

Análise do Desempenho Computacional da Raspberry Pi Executando o Benchmark SysBench

Gabriel de Oliveira Silva¹, Wanderson Roger Azevedo Dias², Danilo Pereira Escudero¹

¹Cooordenadoria do Curso Técnico em Informática (CCTI)

²Cooordenadoria do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas (CCSTADS)

Laboratório de Arquiteturas Computacionais e Computação Paralela (LACCP)

Instituto Federal de Rondônia (IFRO)

Ji-Paraná – RO – Brasil

{gabriel.oliveira.s177, wradias, danilopescudero}@gmail.com

Abstract. *The Raspberry Pi (RPi) platform, which contains a microcomputer, was developed by the Raspberry Pi Foundation and chosen to be used in this project because it is a low-cost platform. Thus, this paper has as main objective to present the results of the executions that measured the computational power (performance) of the RPi platform models 2B, 3B and 4B, through a comparative analysis of the computational performance using the SysBench benchmark running in the distribution of the Raspbian Linux Operating System. Analyzing the CPU metric of the SysBench benchmark, in the experiments it was found that the RPi model 4B was 2.5x and 2x faster than the models 2B and 3B (respectively), running with 1 thread and with a workload that performs the calculation of 2 million prime numbers.*

Resumo. *A plataforma Raspberry Pi (RPi), que contém um microcomputador, foi desenvolvida pela Fundação Raspberry Pi e escolhida para ser usada nesse projeto por se tratar de uma plataforma de baixo custo financeiro. Assim, este artigo tem como principal objetivo apresentar os resultados das execuções que mensuraram o poder computacional (performance) da plataforma RPi modelos 2B, 3B e 4B, por meio de uma análise comparativa do desempenho computacional realizada com o uso do benchmark SysBench executando na distribuição do Sistema Operacional Linux Raspbian. Analisando a métrica CPU do benchmark SysBench, nos experimentos constatou-se que a RPi modelo 4B, foi 2,5x e 2x mais rápida que os modelos 2B e 3B (respectivamente), executando com 1 thread e com carga de trabalho que realiza o cálculo de 2 milhões de números primos.*

1. Introdução

Observando o declínio gradual no interesse dos estudantes pela área da Computação no Reino Unido, em 2006, *Eben Upton, Rob Mullins, Jack Lang, Alan Mycroft, Pete Lomas e David Braben* (idealizadores do projeto), professores e pesquisadores na Universidade de Cambridge, Inglaterra, resolveram propor e criar uma plataforma computacional de pequeno porte físico e de baixo custo financeiro, onde o maior objetivo na criação era promover uma aproximação das crianças e jovens para as áreas de Ciência da Computação, Engenharia da Computação e outras áreas afins, além de despertar a curiosidade e habilidades em programação [Richardson & Wallace, 2013].

Eben Upton, diretor de estudos em Ciência da Computação na Universidade, diagnosticou que os estudantes que se candidatavam a participar da disciplina de Laboratório de Computação, não apresentavam as mesmas habilidades e domínio das máquinas que tinham os estudantes da década de 1990. Sendo que, naquela época, jovens de aproximadamente 17 anos que desejavam cursar a disciplina já chegavam à faculdade com certo conhecimento de linguagens de programação e do funcionamento do hardware, além de outras habilidades.

Então, o primeiro protótipo do pequeno computador (ver Figura 1) surgiu na mesa da cozinha da casa de *Eben*. Ele e seus amigos começaram a soldar, em uma *proto-board* com um *chip* Atmel ATmega644, alguns outros *chips* baratos de microcontroladores para monitorar um aparelho de TV. O projeto contava com apenas 512K de memória RAM, atingindo poucos MIPS (*Millions of Instructions*

Per Second) de processamento. O grande desafio era tornar esse pequeno computador atrativo aos olhos das crianças e jovens, já acostumadas, na época, com diversas outras tecnologias sofisticadas.

