VRIJE
UNIVERSITEIT
BRUSSEL

In order to be awarded the Master's Degree in industrial science: Electronic-ICT with specialization embedded systems

Benchmarking single board computers for embedded systems

Zineddine Wakrim

August 20, 2018

Jan Lemeire

Engineering science

Abstract

Acknowledgements

Contents

1 Introduction **10**

2 Related Work **11**
2.1 Existing benchmarks . . . 11
2.1.1 Running existing benchmark . . . 11

3 Pixy cam **21**
3.1 Limit of Pixy . . . 21

4 Comparison **24**
4.1 board comparison . . . 24

5 Generic vision Benchmark **26**

6 Results **27**

7 Conclusion **28**

8 Appendix **30**
.1 Appendix 1 : Result benchmark [1] . . . 31
.2 Appendix 1 : Result benchmark [1] . . . 32

List of Figures

2.1 FFTW benchmark. 12
2.2 FFTW/Cost benchmark. 12
2.3 John The Ripper benchmark. 13
2.4 John The Ripper / Cost benchmark. 13
2.5 C-Ray benchmark. 14
2.6 C-Ray/Cost benchmark. 14
2.7 smallpt benchmark. 15
2.8 smallpt/cost benchmark. 15
2.9 flac audio benchmark. 16
2.10 flac audio /cost benchmark. 16
2.11 Redis(Set) benchmark. 17
2.12 Redis(set)/cost benchmark. 17
2.13 Redis(Get) benchmark. 18
2.14 Redis(Get)/cost benchmark. 18
2.15 Himeno benchmark. 19
2.16 Himeno/cost benchmark. 19

3.1 Red color detection. 22
3.2 Rebound object detection. 22
3.3 white color detection. 23

List of Tables

4.1 Comparison board table [11] [12] [13] [10] 25

1 Introduction

This thesis will benchmark vision algorithms on several single board computers. The purpose of the thesis is to check how good a single board computer can do some real time image processing.

Vision is a hype for the moment in the IT world, autonomous car, camera with number plate reconision,...

With the ... of the Arduino a lot of hoobbies want to put vision in robots or in other application. Because a microcontroller like the one on the Arduino is not powerful enough to perform real time video processing It is mandatory to combine it with a more powerful computer or a single board computer. in severals application the single board computer will send the image to a more powerful computer or server and it is this one that will perform the imageprocessing. For our application all the image processing will be done on the single board computer.

Two users were defined: students or person that are not alaise with programming. In this case the supposition is that the user will use python. The second user is a person that is more alaise with coding and that will use or C. it will be interesting to see if there is a big difference between python algorithms and C.

Last but not least is also the power consumption. Because most of the application like robots needs a battery to be powered it will be interesting to check the different power consumption during the test.

2 Related Work

2.1 Existing benchmarks

On the market there are a lot of single board computers. They all have different capabilities and cost.[1] Because this big diversity the question that we have to ask is how should we decide which board is right for our needs? Benchmarkt [1] created two categories: Boards under $100 and boards over the $100. The boards under the $100 is more for DIY maker. With this kind of boards it is possible to do a lot be the boards are limited in computing power. Over the $100 the boards are more powerful and built for more specific purpose like machine vision and robotics.[1]

The result of benchmark [1] can be found in appendix .2. The conclusion of this benchmark is that each single board computer comes with his own capabilities, and one isn't perfect for all applications.

An other interesting benchmark is the one from Phoronix.com. The Phoronix Test Suite is an benchmarking platform available for Linux, Solaris, Mac OS X, and BSD operating systems. [2] Benchmark [3] tested several single board computers, some similar two boards are similar as the one tested in this thesis, The jetston TX2 and the Raspberry Pi 3. The benchmark tested performance and performance-per-dollar. In all the performance benchmark the jetson TX2 did it better than the other boards. About cost the Jetson is the best one when using some graphic [3]

2.1.1 Running existing benchmark

Eight benchmarks were tested. The same as[3]

FFTW

The FFTW benchmark is a C subroutine for computing DFT in one of more direction. Figure 2.1 show the result of the benchmark. The Jetston TX 2 scored the best with 1697 Mflops. The Udoo board was the worst one.

Figure 2.2 shows the FFTW benchmarkt devide by the price of the board. The Udoo board is still the worst one. The best one is the Odroid xu4 that is more than 4 times better than the second one(Rasperry pi 3).

Figure 2.1: FFTW benchmark.

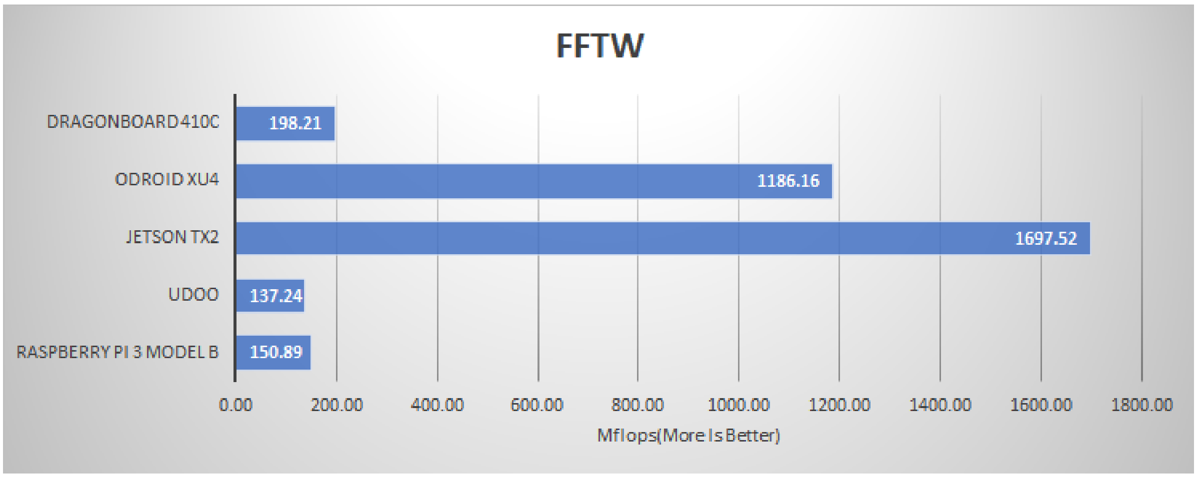
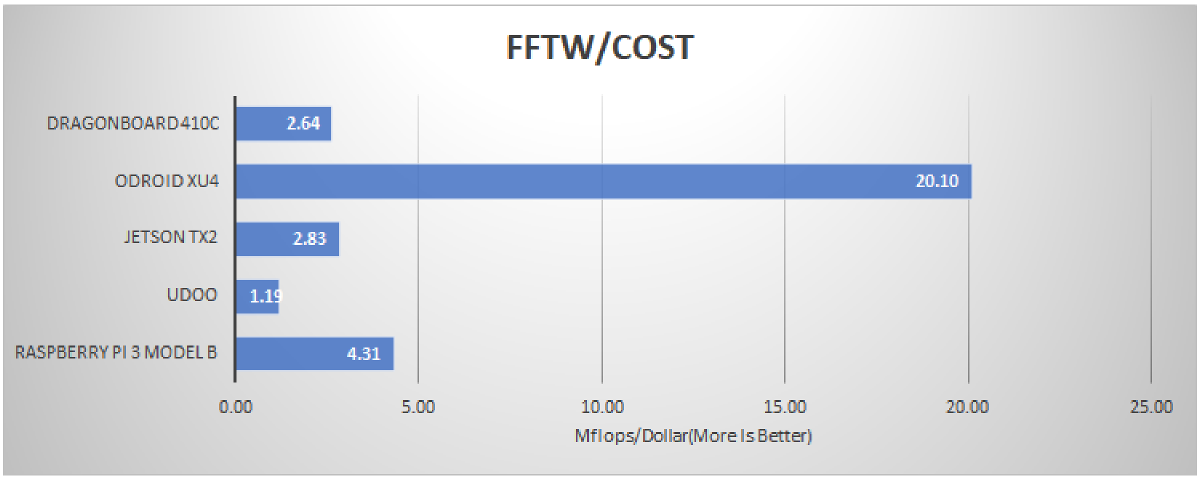


