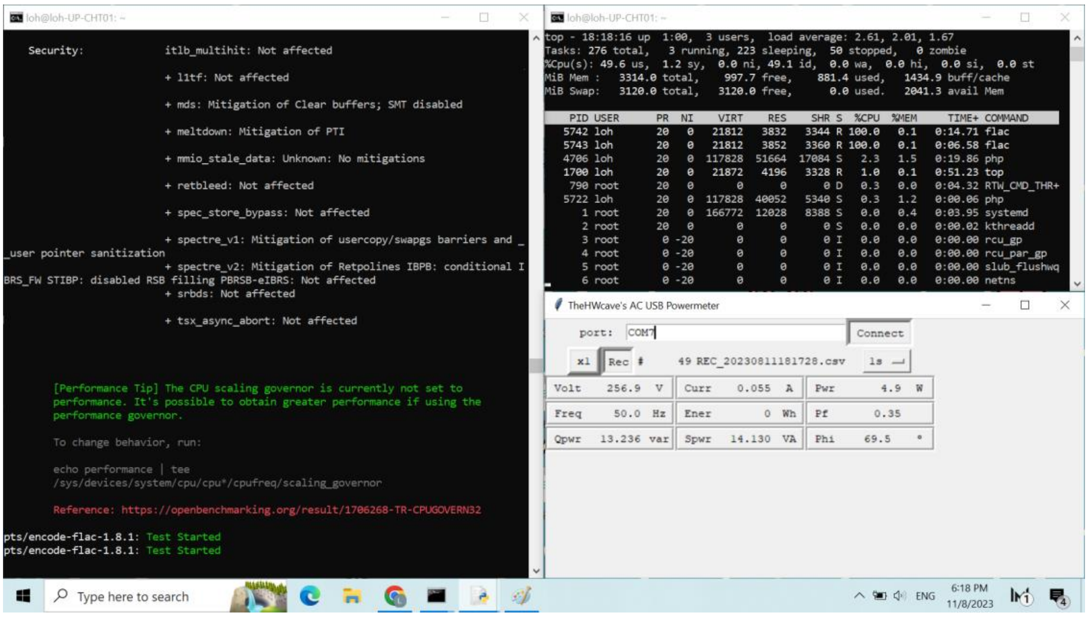

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/dolfyn-1.0.3: Test Started
pts/dolfyn-1.0.3: Test Started
top - 16:09:09 up 1:39, 3 users, load average: 1.74, 2.82, 3.44
Tasks: 407 total, 5 running, 222 sleeping, 178 stopped, 2 zombie
%Cpu(s): 55.6 us, 4.9 sy, 0.0 ni, 39.1 id, 0.4 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 108.3 free, 1503.4 used, 1702.3 buff/cache
MiB Swap: 3120.0 total, 3102.0 free, 18.0 used. 1393.0 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
47844 loh 20 0 11792 6680 3504 R 100.0 0.2 0:04.28 dolfyn
47810 loh 20 0 11792 6768 3588 R 100.0 0.2 0:12.75 dolfyn
47414 loh 20 0 115792 49940 17228 R 39.0 1.5 0:05.79 php
44957 loh 20 0 22000 3844 2876 R 1.6 0.1 0:24.11 top
14 root 20 0 0 0 0 R 0.3 0.0 0:00.58 ksoftirqd/0
15 root 20 0 0 0 0 I 0.3 0.0 0:13.27 rcu_preempt
493 systemd+ 20 0 14824 6144 5348 S 0.3 0.2 0:21.11 systemd-oomd
46671 root 0 -20 0 0 0 I 0.3 0.0 0:00.02 kworker/0:1+
47803 loh 20 0 115792 38452 5276 S 0.3 1.1 0:00.05 php
1 root 20 0 166804 11680 7968 S 0.0 0.3 0:04.19 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.03 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 3 REC_20230812160909.csv 1s
Volt 254.8 V
Curr 0.066 A
Pwr 6.3 W
Freq 50.0 Hz
Ener 25 Wh
Pf 0.37
Qpwr 15.623 var
Spwr 16.817 VA
Phi 68.3 °
Type here to search
ENG
4:09 PM
12/8/2023