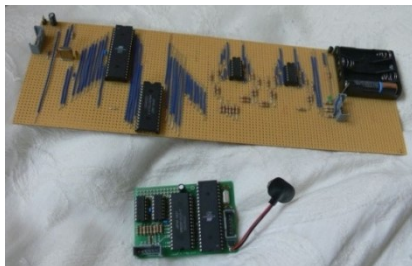


Figura 1. Protótipo da Raspberry Pi

Posteriormente, após discussões sobre o projeto e o estado geral do ensino da Computação, *Eben, Rob, Jack, Alan, Pete e David* resolveram criar a *Raspberry Pi Foundation* (<https://www.raspberrypi.org/>), que rapidamente passou a contar com vários outros colaboradores e voluntários. Assim, a fundação criou a Raspberry Pi (RPi) (ver Figura 2), que é uma plataforma que está em ascensão no mercado mundial, conforme [Upton & Halfacree, 2013], ultrapassando a impressionante quantidade de 46 milhões de unidades vendidas (*circa* fevereiro de 2022) ao redor do mundo, segundo [Martins, 2022]. De acordo com os seus idealizadores a proposta inicial de criação da RPi era somente produzir e comercializar algumas centenas de placas da plataforma, sendo apenas para auxiliar estudantes de Computação do Reino Unido que não possuíam condição financeira o suficiente para adquirir um computador. No entanto, além do seu baixo custo, ou seja, em torno de US\$ 35 [FRpi, 2022] para a maioria dos seus modelos, outras características da RPi se destacam, sendo: (i) poder de processamento, podendo assim executar diversas aplicações que são destinadas especificamente para *desktops*, servidores e outras plataformas computacionais, e (ii) o baixo consumo de energia elétrica.

A maioria dos modelos existentes da plataforma RPi (A, B, B+, 2B, 3B, 3B+, 4B e outros), tem dimensões próximas a de um cartão de crédito (8,5 x 5,6 x 1,7 cm, sendo: comprimento, largura e altura respectivamente), segundo [Merces, 2013], já os modelos Zero, Zero W, Zero 2W (ver Figura 3), com início de lançamento deste modelo em novembro de 2015, tem 6,5 x 3,0 x 0,5 (comprimento, largura e altura respectivamente) de tamanho e com custo de apenas US\$ 5 [FRpi, 2022]. Entretanto, para o funcionamento da RPi é necessário o uso de um sistema operacional embarcado o qual é responsável por todo o gerenciamento da plataforma e de seus recursos, sendo o Linux, o sistema operacional oficial adotado pelos criadores da plataforma RPi.



Figura 2. Raspberry Pi modelo 4B



Figura 3. Raspberry Pi modelo Zero

No entanto, devido ao crescente uso e popularização dessa plataforma, surgiram diversas outras distribuições de Sistemas Operacionais Linux específicos para o gerenciamento da RPi (<https://www.raspberrypi.com/software/>) [Lima *et al.*, 2015]. De acordo com especificações fornecidas pela Fundação Raspberry Pi, nos modelos a partir da 2B, também é possível executar uma versão específica do sistema operacional Microsoft Windows, denominada de **Windows 10 IoT Core Insider Preview**, lançado em 2015 pela Microsoft para ser usado em dispositivos como Raspberry Pi, Arduino e o *Minnow Board*, conforme divulgações feitas pela própria Microsoft [Microsoft, 2016].

A Fundação Raspberry Pi, responsável pela criação da plataforma RPi, não visa lucros e por esse motivo vem mantendo uma política de preço igualitário quando lançados novos modelos da plataforma, mesmo sempre havendo melhorias de hardware na nova versão, contribuindo assim, cada vez mais para o crescimento de suas vendas em torno do mundo, conforme [FRpi, 2022], além de suas características atraentes de custo/benefício, poder de processamento, aplicabilidade, facilidade de uso

e popularização. Portanto, um dos principais objetivos dos criadores da RPi é incentivar a pesquisa, o aprendizado das linguagens de programação e principalmente a aproximação de pessoas sem poder aquisitivo com as tecnologias, em especial o computador.

Então, mediante ao que já foi citado anteriormente, ainda surgem alguns questionamentos que incentivam a realização de uma pesquisa mais detalhada a respeito da plataforma RPi, sendo: (i) será que realmente é possível obter desempenho computacional usando plataformas Raspberry Pi? (ii) qual a diferença do poder computacional entre os diversos modelos da plataforma RPi? (iii) qual o impacto que as diversas configurações de parâmetros de um *benchmark* podem influenciar nos modelos das plataformas Raspberry Pi?

Assim, sabe-se que no decorrer do tempo, a plataforma RPi desde sua criação e lançamento (29 de fevereiro de 2012) vem ganhando cada vez mais espaço no mundo computacional [Merces, 2013]. Portanto, por esses e outros motivos, foi escolhida esta plataforma como objeto central de pesquisa neste artigo e seu foco é a realização de execuções do *benchmark SysBench*, com diferentes configurações de parâmetros e carga de trabalho, no qual será possível extrair dados que permitem realizar uma análise comparativa do desempenho computacional dos diversos modelos da plataforma, mais específico os modelos 2B, 3B e 4B (2GB de RAM) por já possuímos, executando com o sistema operacional Linux *Raspbian* (Raspberry Pi OS).