Figure 2.2: FFTW/Cost benchmark.



John The Ripper

John The Ripper algorithm is a password cracker. The odroid Xu4 scored the best in the benchmark as well as the cost benchmark . The worst one is the Udoo board(2.3). The worst value for money board is the Jetson TX2 (2.4).

Figure 2.3: John The Ripper benchmark.

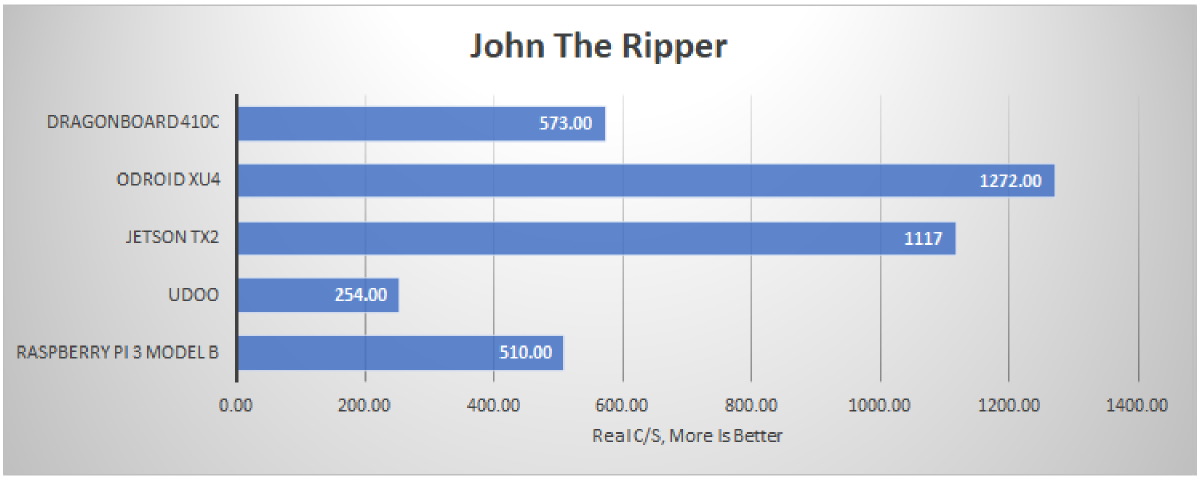
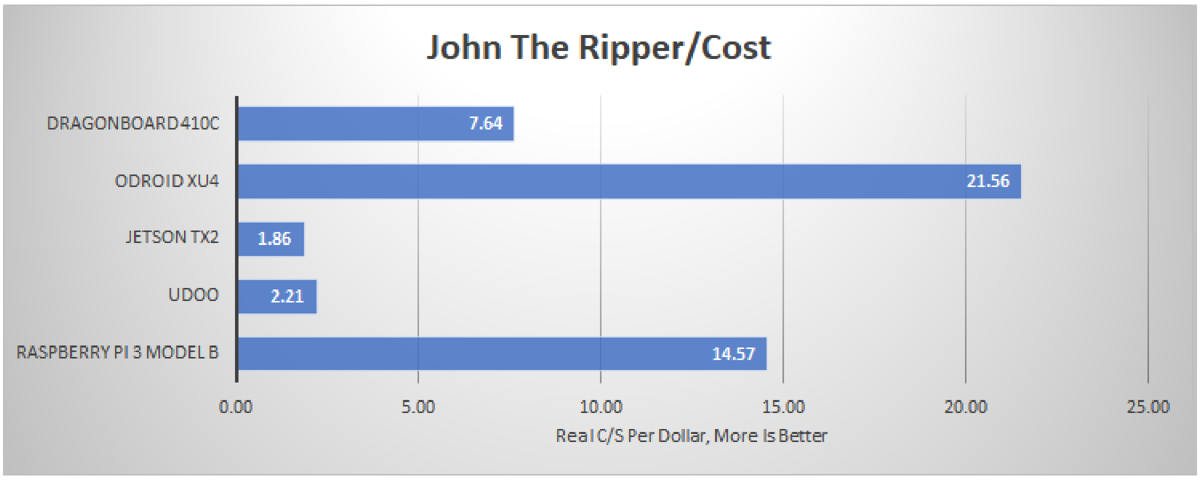


Figure 2.4: John The Ripper / Cost benchmark.



C-ray

The C-Ray is a raytracer it is used to test the floating-point CPU performance. The one used on this benchmark is multi-threaded(16 threads per core). It will generate a 1600 x 1200 image and will shoot 8 rays per pixel for anti-aliasing. [4]

The Jetson TX2 scored the best on the benchmark but the Odroid Xu4 Has the best price value score.

Figure 2.5: C-Ray benchmark.