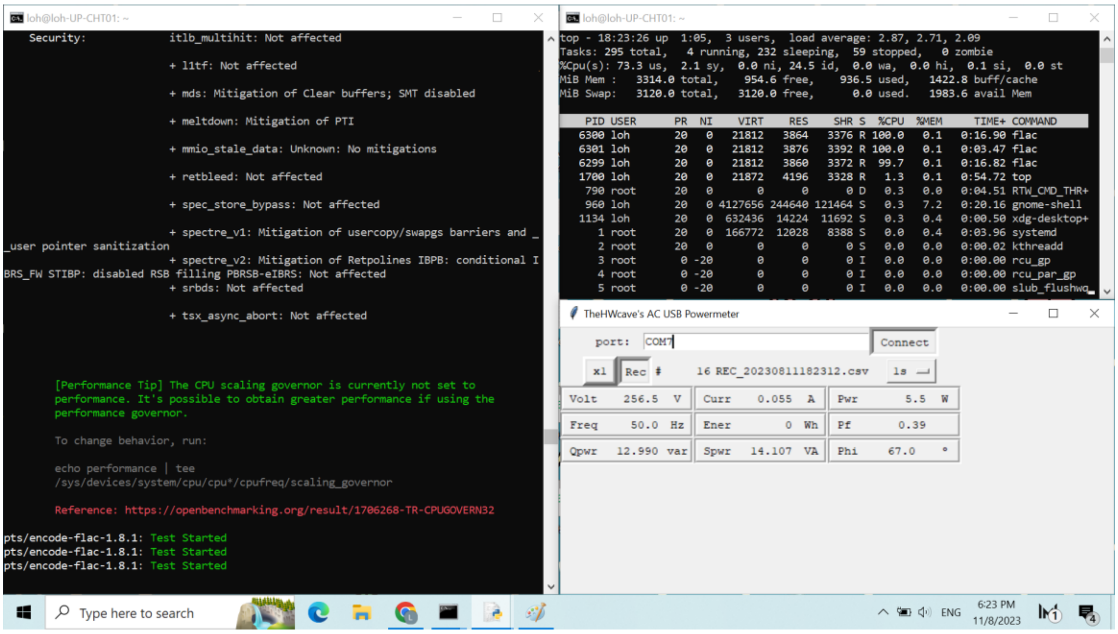


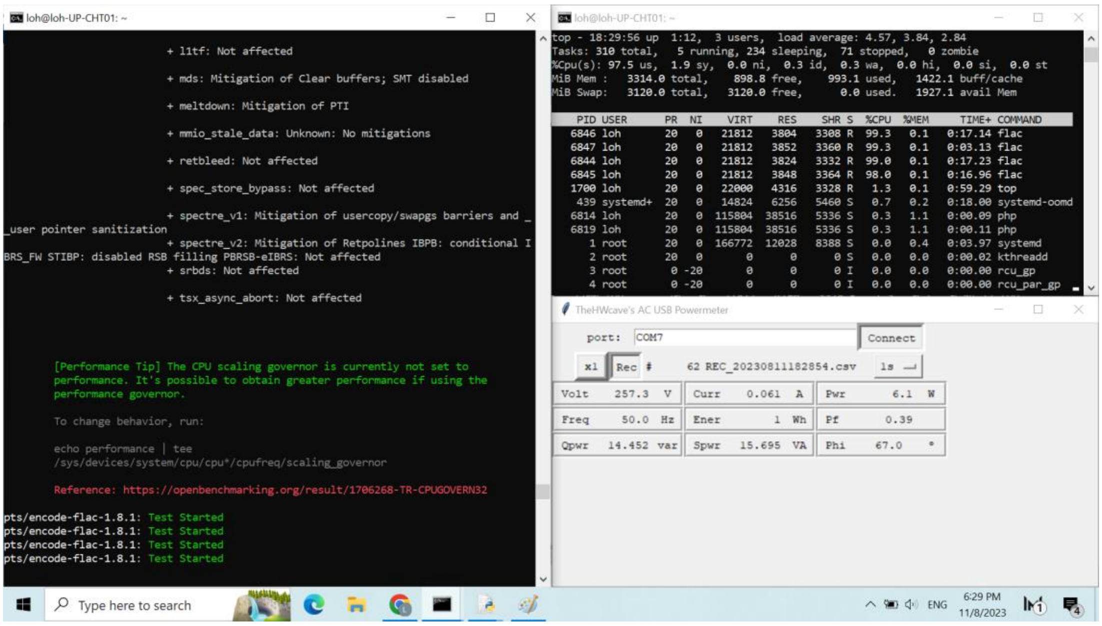


E3: FLAC Audio Encoding

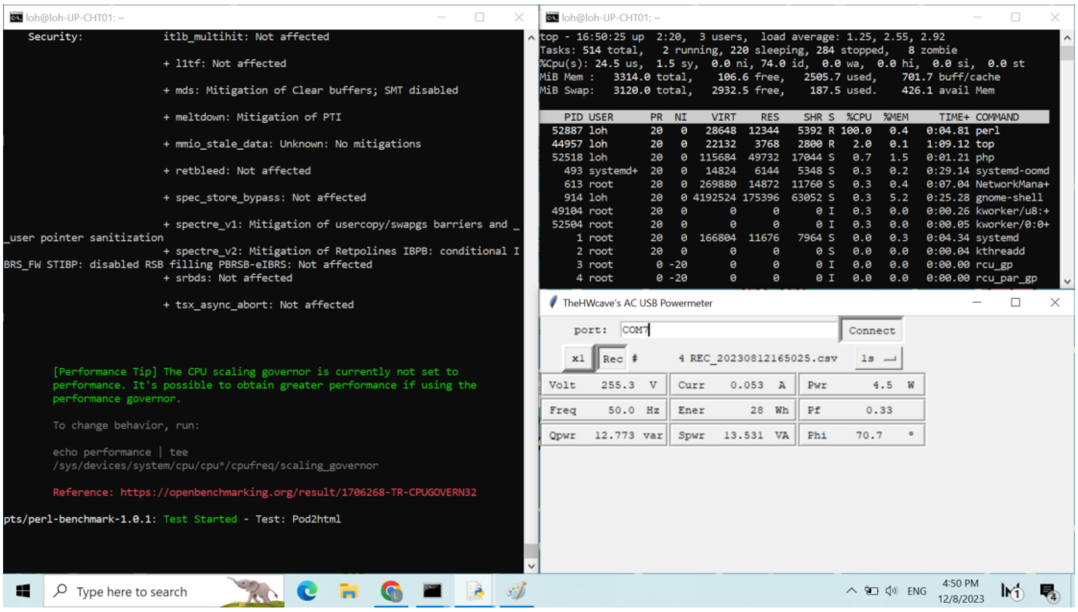

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/encode-flac-1.8.1: Test Started
top - 18:14:10 up 56 min, 3 users, load average: 1.81, 1.67, 1.50
Tasks: 274 total, 2 running, 222 sleeping, 50 stopped, 0 zombie
%Cpu(s): 25.4 us, 0.7 sy, 0.0 ni, 73.7 id, 0.2 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 999.8 free, 881.3 used, 1432.9 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 2041.6 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
5427 loh 20 0 21812 3864 3376 R 100.0 0.1 0:12.38 flac
1700 loh 20 0 21872 4196 3328 R 1.6 0.1 0:47.67 top
960 loh 20 0 4127656 244640 121464 S 1.0 7.2 0:19.44 gnome-shell
439 systemd+ 20 0 14824 6256 5460 S 0.3 0.2 0:15.33 systemd-oomd
586 avahi 20 0 7708 4328 3816 S 0.3 0.1 0:05.62 avahi-daemon
790 root 20 0 0 0 0 D 0.3 0.0 0:04.05 RTW_CMD_THR+
5419 loh 20 0 117828 40040 5340 S 0.3 1.2 0:00.05 php
5426 root 20 0 0 0 0 I 0.3 0.0 0:00.01 kworker/1:0+
1 root 20 0 166772 12028 8388 S 0.0 0.4 0:03.94 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.02 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
TheHWcave's AC USB Powermeter
port: COM7 Connect
x1 Rec # 51 REC_20230811181317.csv 1s
Volt 256.1 V Curr 0.049 A Pwr 4.4 W
Freq 50.0 Hz Ener 0 Wh Pf 0.35
Qpwr 11.755 var Spwr 12.549 VA Phi 69.5 °
Type here to search
ENG 6:14 PM 11/8/2023