O restante do artigo está organizado da seguinte forma: a Seção 2 apresenta uma breve contextualização, a fim, de ambientar o estudo corrente, nesta seção também são apresentados alguns trabalhos correlatos desenvolvidos recentemente e que usaram a plataforma RPi como dispositivo computacional; A Seção 3 apresenta as análises do desempenho computacional das RPi's executando o *benchmark SysBench* e a Seção 4 finaliza com as conclusões e ideias para trabalhos futuros.

2. Contextualização

2.1. Plataforma Raspberry Pi

Raspberry Pi é um computador do tamanho de um cartão de crédito, que se conecta a um monitor e usa um teclado e um mouse padrão. Foi desenvolvido no Reino Unido pela Fundação Raspberry Pi. É capaz de fazer tudo que se esperaria de um computador *desktop* como navegar na internet; reproduzir vídeo; manipular planilhas; processar textos; compilar e executar programas; e jogar jogos. Além disso, Raspberry Pi pode interagir com o mundo exterior através da conexão GPIO (*General Purpose Input/Output*) e tem sido usado numa ampla gama de projetos digitais como: Sistema Embarcado, Internet das Coisas (IoT) e outros.

Conforme já mencionado anteriormente e constatado no *site* oficial da Fundação, a plataforma RPi atualmente (julho de 2022) está disponível em diversos modelos, sendo eles: 1A+, 1B+, 3A+, 3B, 3B+, 4B, Pi 400, Zero, Zero W, Zero 2W, conforme [FRpi, 2022]. A diferença existente entre os modelos, está em sua capacidade computacional (quantidade de memória interna, de portas USB, Ethernet, tipos de processador *single-core* ou *multi-core*, velocidade de *clock*, *bluetooth*, *wireless* entre outras), ainda assim, todos os modelos são baseados em arquitetura SoC (*System on Chip*) de 32 ou 64 *bits*, exemplos: o modelo 2B, o processador é um ARM (*Advanced RISC Machine*) Cortex-A7 de 900MHz que suporta instruções de 32 *bits*, usando o conjunto de instruções ARMv7 [Furber 2000; Seal, 2001], já o modelo mais atual da RPi, ou seja, o 4B é composta pelo processador *quad-core* ARM Cortex-A72 de 1.5GHz, podendo ter 2, 4 ou 8GB de memória RAM e suporta dois monitores com resolução 4K, já especificamente o modelo Pi 400, a placa do SoC está inserida dentro de um teclado (ver Figura 4). A Tabela 1 apresenta as principais características inerentes aos diversos modelos de RPi.



Figura 4. Protótipo da Raspberry Pi

Tabela 1. Características de Alguns Modelos da RPi

Características	Modelos de Raspberry Pi					
	1B	2B	3B	4B	Pi 400	Zero 2W
Lançamento	15/02/2012	01/02/2015	29/02/2016	24/06/2019	02/11/2020	28/10/2021
SoC	BCM2835	BCM2836	BCM2837	BCM2711	BCM2711	BCM2710
CPU	ARM11 @ 700MHz	Cortex-A7 @ 900MHz	Cortex-A53 @ 1.2GHz	Cortex-A72 @ 1.5GHz	Cortex-A72 @ 1.8GHz	Cortex-A53 @ 1.0GHz
Nº de Cores	1	4	4	4	4	4
ISA	ARMv6 32 bits	ARMv7 32 bits	ARMv8 64 bits	ARMv8 64 bits	ARMv8 64 bits	ARMv8 64 bits
GPU	VideoCore IV @ 250MHz	VideoCore IV @ 250MHz	VideoCore IV @ 400MHz	VideoCore VI @ 500MHz 4K (2 portas)	VideoCore VI @ 500MHz 4K (2 portas)	VideoCore IV @ 400MHz
RAM	256MB SDRAM	1GB SDRAM	1GB SDRAM	2GB, 4GB ou 8GB SDRAM	4GB SDRAM	512MB SDRAM
Armazenamento	SD	microSD	microSD	microSD	microSD	microSD
Ethernet	-----	10/100 Mbit/s	10/100 Mbit/s	1 Gbit/s	1 Gbit/s	-----
USB	-----	4 portas	4 portas	4 portas	4 portas	1 porta micro
Wireless	-----	-----	802.11n	802.11ac	802.11ac	802.11n
Bluetooth	-----	-----	4.1 (BLE)	5.0 (BLE)	5.0 (BLE)	4.2 (BLE)
Saída de Vídeo	RC Composto e HDMI	HDMI	HDMI	micro HDMI	micro HDMI	mini HDMI
Saída de Áudio	áudio jack 3.5mm e HDMI	áudio jack 3.5mm e HDMI	áudio jack 3.5mm e HDMI	áudio jack 3.5mm e HDMI	micro HDMI	mini HDMI
GPIO	26 pinos	40 pinos	40 pinos	40 pinos	40 pinos embutidos	40 pinos embutidos
Alimentação	5V - 500mA	5V - 2A	5V - 2A	5.1V - 3A	5.1V - 3A	5V - 2A
Preço	US\$ 25	US\$ 35	US\$ 35	US\$ 35 a 55	US\$ 99	US\$ 15