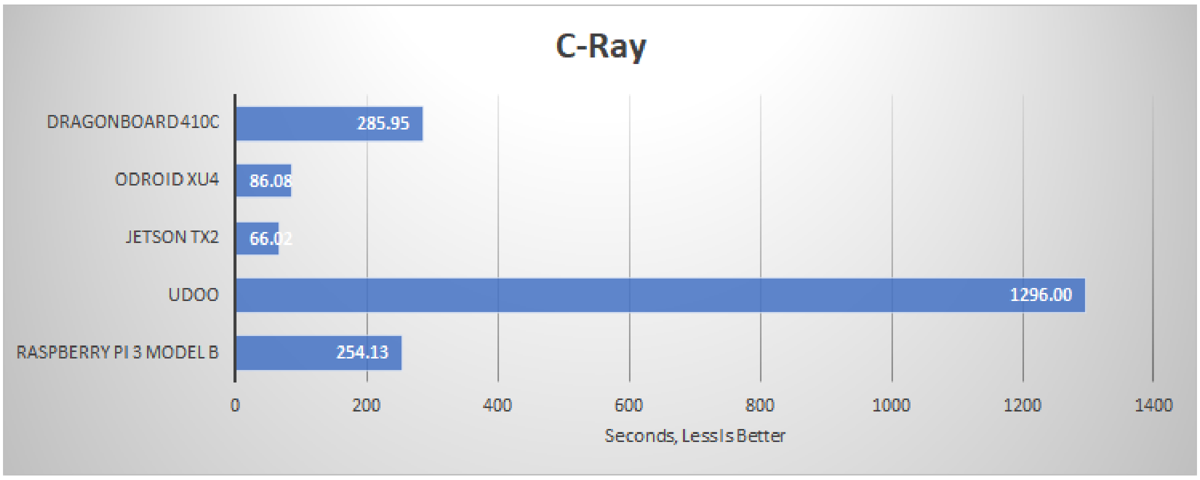
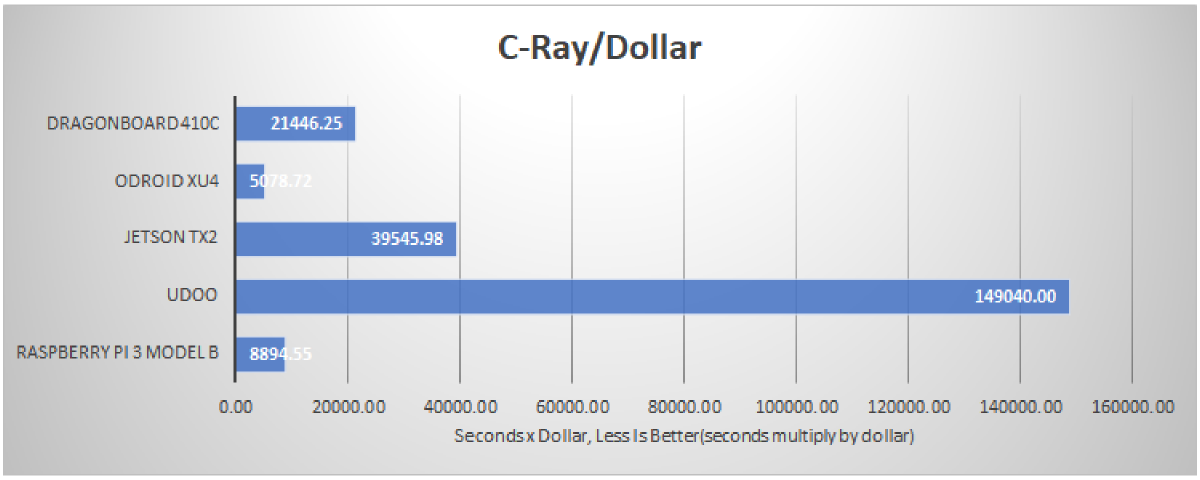


Figure 2.6: C-Ray/Cost benchmark.



Smallpt

The Odroid Xu4 scored the best it has the best result in for the value benchmark but the jetson has the best performance.

Figure 2.7: smallpt benchmark.

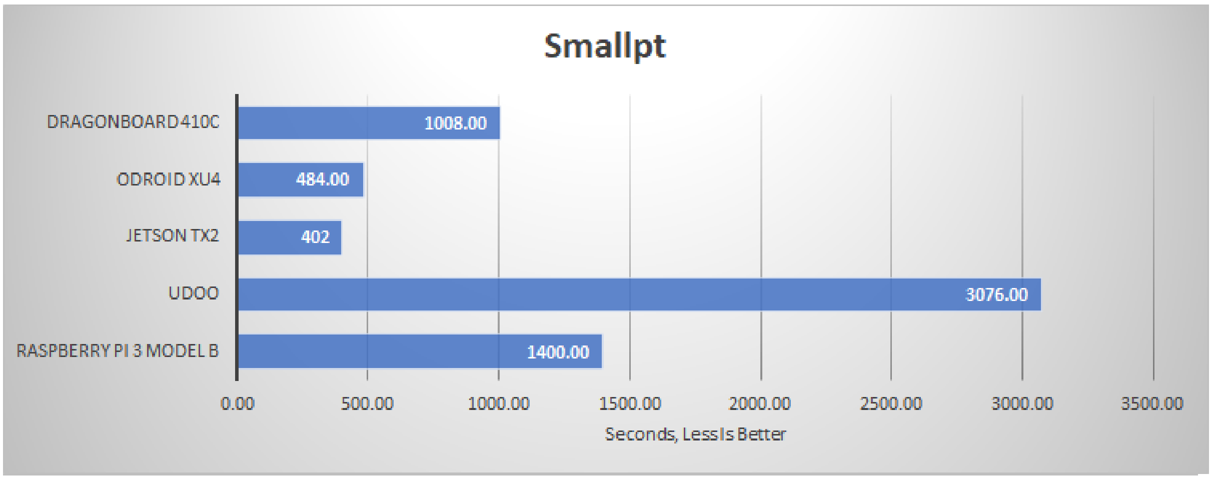
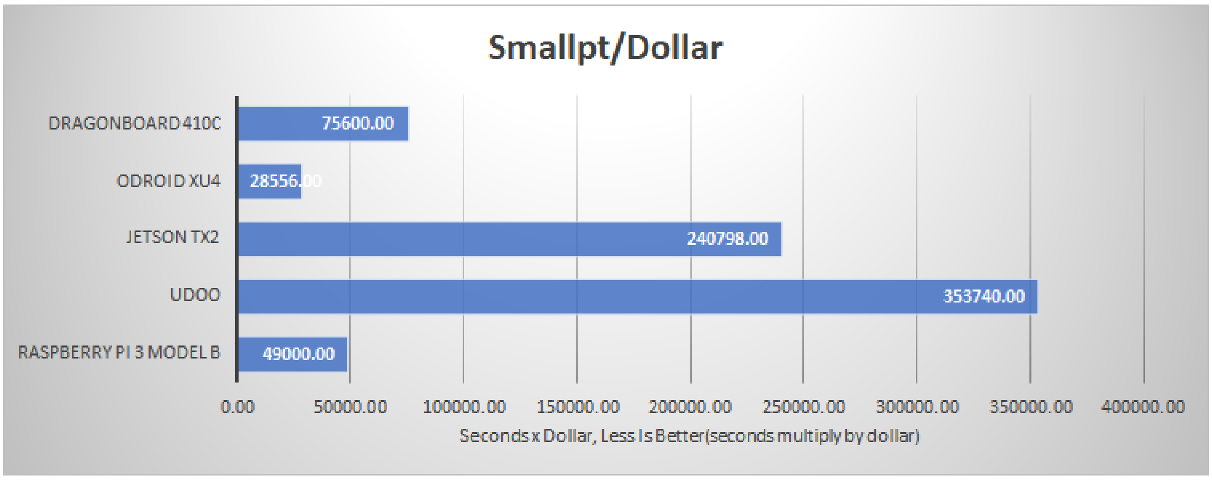