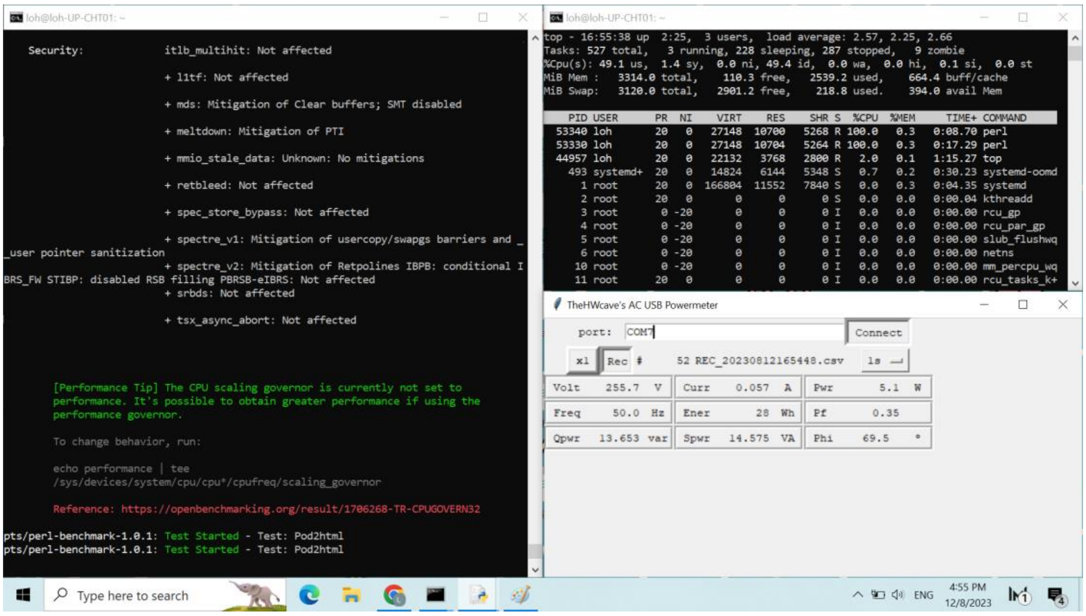

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/encode-flac-1.8.1: Test Started
pts/encode-flac-1.8.1: Test Started
top - 18:18:16 up 1:00, 3 users, load average: 2.61, 2.01, 1.67
Tasks: 276 total, 3 running, 223 sleeping, 50 stopped, 0 zombie
%Cpu(s): 49.6 us, 1.2 sy, 0.0 ni, 49.1 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 997.7 free, 881.4 used, 1434.9 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 2041.3 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
5742 loh 20 0 21812 3832 3344 R 100.0 0.1 0:14.71 flac
5743 loh 20 0 21812 3852 3360 R 100.0 0.1 0:06.58 flac
4706 loh 20 0 117828 51664 17084 S 2.3 1.5 0:19.86 php
1700 loh 20 0 21872 4196 3328 R 1.0 0.1 0:51.23 top
790 root 20 0 0 0 0 D 0.3 0.0 0:04.32 RTW_CMD_THR+
5722 loh 20 0 117828 40052 5340 S 0.3 1.2 0:00.06 php
1 root 20 0 166772 12028 8388 S 0.0 0.4 0:03.95 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.02 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 slub_flushwq
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 netns
TheHWcave's AC USB Powermeter
port: COM7 Connect
x1 Rec # 49 REC_20230811181728.csv 1s
Volt 256.9 V Curr 0.055 A Pwr 4.9 W
Freq 50.0 Hz Ener 0 Wh Pf 0.35
Qpwr 13.236 var Spwr 14.130 VA Phi 69.5 °
Type here to search
ENG 6:18 PM 11/8/2023