2.2. Sistemas Operacionais para Raspberry Pi

Atualmente, existem diversas distribuições de Sistemas Operacionais (SO) que podem ser usadas na plataforma RPi. Assim, nesta seção é apresentado de forma resumida as principais características de duas distribuições de SOs Linux, utilizadas em RPi, sendo elas:

- **Raspbian:** é uma derivação do sistema operacional Linux distribuição *Debian*, que foi projetado e compilado para a plataforma Raspberry Pi, atualmente esta distribuição é chamada de Raspberry Pi OS e é a distribuição oficial da Fundação [FRpi, 2022]. O *Raspbian* oferece mais de 35.000 pacotes *deb* de software, que estão pré-compilados para serem facilmente instalados nesta distribuição de sistema operacional. O *Raspbian* também é otimizado para obter o desempenho máximo da RPi. Então, o *Raspbian* basicamente contém o ambiente de *desktop* LXDE (*Lightweight X11 Desktop Environment*), o gerenciador de janelas *OpenBox*, o navegador *Midori*, ferramentas para desenvolvimento de software e código fonte de exemplo de funções multimídia;
- **Ubuntu Mate:** é uma distribuição baseada no tradicional *Ubuntu*, que vem com o ambiente gráfico *Mate* como *desktop* padrão [FRpi, 2022]. O *Mate* é um *fork* do GNOME 2 que permite o uso da antiga interface e aplicativos desse ambiente (*Caja*, *Pluma*, *Eye of Mate*, *Atril*, *Engrampa* e outros) [Mate, 2022] e o uso das novas aplicações. Além disso, também pode ser instalado em paralelo com o GNOME 3. Assim, esta distribuição possui uma área de trabalho baseada na tradicional metáfora do *desktop*, com uma barra de tarefa, menu de aplicativos e outros itens comuns dessa abordagem. Um dos maiores diferenciais do *Ubuntu Mate* é seu baixo consumo de recursos. Isso o torna adequado tanto para *desktops* e *laptops* modernos, como para microcomputadores e hardware mais antigo.

Neste artigo, optamos pela distribuição do sistema operacional *Raspbian* por ser a distribuição oficial da Fundação.

2.3. Benchmark SysBench

SysBench é uma ferramenta de *benchmark* modular, multiplataforma e *multithread*, usada para avaliar o desempenho computacional entre diferentes modelos de arquiteturas computacionais. O objetivo desse *benchmark* é obter rapidamente métricas sobre o desempenho do sistema computacional executando uma carga de trabalho.

O *SysBench* executa um número específico de *threads* e todas elas executam solicitações em paralelo. A real carga de trabalho produzida por solicitações depende do modo de teste especificado. É

possível limitar o total número de solicitações ou o tempo total para execução do *benchmark*, ou ambos. Os modos de teste disponíveis são implementados por módulos compilados e o *SysBench* foi projetado para tornar a adição de novos modos de teste uma tarefa fácil. Cada modo de teste pode ter opções adicionais (ou específicas da carga de trabalho). Os atuais recursos de análise do *SysBench*, permitem testar os seguintes parâmetros do sistema:

- Desempenho dos arquivos de E/S;
- Desempenho do agendador;
- Alocação de memória e velocidade de transferência;
- Desempenho de implementação de *threads* POSIX;
- Desempenho do servidor de banco de dados.

Então, neste projeto é utilizado o *SysBench* analisando diferentes cargas de trabalho para os recursos de “CPU”, quando executados nas plataformas RPi modelos 2B, 3B e 4B.

2.4. Trabalhos Correlatos

A Raspberry Pi, é uma plataforma interessante para ser usada em diversos tipos de projetos computacionais, sendo que a mesma se destaca pelo seu baixo custo e seu significativo poder de processamento comparado ao seu tamanho. Assim, a RPi apresenta um desempenho considerável em diversas tarefas executadas.