Figure 2.8: smallpt/cost benchmark.



FLAC Audio Encoding

Figure 2.9: flac audio benchmark.

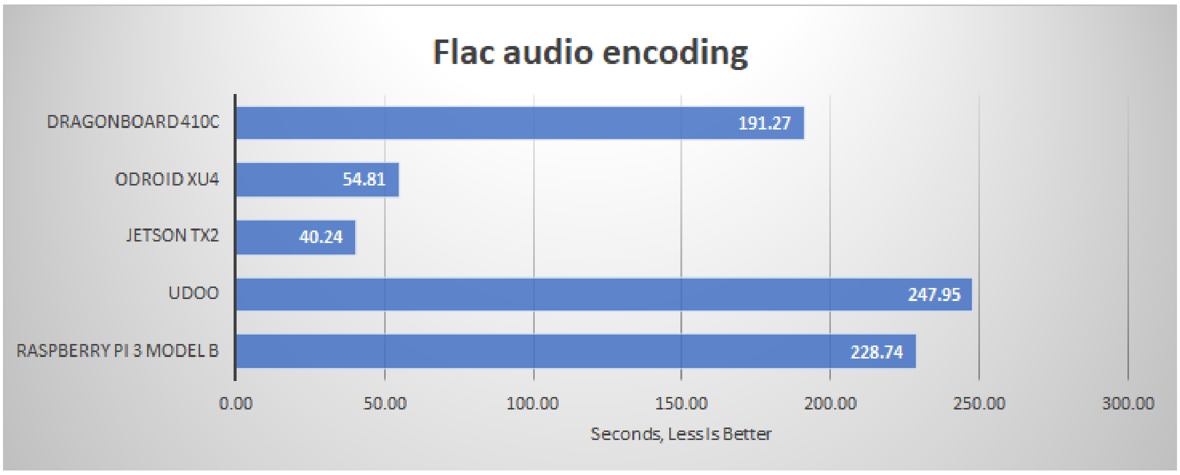
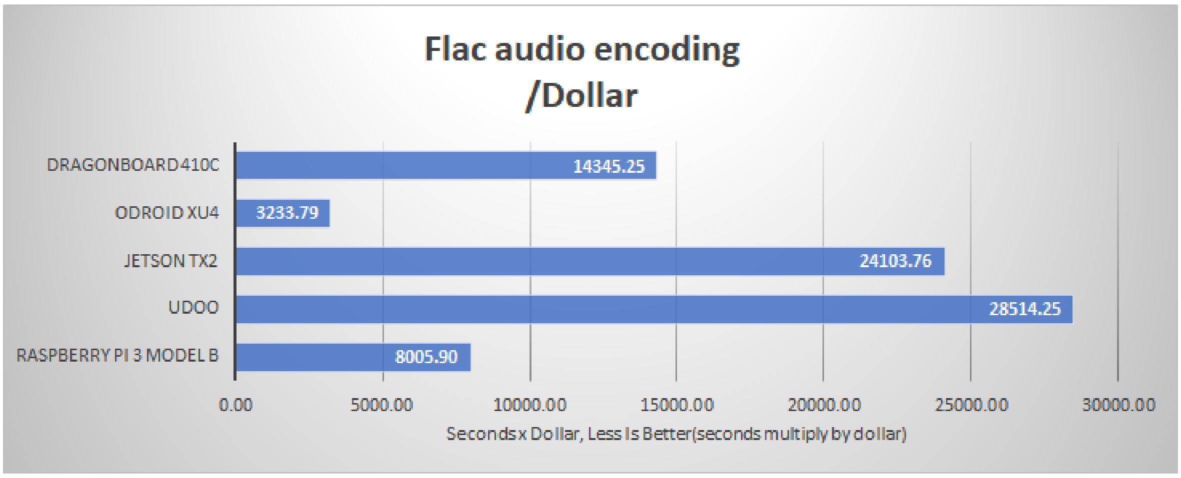


Figure 2.10: flac audio /cost benchmark.



Redis

Figure 2.11: Redis(Set) benchmark.

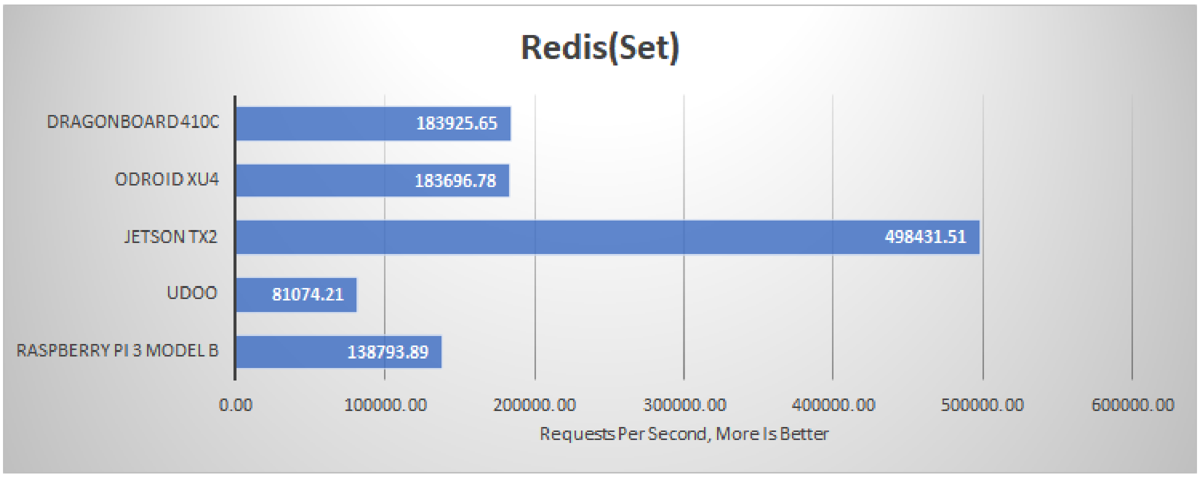


Figure 2.12: Redis(set)/cost benchmark.