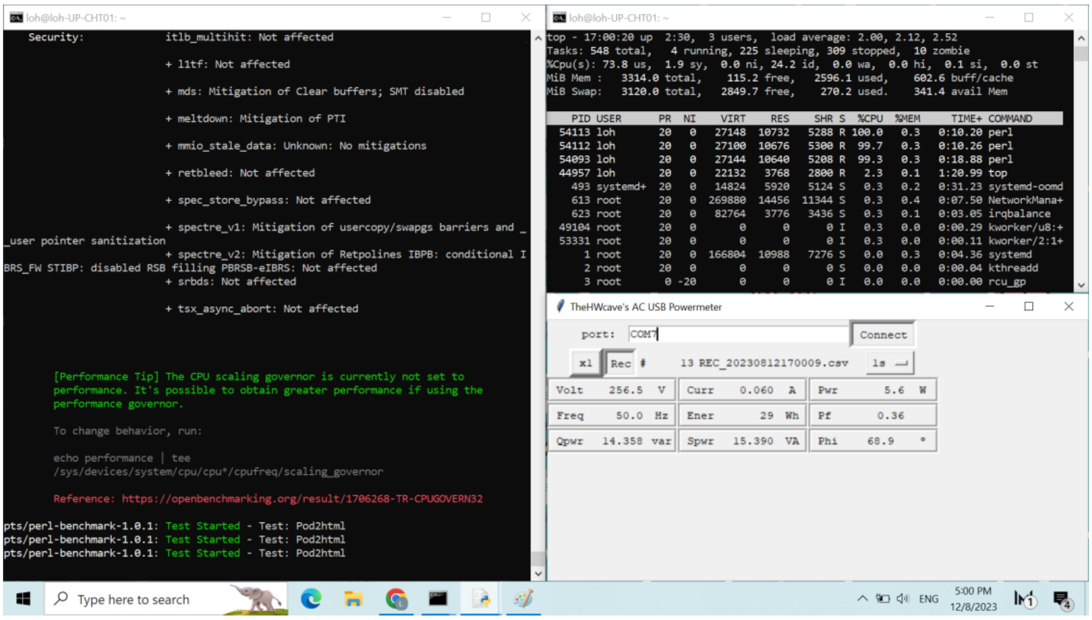


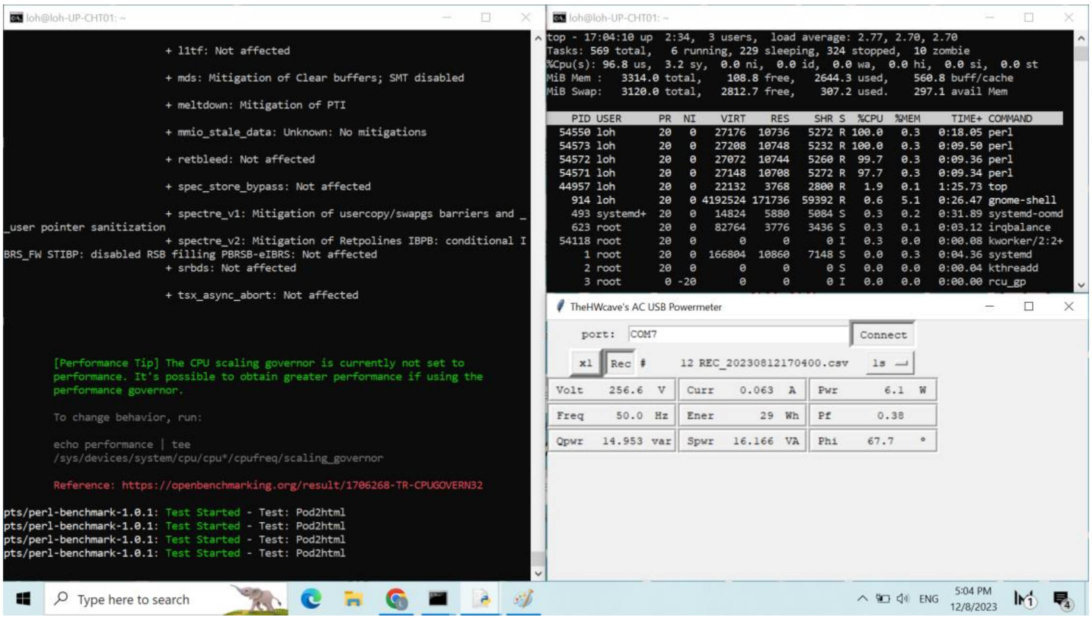


E4: Perl Benchmark (Pod2html)