O trabalho de [Dinndorf, 2014] realiza a comparação do desempenho computacional de dois sistemas operacionais projetados para a arquitetura ARM, utilizando uma Raspberry Pi modelo B, sendo as distribuições Linux *Raspbian* e *Pidora*. Os testes realizados avaliaram o gerenciamento dos processos, memória e da memória secundária (armazenamento). Para isso, o autor utilizou o conjunto de *benchmarks* do pacote *Phoronix Test Suite*. O autor concluiu que um sistema operacional não supera claramente o outro em todos os aspectos de gerenciamento.

Já em outro projeto, Morabito (2016), realizou testes com *benchmarks* executando em uma Raspberry Pi modelo 2B usando a distribuição do sistema operacional *Raspbian* na versão *Jessie*. Os testes buscavam comparar a performance do processador, da memória e do armazenamento com e sem a virtualização. A virtualização aplicada foi a baseada em *containers*, e para isso, utilizaram o *docker* para a criação e administração destes ambientes. Os resultados mostram um impacto quase insignificante na camada de virtualização de *containers* em termos de desempenho, quando comparado à execução nativa.

Os autores Schot *et al.* (2016), desenvolveram uma solução de baixo custo financeiro e de consumo energético para atuar como sendo um micro *Datacenter*, que opera *Big Data* e *Streaming* audiovisual. Para esta solução, foram utilizados oito Raspberry Pi modelo 2B operando como um *Cluster*. A análise levou em consideração os resultados de diversos *benchmarks* que mensuraram o consumo de energia, a temperatura, armazenamento, memória e rede.

Semelhantemente, [Mappuji *et al.*, 2016] realiza a montagem de um *Cluster* usando oito Raspberry Pi modelo 2B. Os autores relatam que a Computação de Alto Desempenho (HPC) está se popularizando, mas ainda existe a barreira de preços, tanto de aquisição quanto de consumo energético. A solução encontrada foi utilizar a plataforma Raspberry Pi para a montagem do *Cluster* que trabalha com computação paralela e ao mesmo tempo, exige pouca demanda energética para o seu funcionamento. Foi utilizado o *benchmark High Performance Linpack* (HPL) para mensurar o desempenho computacional. Os experimentos mostraram que o aumento do número de nós no *Cluster*, não obtém um aumento significativo no poder de processamento. Sendo assim, os autores recomendaram o uso de até 4 nós, para uma relação aceitável entre desempenho e consumo energético.

Em [Kennedy, 2016], o autor simulou e analisou, mediante ao uso de diversos *benchmarks*, o desempenho de várias arquiteturas computacionais em relação a plataforma Raspberry Pi modelo B instalado o sistema operacional Linux distribuição *Raspbian*. O autor destacou que por meio das execuções realizadas foi possível verificar que o desempenho da CPU ARM1176JZF-S contida na RPi modelo B, se comportou de forma semelhante, para algumas tarefas, com o processador Intel Atom S1260. Portanto, o autor conclui que a RPi apesar de ser uma plataforma que não visa o alto desempenho, nas execuções comprovou que consegue alcançar desempenho semelhante a de outras plataformas que foram projetadas para este fim.

3. Resultados e Discussão

Foram utilizados três cartões de memória microSD Ultra de 16GB e Classe 10 (que servem como disco rígido da RPi) idênticos, posteriormente foi instalada e configurada a distribuição do sistema operacional *Raspbian*, além do *benchmark SysBench*. Cada uma das Raspberry Pi usadas nesta pesquisa (2B, 3B e 4B), recebeu um dos cartões de memória.

Para os testes e avaliações, foi analisado especificamente a métrica de CPU (conforme detalhado abaixo) do *benchmark SysBench* quando executado com diferentes cargas de trabalho e em modelos distintos de RPi's.

- **CPU:** cada solicitação consiste no cálculo de números primos até um valor especificado no parâmetro `--cpu-max-primos`, sendo que neste projeto foram definidos valores que variaram no intervalo de 50 mil até 2 milhões. Todos os cálculos de CPU são executados usando números inteiros de 64 *bits*. Também foi configurado para a execução do *SysBench* o uso de 1, 2, 3 ou 4 *threads*, assim, cada *thread* executa as solicitações simultaneamente até que o número total de solicitações ou ainda o tempo total de execução (mínimo de 10 segundos), exceder os limites especificados no parâmetro de configuração (`--num-threads`). Os resultados obtidos para o tempo de execução (em segundos) podem ser observados na Figura 5 e na Tabela 2.