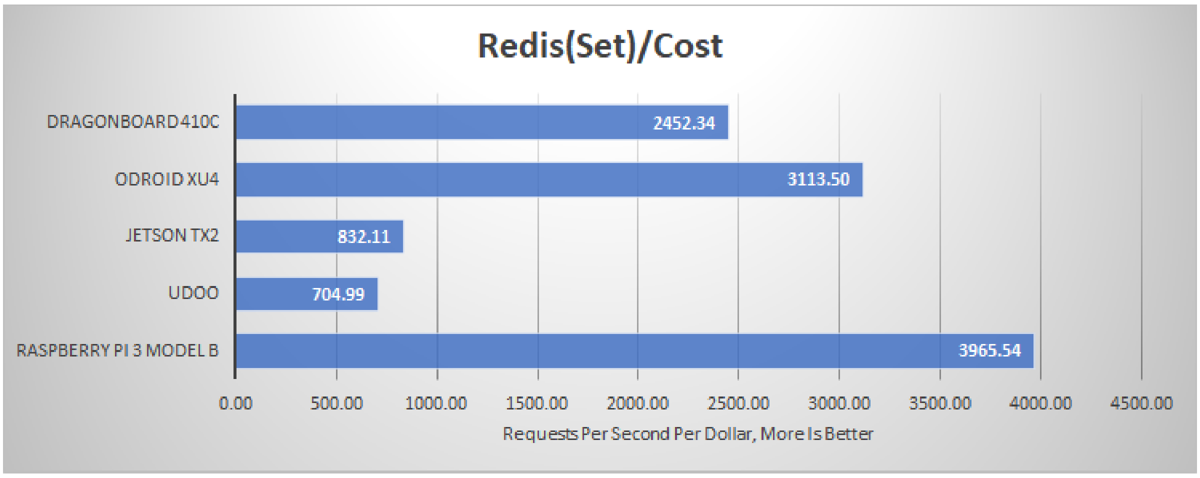


Figure 2.13: Redis(Get) benchmark.

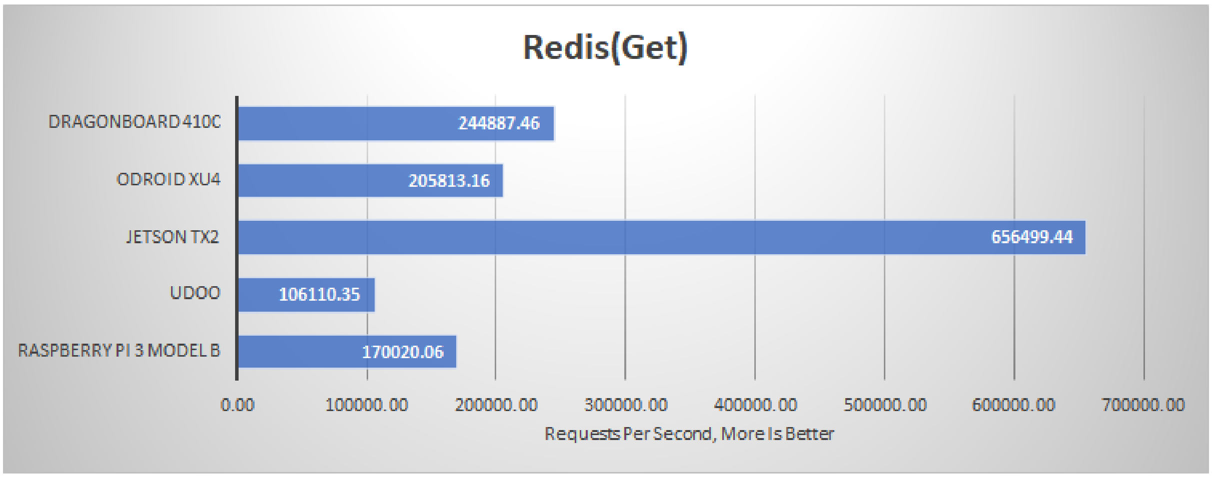
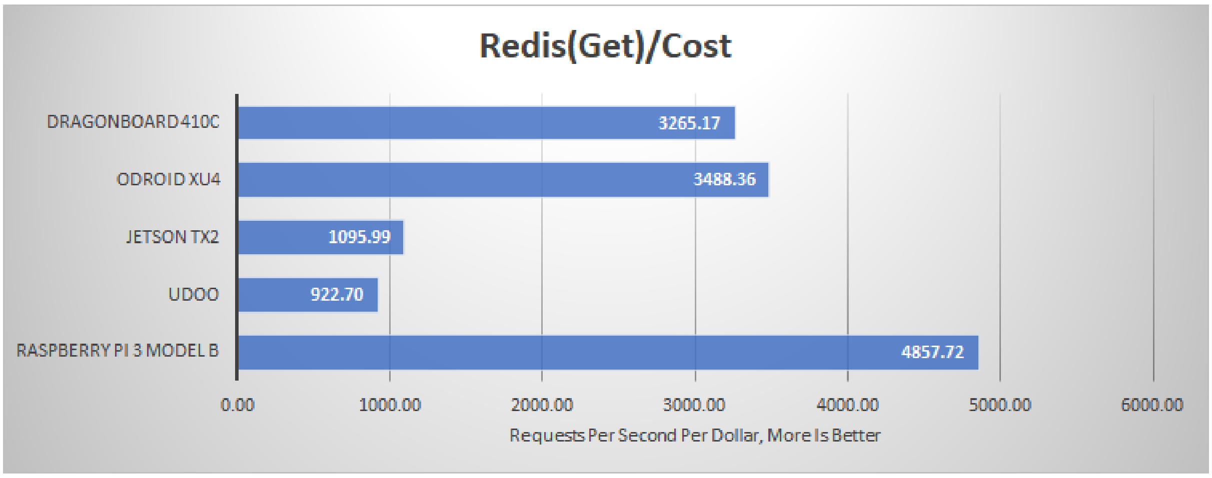


Figure 2.14: Redis(Get)/cost benchmark.