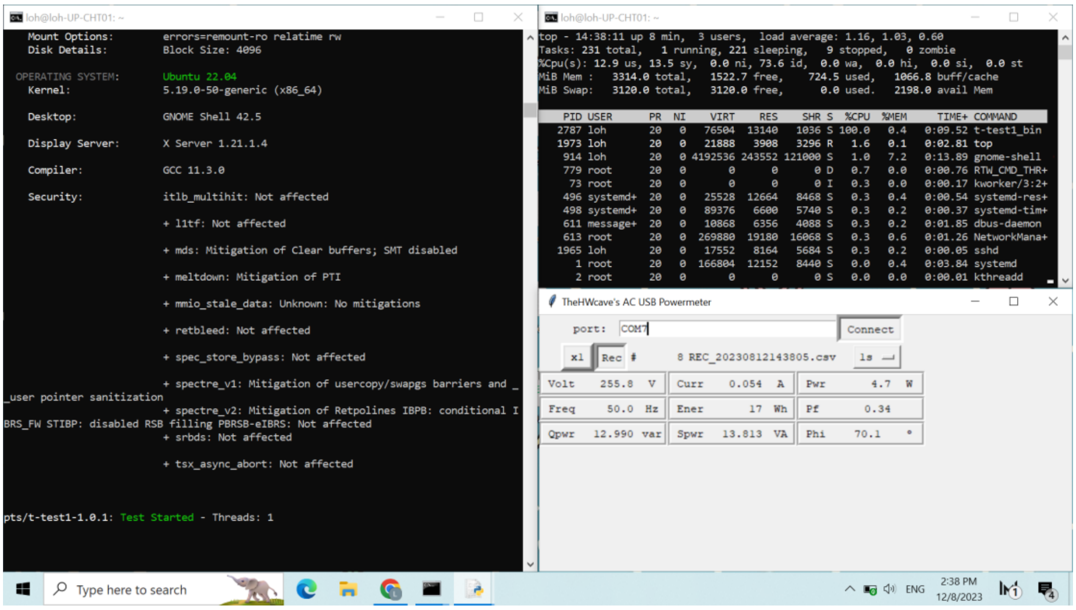


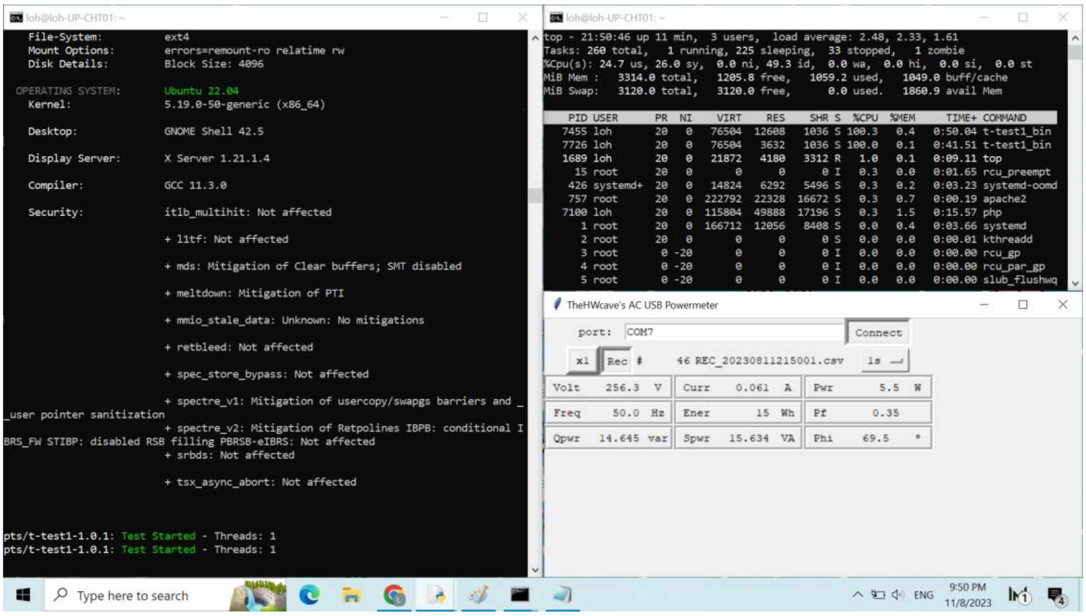


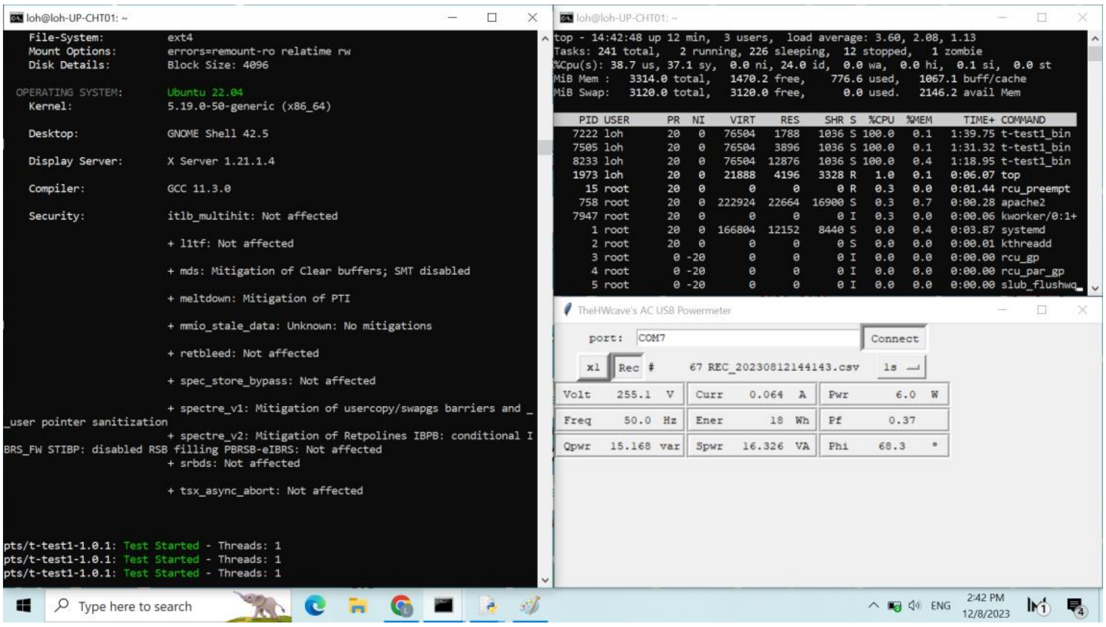

loh@loh-UP-CHT01: ~
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
top - 17:00:20 up 2:30, 3 users, load average: 2.00, 2.12, 2.52
Tasks: 548 total, 4 running, 225 sleeping, 309 stopped, 10 zombie
%Cpu(s): 73.8 us, 1.9 sy, 0.0 ni, 24.2 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3314.0 total, 115.2 free, 2596.1 used, 602.6 buff/cache
MiB Swap: 3120.0 total, 2849.7 free, 270.2 used. 341.4 avail Mem
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 13 REC_20230812170009.csv 1s
Volt 256.5 V
Curr 0.060 A
Pwr 5.6 W
Freq 50.0 Hz
Ener 29 Wh
Pf 0.36
Qpwr 14.358 var
Spwr 15.390 VA
Phi 68.9 °
Type here to search
ENG
5:00 PM
12/8/2023