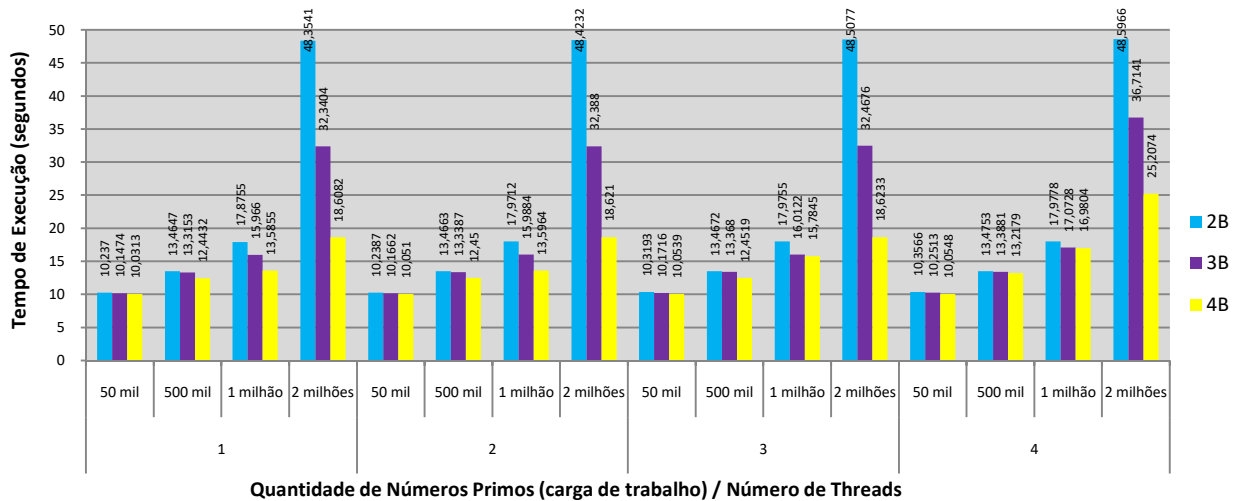


Figura 5. Tempo de Execução do SysBench – CPU Test

Para fins de análise do desempenho computacional nos diferentes modelos da plataforma Raspberry Pi usadas nesse projeto, primeiramente analisamos o comportamento de um único modelo de RPi executando com diferentes números de *threads* e cargas de trabalho, posteriormente analisamos o desempenho entre os diferentes modelos de RPi's.

Podemos observar na Tabela 2 um ligeiro aumento no tempo de execução do *SysBench*, conforme alteração na quantidade do número de *threads* e na carga de trabalho, tal como: para a RPi modelo 4B, calculando 2 milhões de números primos, quando executada com uma única *thread*, o tempo de execução foi de 6.6s mais rápido, ou seja, $\cong 35\%$ quando comparado com o tempo de execução do *SysBench* com 4 *threads* (ver Figura 6) e também $\cong 0,6\%$ e $\cong 0,8\%$ mais rápido comparado com as execuções usando 2 e 3 *threads* (respectivamente). O mesmo comportamento se repetiu para as demais cargas de trabalho (total de números primos) usadas nos experimentos. Conforme análises dos resultados, podemos inferir que tal fato ocorreu em virtude das constantes trocas de *threads* durante o processamento (granularidade fina), sendo que os experimentos realizados neste projeto foram baseados em um único nó, no entanto, geralmente o *benchmark SysBench* obtém um melhor desempenho computacional quando executado em uma estrutura arquitetural de *Cluster*, uma vez que o algoritmo é implementado utilizando as diretivas da biblioteca MPI (*Message Passing Interface*).