Himeno

Figure 2.15: Himeno benchmark.

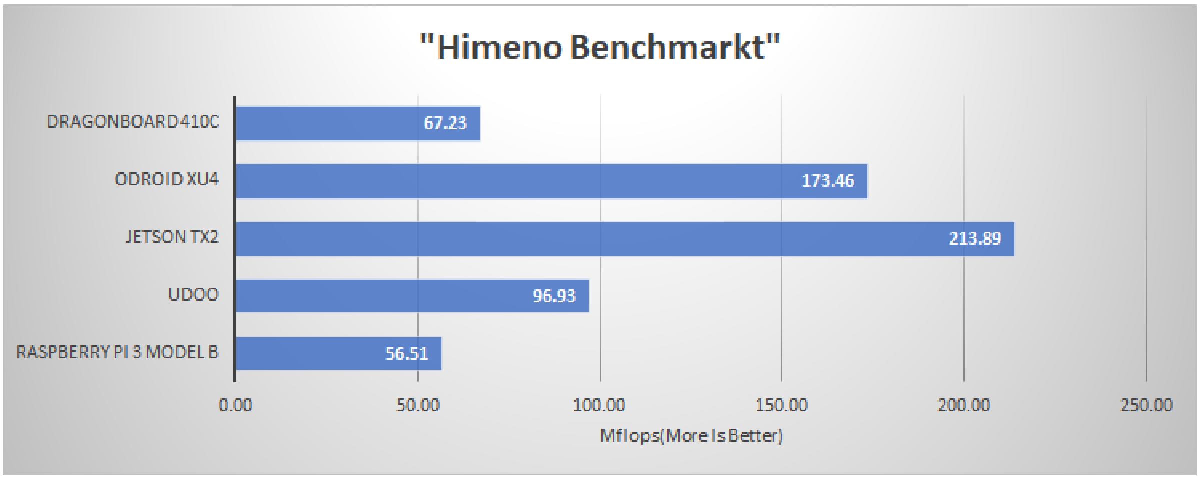
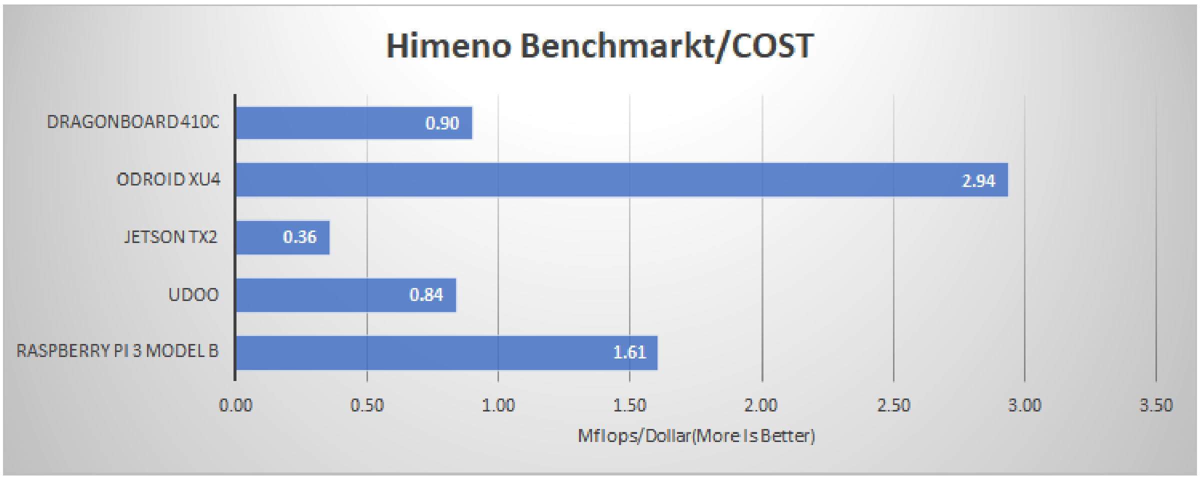


Figure 2.16: Himeno/cost benchmark.



conlusion

In all the processors benchmarks the best board is without surprise the Jetson TX2. The reason is because this board is more powerful it has 8Gb of rams compared to the Odroid Xu4 it has 4 times the amount of ram memory. The best price value board is the Odroid Xu4 for performance. Its scores much better than all the other board for the Performance benchmarks.

For the Redis benchmarks wich is a system benchmark. The Jetson still score the best but on the Rasperry Pi3 score the best in term of price value. All the result are available on following link : https://openbenchmarking.org/result/1805273-FO-1805279FO65

The user has do decides if he want to save money and work with a board like a Raspberry Pi 3 or get maximum performance and work with a jetson board.A good compromise is the

Odroid Xu4. This board looks powerful and is low cost.

About vision we can find a lot of benchmarks that compares algorithms to do some specific task like face-detection. There are also more general benchmarks that compares single boards computers (2.1), but there is no benchmark that will compare boards on how they are good to do real time image processing.

3 Pixy cam

The pixycacam is an vision sensor. This board can help an arduino or simal board to do fast image processing. the pixicam will process the image and send only useful information to the microcontroller. The frame rate is 50Hz and it can use several interfaces to send the inforamtion:UART serial, SPI, I2C, USB, or digital/analog output. In our use cas the pixi will do the iamge processing while the micronctorller(Arduino) can use his CPU for other tasks.[5]

Technical specs:

- Processor: NXP LPC4330, 204 MHz, dual core
- Image sensor: Omnivision OV9715, 1/4", 1280x800
- Lens field-of-view: 75 degrees horizontal, 47 degrees vertical
- Lens type: standard M12 (several different types available)
- Power consumption: 140 mA typical
- Power input: USB input (5V) or unregulated input (6V to 10V)
- RAM: 264K bytes
- Flash: 1M bytes
- Available data outputs: UART serial, SPI, I2C, USB, digital, analog
- Dimensions: 2.1" x 2.0" x 1.4
- Weight: 27 grams

3.1 Limit of Pixy

The Pixy can only be used for color detection not for object detection. The second problem is that the Pixy can only detect flashy colors like red yellow blue (figure: 3.1). If the object has any rebound color the Pixy will not detect the object (figure:3.2). Last point is that the pixy will not detect white colors (figure: 3.3).

Figure 3.1: Red color detection.