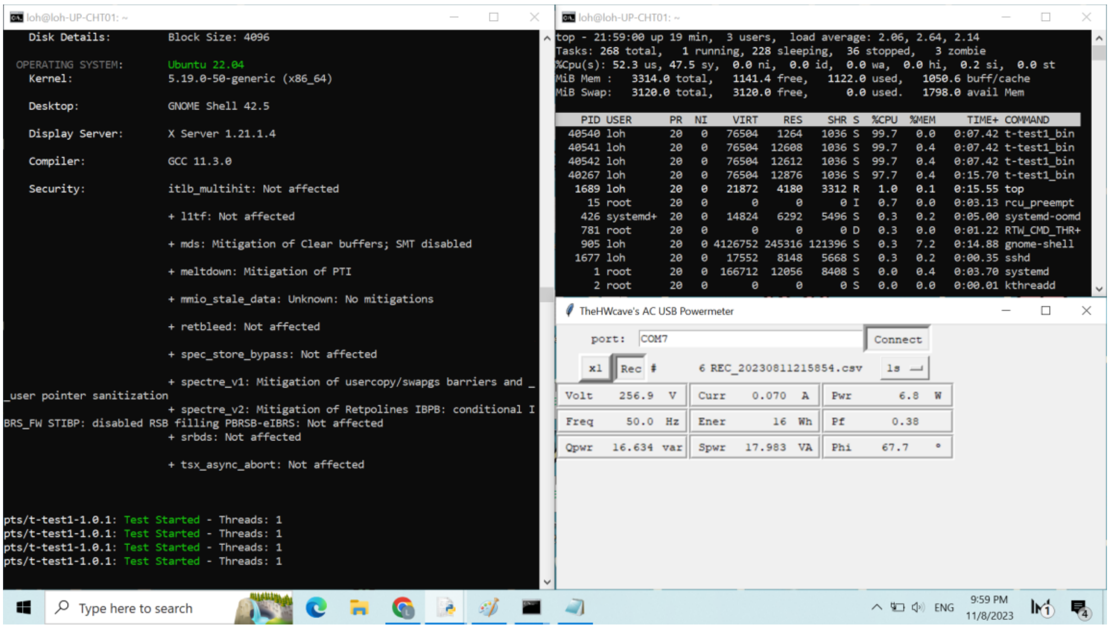

loh@loh-UP-CHT01: ~
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
[Performance Tip] The CPU scaling governor is currently not set to performance. It's possible to obtain greater performance if using the performance governor.
To change behavior, run:
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
Reference: https://openbenchmarking.org/result/1706268-TR-CPUGOVERN32
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
pts/perl-benchmark-1.0.1: Test Started - Test: Pod2html
top - 17:04:10 up 2:34, 3 users, load average: 2.77, 2.70, 2.70
Tasks: 569 total, 6 running, 229 sleeping, 324 stopped, 10 zombie
%Cpu(s): 96.8 us, 3.2 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 108.8 free, 2644.3 used, 560.8 buff/cache
MiB Swap: 3120.0 total, 2812.7 free, 307.2 used. 297.1 avail Mem
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 12 REC_20230812170400.csv 1s
Volt 256.6 V
Curr 0.063 A
Pwr 6.1 W
Freq 50.0 Hz
Ener 29 Wh
Pf 0.38
Qpwr 14.953 var
Spwr 16.166 VA
Phi 67.7 °
Type here to search
ENG
5:04 PM
12/8/2023

E5: t-test1


loh@loh-UP-CHT01: ~
Mount Options: errors=remount-ro relatime rw
Disk Details: Block Size: 4096
OPERATING SYSTEM: Ubuntu 22.04
Kernel: 5.19.0-50-generic (x86_64)
Desktop: GNOME Shell 42.5
Display Server: X Server 1.21.1.4
Compiler: GCC 11.3.0
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
pts/t-test1-1.0.1: Test Started - Threads: 1
top - 14:38:11 up 8 min, 3 users, load average: 1.16, 1.03, 0.60
Tasks: 231 total, 1 running, 221 sleeping, 9 stopped, 0 zombie
%Cpu(s): 12.9 us, 13.5 sy, 0.0 ni, 73.6 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 1522.7 free, 724.5 used, 1066.8 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 2198.0 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
2787 loh 20 0 76504 13140 1036 S 100.0 0.4 0:09.52 t-test1_bin
1973 loh 20 0 21888 3908 3296 R 1.6 0.1 0:02.81 top
914 loh 20 0 4192536 243552 121000 S 1.0 7.2 0:13.89 gnome-shell
779 root 20 0 0 0 0 D 0.7 0.0 0:00.76 RTW_CMD_THR+
73 root 20 0 0 0 0 I 0.3 0.0 0:00.17 kworker/3:2+
496 systemd+ 20 0 25528 12664 8468 S 0.3 0.4 0:00.54 systemd-res+
498 systemd+ 20 0 89376 6600 5740 S 0.3 0.2 0:00.37 systemd-tim+
611 message+ 20 0 10868 6356 4088 S 0.3 0.2 0:01.85 dbus-daemon
613 root 20 0 269880 19180 16068 S 0.3 0.6 0:01.26 NetworkMana+
1965 loh 20 0 17552 8164 5684 S 0.3 0.2 0:00.05 sshd
1 root 20 0 166804 12152 8440 S 0.0 0.4 0:03.84 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.01 kthreadd
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 8 REC_20230812143805.csv 1s
Volt 255.8 V
Curr 0.054 A
Pwr 4.7 W
Freq 50.0 Hz
Ener 17 Wh
Pf 0.34
Qpwr 12.990 var
Spwr 13.813 VA
Phi 70.1 °
Type here to search
ENG
2:38 PM
12/8/2023


loh@loh-UP-CHT01: ~
File-System: ext4
Mount Options: errors=remount-ro relatime rw
Disk Details: Block Size: 4096
OPERATING SYSTEM: Ubuntu 22.04
Kernel: 5.19.0-50-generic (x86_64)
Desktop: GNOME Shell 42.5
Display Server: X Server 1.21.1.4
Compiler: GCC 11.3.0
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
top - 21:50:46 up 11 min, 3 users, load average: 2.48, 2.33, 1.61
Tasks: 260 total, 1 running, 225 sleeping, 33 stopped, 1 zombie
%Cpu(s): 24.7 us, 26.0 sy, 0.0 ni, 49.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3314.0 total, 1205.8 free, 1059.2 used, 1049.0 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 1860.9 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
7455 loh 20 0 76504 12608 1036 S 100.3 0.4 0:50.04 t-test1_bin
7726 loh 20 0 76504 3632 1036 S 100.0 0.1 0:41.51 t-test1_bin
1689 loh 20 0 21872 4180 3312 R 1.0 0.1 0:09.11 top
15 root 20 0 0 0 0 I 0.3 0.0 0:01.65 rcu_preempt
426 systemd+ 20 0 14824 6292 5496 S 0.3 0.2 0:03.23 systemd-oomd
757 root 20 0 222792 22328 16672 S 0.3 0.7 0:00.19 apache2
7100 loh 20 0 115804 49888 17196 S 0.3 1.5 0:15.57 php
1 root 20 0 166712 12056 8408 S 0.0 0.4 0:03.66 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.01 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 slub_flushwq
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 46 REC_20230811215001.csv 1s
Volt 256.3 V
Curr 0.061 A
Pwr 5.5 W
Freq 50.0 Hz
Ener 15 Wh
Pf 0.35
Qpwr 14.645 var
Spwr 15.634 VA
Phi 69.5 °
Type here to search
ENG
9:50 PM
11/8/2023