Tabela 2. Tempo de Execução do SysBench – CPU Test

Sistema Operacional	Threads	Total de Números Primos	Modelo da Rpi		
			2B	3B	4B
Raspbian	1	50 mil	10,2370s	10,1474s	10,0313s
		500 mil	13,4647s	13,3153s	12,4432s
		1 milhão	17,8755s	15,9660s	13,5855s
		2 milhões	48,3541s	32,3404s	18,6082s
	2	50 mil	10,2387s	10,1662s	10,0510s
		500 mil	13,4663s	13,3387s	12,4500s
		1 milhão	17,9712s	15,9884s	13,5964s
	3	2 milhões	48,4232s	32,3880s	18,6210s
		50 mil	10,3193s	10,1716s	10,0539s
		500 mil	13,4672s	13,3680s	12,4519s
		1 milhão	17,9755s	16,0122s	15,7845s
	4	2 milhões	48,5077s	32,4676s	18,6233s
		50 mil	10,3566s	10,2513s	10,0548s
		500 mil	13,4753s	13,3881s	13,2179s
		1 milhão	17,9778s	17,0728s	16,9804s
	2 milhões	48,5966s	36,7141s	25,2074s	

```

sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)
2
3 Running the test with following options:
4 Number of threads: 1
5 Initializing random number generator from current time
6
7
8 Prime numbers limit: 2000000
9
10 Initializing worker threads ...
11
12 Threads started!
13
14 CPU speed:
15   events per second: 0.05
16
17 General statistics:
18   total time: 18.6082s
19   total number of events: 1
20
21 Latency (ms):
22   min: 18608.10
23   avg: 18608.10
24   max: 18608.10
25   95th percentile: 18738.17
26   sum: 18608.10
27
28 Threads fairness:
29   events (avg/stddev): 1.0000/0.00
30   execution time (avg/stddev): 18.6081/0.00
    
```

(a) 1 Thread

```

sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)
2
3 Running the test with following options:
4 Number of threads: 4
5 Initializing random number generator from current time
6
7
8 Prime numbers limit: 2000000
9
10 Initializing worker threads ...
11
12 Threads started!
13
14 CPU speed:
15   events per second: 0.16
16
17 General statistics:
18   total time: 25.2074s
19   total number of events: 4
20
21 Latency (ms):
22   min: 25187.46
23   avg: 25194.72
24   max: 25207.32
25   95th percentile: 24994.79
26   sum: 100778.90
27
28 Threads fairness:
29   events (avg/stddev): 1.0000/0.00
30   execution time (avg/stddev): 25.1947/0.01
    
```

(b) 4 Threads

Figura 6. Execução do SysBench na RPi 4B Calculando 2 Milhões de Números Primos

Analisando os modelos de RPi's executando diferentes cargas de trabalho e números de *threads*, podemos destacar que a plataforma RPi modelo 4B obteve o menor tempo de execução do *SysBench*, em todas os experimentos, quando comparado com as plataformas modelos 2B e 3B, ainda vale ressaltar que a versão do modelo 4B usada nos experimentos possui 2GB de memória RAM enquanto as demais apenas 1GB.

Observando a Figura 5 e os valores da Tabela 2, constatamos que o *SysBench* foi executado $\cong 2,5x$ e $\cong 2x$ mais rápido na Raspberry Pi 4B, quando comparado com os modelos de RPi 2B e 3B (respectivamente), usando uma carga de trabalho de 2 milhões de números primos e apenas uma *thread*. Já quando o número de *thread* é igual a 4 e a carga de trabalho se mantém a mesma, a diferença reduz para $\cong 2x$ e $\cong 1,5x$. No entanto, conforme já apresentado na Tabela 1, as especificações arquiteturais de hardware do modelo da RPi 4B (CPU, ISA, RAM e outros), são superiores aos dos modelos 2B e 3B, consequentemente, o tempo de execução das aplicações nesta versão da RPi, teoricamente são sempre mais rápido.

Portanto, conforme as hipóteses apresentadas na Seção 1 (Introdução) deste artigo, podemos afirmar que a plataforma Raspberry Pi, pode sim proporcionar desempenho computacional aceitável quando executado diversos tipos de aplicações que demandam por poder de processamento. Também, vimos que os diferentes modelos de RPi's existentes na atualidade, embora alguns apresentem componentes de hardware e configurações semelhantes, sempre que a Fundação lança um novo modelo o mesmo tende a apresentar um melhor desempenho computacional em comparação com os seus antecessores, conforme anúncios da Fundação. E por fim, conforme alterações nos parâmetros de configuração de um *benchmark*, é possível afirmar que para cada modelo da plataforma RPi, há um impacto distinto nos resultados dos experimentos.

4. Conclusões e Trabalhos Futuros

Mediante aos experimentos realizadas com o *benchmark SysBench*, usando o sistema operacional Linux distribuição *Raspbian*, executando nas plataformas Raspberry Pi modelos 2B, 3B e 4B, foi constatado e apresentado neste artigo (ver Figura 5 e Tabela 2) que o modelo 4B da RPi, para a métrica CPU, atingiu uma performance superior a 2,5x mais rápida na execução do *SysBench* com uma carga de trabalho que realiza o cálculo de 2 milhões de números primos em apenas uma *thread*. Infere-se que tal fato ocorre em virtude deste modelo apresentar melhores componentes de hardware, além de outros fatores. Também destacamos que para o mesmo modelo de RPi, quando executado apenas com uma *thread*, o tempo de execução foi menor quando comparado com a execução com duas ou mais *threads* (ver Figura 6 e Tabela 2), isso ocorreu em virtude das constantes trocas entre as *threads* (granularidade fina) durante o processamento do *SysBench*.