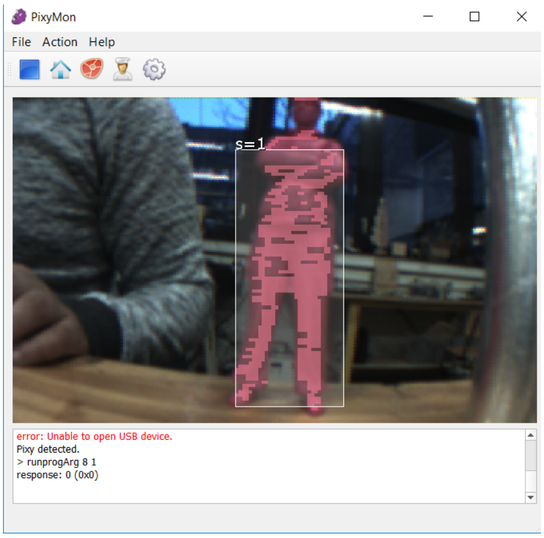


Figure 3.2: Rebound object detection.

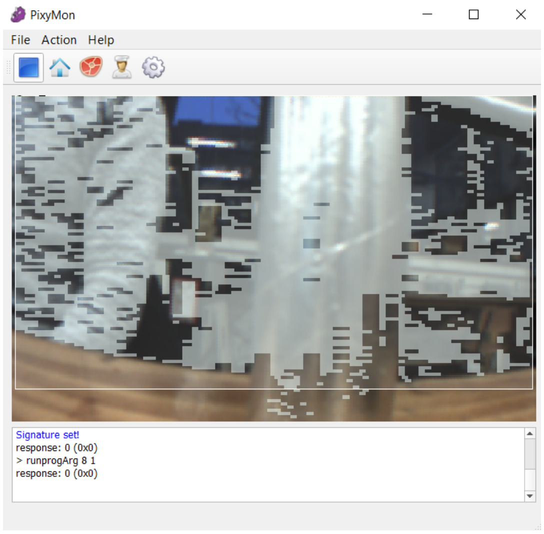
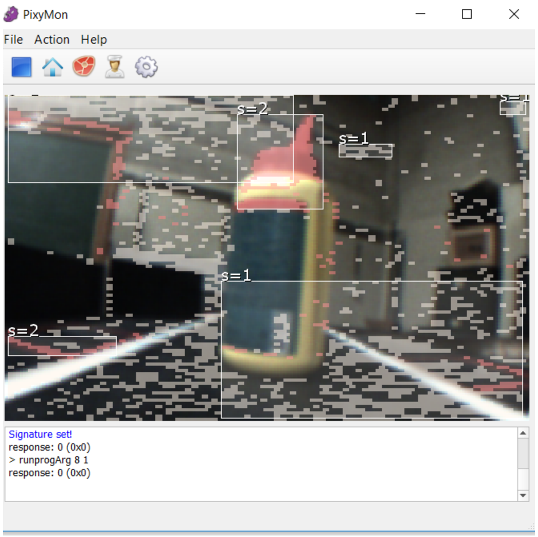


Figure 3.3: white color detection.



4 Comparison

4.1 board comparison

This thesis will compare 5 boards. The chosen boards all run an arm architecture. The boaérd are selected because of they popularity, price(depending on use case) and power consumption:

- **Raspberry Pi 3**: The most famous single board computer of our list. The cost of the board is around $35. It is possible to run several operating system on it like Linux or Windows 10 IOT core. [6]

- **Udoo**: The Udoo Dual is a computer that can be used with Android and Linux. It is a powerful prototyping board because it contains a processor and a Arduino Due microcontroller. To summerize the Udoo we can say that it bring the best of an Raspberry Pi and an Arduino togheter in-to one single board. **maksimovic2014raspberry** On the market we can find three type of Udoo that combine a processor and an Arduino. In this thesis we will discuss the Udoo dual. The dual is the mid-range Udoo and the cost of it is around $115 [7].

- **Odroid Xu4**: The Odroid X4 is and powerful Single board computer. It combine two processor a Samsung Exynos5422 Cortex A15 (2 GHz) and a Cortex A7 Octa core CPU (1.4 Ghz). This two processors are combined with 2 GB of RAM. **abdullah2016position** The board can run several flavors of Linux. The computing power of the X4 seems to be seven times faster than the Raspberry Pi3. The Xu4 can be found on the market at the price of $59. [8]

- **Dragonboard 410c**: The Dragonboard 410 c is a single board computer basd on the mid-tier Qualcomm® Snapdragon™ 410E processor. It can run Windows 10 IOT core and Linux(Andoid, Debian). The price is around $75. [9]

- **Nvidia jetston tx2**: Jetson is an NVIDIA® high performane computer. it is low-power and it is ideal for embedded compute-intensive projects like deep learning or computer vision. [10] This thesis will compare the Jetson TX2 Developer Kit with other boards. On the market the TX2 Devloper kit can bi found for $599.

  Table 4.1 compares the technical specs of all the cited boards.

	udoo dual	**raspberry pi 3**	**odroid Xu**	**Dragonboard 410c**	**NVIDIA jetston TX2 devlopper kit**	**Pixycam**
cpu	NXP i.MX 6 DualLite Atmel SAM3X8E (Arduino Due)	ARM Cortex-A53 Quad-core	Samsung Exynos5422 Cortex™A15 2Ghz + Cortex™-A7 Octa core CPUs	ARM Cortex-A53 Quad-core	HMP Dual Denver 2/2MB L2 + Quad ARM A57/2MB L2	NXP LPC4330
# cores	2	4	8	4	6	2
clk(GHz)	1	1.2	2	1.2	2	0.204
ram(GB)	1	1	2	1	8	0.000264
memory(GB)	unlimitted (sd card)	unlimitted (sd card)	unlimitted (sd card)	unlimitted (sd card)	32	1M
ipc(instruction per cycles)	1	1	1	1	1	1
GFLOPs	2	4.8	16	4.8	12	0.408
Power supply (DC V)	6	5	5	6.5	19	
GPU	yes Vivante GC 880 + Vivante GC 320	yes broadcom videocore 4	yes mali T628	yes Qualcomm Ardeno 306	yes NVIDIA Pascal, 256 NVIDIA CUDA Cores	no
opencl/cuda	maybe	no	yes	yes	yes(Cuda)	
core clk (mhz)	400	250	600	450		
pixel/rate(Mpixel/sec)	80	1000				
operating System	Linux debian	linux windows10 debian	linux debian	linux windows debian	linux debian	no
camera plug	yes	yes	yes	yes	yes	
python /opencv	yes	yes	yes	yes	yes	
price(€)	110	40	75	75	599	
arduino compatible (connector for Arduino)	yes	no	no	yes	no	yes

Table 4.1: Comparison board table [11] [12] [13] [10]

5 Generic vision Benchmark

facedetection, colordetection, openCl, edgedetection, convolution, matrixmultiplucation

6 Results

7 Conclusion

Bibliography

[1] NickPowers, *Benchmarkingpopularsingleboardcomputers—arrow.com*, https://www.arrow.com/en/research-and-events/articles/benchmarking-popular-single-board-computers, (Accessedon05/09/2018), Feb. 2017 (cit. on pp. 11, 31, 32).