loh@loh-UP-CHT01: ~
File-System: ext4
Mount Options: errors=remount-ro relatime rw
Disk Details: Block Size: 4096
OPERATING SYSTEM: Ubuntu 22.04
Kernel: 5.19.0-50-generic (x86_64)
Desktop: GNOME Shell 42.5
Display Server: X Server 1.21.1.4
Compiler: GCC 11.3.0
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
top - 14:42:48 up 12 min, 3 users, load average: 3.60, 2.08, 1.13
Tasks: 241 total, 2 running, 226 sleeping, 12 stopped, 1 zombie
%Cpu(s): 38.7 us, 37.1 sy, 0.0 ni, 24.0 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3314.0 total, 1470.2 free, 776.6 used, 1067.1 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 2146.2 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
7222 loh 20 0 76504 1788 1036 S 100.0 0.1 1:39.75 t-test1_bin
7505 loh 20 0 76504 3896 1036 S 100.0 0.1 1:31.32 t-test1_bin
8233 loh 20 0 76504 12876 1036 S 100.0 0.4 1:18.95 t-test1_bin
1973 loh 20 0 21888 4196 3328 R 1.0 0.1 0:06.07 top
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 67 REC_20230812144143.csv 1s
Volt 255.1 V
Curr 0.064 A
Pwr 6.0 W
Freq 50.0 Hz
Ener 18 Wh
Pf 0.37
Qpwr 15.168 var
Spwr 16.326 VA
Phi 68.3 °
Type here to search
ENG
2:42 PM
12/8/2023

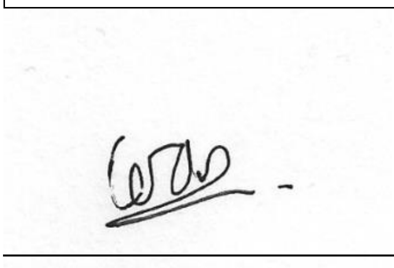

loh@loh-UP-CHT01: ~
Disk Details: Block Size: 4096
OPERATING SYSTEM: Ubuntu 22.04
Kernel: 5.19.0-50-generic (x86_64)
Desktop: GNOME Shell 42.5
Display Server: X Server 1.21.1.4
Compiler: GCC 11.3.0
Security: itlb_multihit: Not affected
+ l1tf: Not affected
+ mds: Mitigation of Clear buffers; SMT disabled
+ meltdown: Mitigation of PTI
+ mmio_stale_data: Unknown: No mitigations
+ retbleed: Not affected
+ spec_store_bypass: Not affected
+ spectre_v1: Mitigation of usercopy/swapgs barriers and _user pointer sanitization
+ spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: disabled RSB filling PBRSB-eIBRS: Not affected
+ srbds: Not affected
+ tsx_async_abort: Not affected
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
pts/t-test1-1.0.1: Test Started - Threads: 1
top - 21:59:00 up 19 min, 3 users, load average: 2.06, 2.64, 2.14
Tasks: 268 total, 1 running, 228 sleeping, 36 stopped, 3 zombie
%Cpu(s): 52.3 us, 47.5 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st
MiB Mem : 3314.0 total, 1141.4 free, 1122.0 used, 1050.6 buff/cache
MiB Swap: 3120.0 total, 3120.0 free, 0.0 used. 1798.0 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
40540 loh 20 0 76504 1264 1036 S 99.7 0.0 0:07.42 t-test1_bin
40541 loh 20 0 76504 12608 1036 S 99.7 0.4 0:07.42 t-test1_bin
40542 loh 20 0 76504 12612 1036 S 99.7 0.4 0:07.42 t-test1_bin
40267 loh 20 0 76504 12876 1036 S 97.7 0.4 0:15.70 t-test1_bin
1689 loh 20 0 21872 4180 3312 R 1.0 0.1 0:15.55 top
TheHWcave's AC USB Powermeter
port: COM7
Connect
x1 Rec # 6 REC_20230811215854.csv 1s
Volt 256.9 V
Curr 0.070 A
Pwr 6.8 W
Freq 50.0 Hz
Ener 16 Wh
Pf 0.38
Qpwr 16.634 var
Spwr 17.983 VA
Phi 67.7 °
Type here to search
ENG
9:59 PM
11/8/2023

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 2**
Student Name & ID: 21ACB01871	
Supervisor: Ts. Wong Chee Siang	
Project Title: Performance Evaluation of StarFive VisionFive V2 and comparison to Intel UP and Raspberry Pi 4B	

1. WORK DONE
[Please write the details of the work done in the last fortnight.] Done setup on starfive v2.
2. WORK TO BE DONE
Conduct benchmarks and Machine Learning benchmark on StarFive V2.
3. PROBLEMS ENCOUNTERED
Attempted to install wifi driver for StarFive V2 but failed. Will use ethernet cable as solution.
4. SELF EVALUATION OF THE PROGRESS
So far no problem.



Student's signature Supervisor's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 4**
Student Name & ID: 21ACB01871	
Supervisor: Ts. Wong Chee Siang	
Project Title: Performance Evaluation of StarFive VisionFive V2 and comparison to Intel UP and Raspberry Pi 4B	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Conducted performance evaluation on StarFive V2.
2. WORK TO BE DONE Machine Learning Benchmarks and Power Consumption test on StarFive V2.
3. PROBLEMS ENCOUNTERED Attempted to install pytorch for V2 but failed as it lacks support, will find alternative option.
4. SELF EVALUATION OF THE PROGRESS So far no problem.

Student's signature

Supervisor's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 6**
Student Name & ID: 21ACB01871	
Supervisor: Ts. Wong Chee Siang	
Project Title: Performance Evaluation of StarFive VisionFive V2 and comparison to Intel UP and Raspberry Pi 4B	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Done power consumption test and ML benchmarks on V2
2. WORK TO BE DONE Redo tests for Intel UP and starfive v1 in init level 3.
3. PROBLEMS ENCOUNTERED No problem.
4. SELF EVALUATION OF THE PROGRESS So far no problem.