Como ideias para trabalhos futuros sugerem-se: (i) análise do desempenho computacional incluindo outros SO's, como: *Ubuntu Mate*, *Arch Linux ARM*, Windows 10 IoT, RISC OS e outros; (ii) a execução do *benchmark SysBench* em um *Cluster*; (iii) análise do desempenho computacional utilizando outras suítes de *benchmarks*, tais como: *HPL*, *nBench*, *UnixBench* e outras.

Agradecimentos

Os autores agradecem ao Instituto Federal de Rondônia (IFRO), pelo apoio financeiro concedido através do Edital Nº 4/2021/REIT - PROPESP/IFRO, que proporcionou a execução desta pesquisa.

Referências

- Dinndorf, J. C. (2014) "Performance Comparison of Operating Systems for the Raspberry Pi". In *14th Winona Computer Science Undergraduate Research Symposium*, Winona, MN, USA, pg 11-17.
- Frpi. (2022) "Foundation Raspberry Pi". Disponível em <https://www.raspberrypi.org/>. Acessado em 25 de maio de 2022.
- Furber, S. (2000) "ARM System-on-Chip Architecture". Addison-Wesley, 2nd edition, 432p.
- Kennedy, P. (2016) "Raspberry Pi model B 512MB Linux Test Suite Benchmarks". Disponível em <http://www.servethehome.com/raspberry-pi-model-b-512mb-linux-test-suite-benchmarks/>. Acessado em 30 de junho de 2022.
- Lima, V. G. S., Dias, W. R. A., Melo, J. D. de, Moreno, E. D. (2015) "Análises de Sistemas Operacionais Linux Usando Plataforma Embarcada". In *XVI Simpósio em Sistemas Computacionais de Alto Desempenho (WSCAD-WIC 2015)*, Florianópolis, Santa Catarina, Brasil, pg 1-8.
- Mappuji, A., Effendy, N., Mustaghfirin, M., Sondok, F., Yuniar, R. P., Pangesti, S. P. (2016) "Análises de Sistemas Operacionais Linux Usando Plataforma Embarcada". In *8th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Yogyakarta, Indonesia, pg 16-20.
- Martins, Claylson. (2022) "Raspberry Pi Comemora 10 Anos com 46 Milhões de Dispositivos Vendidos". Disponível em <https://sempreupdate.com.br/raspberry-pi-comemora-10-anos-com-46-milhoes-de-dispositivos-vendidos/>. Acessado em 01 de julho de 2022.
- Mate, Ubuntu. (2022) "Ubuntu-MATE Community". Disponível em <https://ubuntu-mate.community/t/ubuntu-mate-15-04-possible-on-original-pi-b-v1-2-not-newer-2-mod-b/1497>. Acessado em 01 de junho de 2022.
- Merces, R. (2013) "Raspberry Pi - Conceito & Prática". Rio de Janeiro, Brasil: Ciência Moderna, 81p.
- Microsoft. (2016) "Windows 10 IoT Core Insider Preview - Overview and Getting Started". Disponível em <https://dev.windows.com/en-US/iot>. Acessado em 10 de junho de 2022.
- Morabito, R. (2016) "A Performance Evaluation of Container Technologies on Internet of Things Devices". In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs)*, San Francisco, CA, USA, pg 297-298.
- Richardson, M., Wallace, S. (2013) "Primeiros Passos com o Raspberry Pi". São Paulo, Brasil: Novatec, 192p.
- Seal, D. (2001) "ARM Architecture Reference Manual". Addison-Wesley, 2nd edition, 816p.
- Schot, N. J., Velthuis, P. J. E., Postema, B. F. (2016) "Capabilities of Raspberry Pi 2 for Big Data and Video Streaming Applications in Data Centres". In *18th International GI/ITG Conference on "Measurement, Modelling and Evaluation of Computing Systems" and "Dependability and Fault-Tolerance"* (MMB & DFT), Münster, North Rhine-Westphalia, Germany, pg 200-214.
- Upton, E., Halfacree, G. (2013) "Raspberry Pi Manual do Usuário". São Paulo, Brasil: Novatec, 269p.