[2] MichaelLarabelandMatthewTippett, *Phoronixtestsuitev7.8.0*, https://www.phoronix-test-suite.com/documentation/phoronix-test-suite.pdf, (Accessedon05/14/2018), (cit. on p. 11).

[3] MichaelLarabel, *Benchmarksofmanyarmboardsfromtheraspberrypitonvidiajetsontx2-phoronix*, https://www.phoronix.com/scan.php?page=article&item=march-2017-arm&num=5, (Accessedon05/14/2018), Mar. 2017 (cit. on p. 11).

[4] *Openbenchmarking.org - c-ray test profile*, https://openbenchmarking.org/test/pts/c-ray-1.1.0, (Accessed on 06/04/2018), (cit. on p. 13).

[5] *Overview-CMUcam5Pixy-CMUcam:OpenSourceProgrammableEmbeddedColorVisionSensors*. [Online]. Available: http://cmucam.org/projects/cmucam5 (visited on 05/01/2018) (cit. on p. 21).

[6] *Raspberrypifaqs-frequentlyaskedquestions*, https://www.raspberrypi.org/help/faqs/, (Accessedon05/02/2018), (cit. on p. 24).

[7] *Dualquad-shop–udoo*, https://shop.udoo.org/eu/quad-dual.html?___from_store=other&popup=no, (Accessedon05/02/2018), (cit. on p. 24).

[8] *Odroid—hardkernel*, http://www.hardkernel.com/main/products/prdt_info.php, (Accessedon05/02/2018), (cit. on p. 24).

[9] *Dragonboard410cbyarrowdevelopmenttools—embeddedsystemdevelopmentboardsandkits—arrow.co* https://www.arrow.com/en/products/dragonboard410c/arrow-development-tools, (Accessedon05/02/2018), (cit. on p. 24).

[10] *Jetsonfaq—nvidiadeveloper*, https://developer.nvidia.com/embedded/faq, (Accessedon05/04/2018), (cit. on pp. 24, 25).

[11] *Samsungexynos5420octasoc-notebookcheck.nettech*, https://www.notebookcheck.net/Samsung-Exynos-5420-Octa-SoC.103633.0.html, (Accessedon05/04/2018), (cit. on p. 25).

[12] *Videocore®iv3darchitecturereferenceguide*, https://docs.broadcom.com/docs/12358545, (Accessedon05/04/2018), 5300CaliforniaAvenue•Irvine,CA92617: BROADCOM, (cit. on p. 25).

[13] *I.mx6solo/6dualliteapplicationsprocessorsforindustrialproductsdatasheet*, https://www.nxp.com/docs/en/data-sheet/IMX6SDLIEC.pdf, (Accessedon05/04/2018), (cit. on p. 25).

8 Appendix

.1 Appendix 1 : Result benchmark [1]

4 CPUs/1 Parallel Process

Benchmark	Units	BBB	DragonBoard	Jetson	Joule	Raspberry Pi 3
Dhrystone 2 using register variables	lps	3675629.7	6412076.1	16062505.4	20379093.6	4409370.5
Dhrystone 2 using register variables	DMIPS	2092.0	3649.4	9142.0	11598.8	2509.6
Double-Precision Whetstone	MWIPS	191.3	726.3	1939.8	3366.4	765.9
Execl Throughput	lps	420	584.3	1708.2	1012.1	547.4
File Copy 1024 bufsize 2000 maxblocks	KBps	67210.2	175556.2	178734.4	536476.1	147852
File Copy 256 bufsize 500 maxblocks	KBps	20736.3	52613.5	50675.1	175890.5	44358
File Copy 4096 bufsize 8000 maxblocks	KBps	154563.2	444977.6	560630.5	1136087.5	363495
Pipe Throughput	lps	159914.8	448412.7	326667.6	1221011.2	313126.2
Pipe-based Context Switching	lps	24435.6	39752.1	70472.7	56665.2	55660.8
Process Creation	lps	1081.5	1733.9	4638.9	2171.4	2533.6
Shell Scripts (1 concurrent)	lpm	737.4	1381.3	2195.5	4384.9	2171.9
Shell Scripts (8 concurrent)	lpm	94.8	712.4	1264.7	1856.7	590.7
System Call Overhead	lps	348986.1	851850	777035.1	1963668.6	699242.9
System Benchmarks Index Score:		131.2	320	500.7	802.4	306.8

.2 Appendix 1 : Result benchmark [1]

4 CPUs/4 Parallel Process

Benchmark	Units	BBB	DragonBoard	Jetson	Joule	Raspberry Pi 3
Dhrystone 2 using register variables	lps	N/A	18546851.3	65127838.2	66693865.6	17635586.6
Dhrystone 2 using register variables	DMIPS	N/A	10556.0	37067.6	37958.9	10037.3
Double-Precision Whetstone	MWIPS	N/A	2125.1	7802.5	11602.4	3063.1
Execl Throughput	lps	N/A	2984.7	6243.4	8878.1	2277.1
File Copy 1024 bufsize 2000 maxblocks	KBps	N/A	200079.7	254376.7	629679.3	246720.9
File Copy 256 bufsize 500 maxblocks	KBps	N/A	54394.5	75190.3	184008.3	64040.1
File Copy 4096 bufsize 8000 maxblocks	KBps	N/A	560188.4	726453.7	1503381.1	579721.3
Pipe Throughput	lps	N/A	1216158.7	790668.8	4066808.5	1249299.5
Pipe-based Context Switching	lps	N/A	174856.2	250735.2	631020.7	210350.8
Process Creation	lps	N/A	7602.8	8602.8	27367.6	5097.1
Shell Scripts (1 concurrent)	lpm	N/A	4564.4	10213	16146.5	4954.9
Shell Scripts (8 concurrent)	lpm	N/A	588.1	1305.8	2038.1	639.2
System Call Overhead	lps	N/A	2321293.3	2868714.1	5160156	2689482.6
System Benchmarks Index Score:		N/A	744	1225.4	2490.2	780.6