Student's signature

Supervisor's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 8**
Student Name & ID: 21ACB01871	
Supervisor: Ts. Wong Chee Siang	
Project Title: Performance Evaluation of StarFive VisionFive V2 and comparison to Intel UP and Raspberry Pi 4B	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Done benchmarks and power consumption tests on Intel UP and starfive v1.
2. WORK TO BE DONE Performance evaluation on Raspberry Pi 4B.
3. PROBLEMS ENCOUNTERED No problem.
4. SELF EVALUATION OF THE PROGRESS So far no problem.

Student's signature

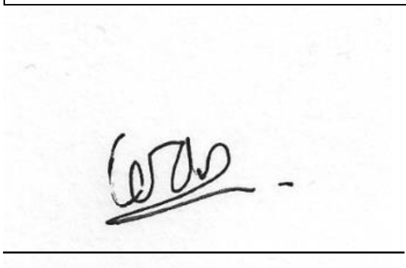
Supervisor's signature

FINAL YEAR PROJECT WEEKLY REPORT

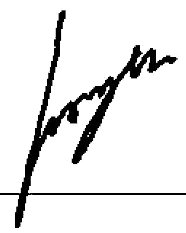
(Project II)

Trimester, Year: Y3S3	**Study week no.: 10**
Student Name & ID: 21ACB01871	
Supervisor: Ts. Wong Chee Siang	
Project Title: Performance Evaluation of StarFive VisionFive V2 and comparison to Intel UP and Raspberry Pi 4B	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Done all benchmarks for RPI 4B.
2. WORK TO BE DONE Documentation work.
3. PROBLEMS ENCOUNTERED No problem.
4. SELF EVALUATION OF THE PROGRESS So far no problem.



Student's signature



Supervisor's signature

POSTER

Performance Benchmark of StarFive VisionFive V2 and Comparison to StarFive VisionFive and Intel UP Board

WHAT IS RISC-V?

RISC-V is an open source ISA that can be used to build processors.
Contrary to proprietary ISAs, RISC-V is open source and values collaboration and openness, enabling programmers to modify processors to suit requirements.
The RISC-V ecosystem includes a wide range of applications, from embedded devices to supercomputers, opening the door for processor architectures that are optimized for purposes.
The tested subjects are VisionFive, VisionFive V2 and Intel UP Board.

HOW TO EVALUATE?

The Phoronix Test Suite is a powerful tool for evaluating hardware and software performance concurrently with the RISC-V revolution. By providing a comprehensive set of tests that cover many facets of system performance, from compute throughput to graphics rendering, the Phoronix Test Suite streamlines benchmarking.

pts Phoronix Test Suite

WHAT ARE EVALUATED?

In this project, we will measure the performance metrics in the benchmarks provided by Phoronix test suite.

The power consumption of the boards during the execution of the benchmarks are also measured.

The results obtained from the benchmarks are compared.

We will also test different compiler flag settings on the tests to seek improvements.

PLAGIARISM CHECK RESULT

Performance Evaluation of StarFive VisionFive V2 and Comparison to Intel UP Board and Raspberry Pi 4B

ORIGINALITY REPORT

7%	6%	3%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.theseus.fi Internet Source	1%
2	github.com Internet Source	1%
3	openbenchmarking.org Internet Source	1%
4	ia800502.us.archive.org Internet Source	1%
5	ericdraken.com Internet Source	<1%
6	planet.linux.org.au Internet Source	<1%
7	perf.mvermeulen.com Internet Source	<1%
8	"Parallel Processing and Applied Mathematics", Springer Nature, 2008 Publication	<1%

sizeof.cat

9	Internet Source	<1 %
10	www.linode.com Internet Source	<1 %
11	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1 %
12	cesg.tamu.edu Internet Source	<1 %
13	wiki.seeedstudio.com Internet Source	<1 %
14	Submitted to Universiti Teknologi Petronas Student Paper	<1 %
15	Alaa Hussein Abdulaal, Morteza Valizadeh, Mehdi Chehel Amirani, A.F.M. Shahen Shah. "A self-learning deep neural network for classification of breast histopathological images", Biomedical Signal Processing and Control, 2024 Publication	<1 %
16	helpdeskgeek.com Internet Source	<1 %
17	Submitted to Universiti Teknologi Malaysia Student Paper	<1 %
18	archive.org Internet Source	<1 %

19	Submitted to Brunel University Student Paper	<1%
20	Submitted to Bolton Institute of Higher Education Student Paper	<1%
21	www.electronics-lab.com Internet Source	<1%
22	rvspace.org Internet Source	<1%
23	www.essays.se Internet Source	<1%
24	mycroft-ai.gitbook.io Internet Source	<1%
25	www.curee.org Internet Source	<1%
26	MySQL for the Internet of Things, 2016. Publication	<1%
27	ima.ejournal.org Internet Source	<1%
28	opendata.uni-halle.de Internet Source	<1%
29	www.electromaker.io Internet Source	<1%

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date:	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of	LOH LE ZHOU
ID Number(s)	2101871
Programme / Course	BACHELOR OF COMPUTER SCIENCE
Title of Final Year Project	PERFORMANCE EVALUATION OF STARFIVE VISIONFIVE V2 AND COMPARISON TO INTEL UP AND RASPBERRY PI 4B

Similarity	Supervisor's Comments
Overall similarity index: 7 **%** **Similarity by source** Internet Sources: 6 % Publications: 3 % Student Papers: 4 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words**	

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Signature of Co-Supervisor

Name: Wong Chee Siang

Name: ______________________

Date: 23/4/2024

Date: ______________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	21ACB01871
Student Name	Loh Le Zhou
Supervisor Name	Ts. Wong Chee Siang

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content

√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

____

(Signature of Student)

Date: 20/4/2024