

**Entrega prática 2 - Introdução à instalação de sistema operacional em sistemas embarcados**  
**SEL0337 - Projetos em Sistemas Embarcados**

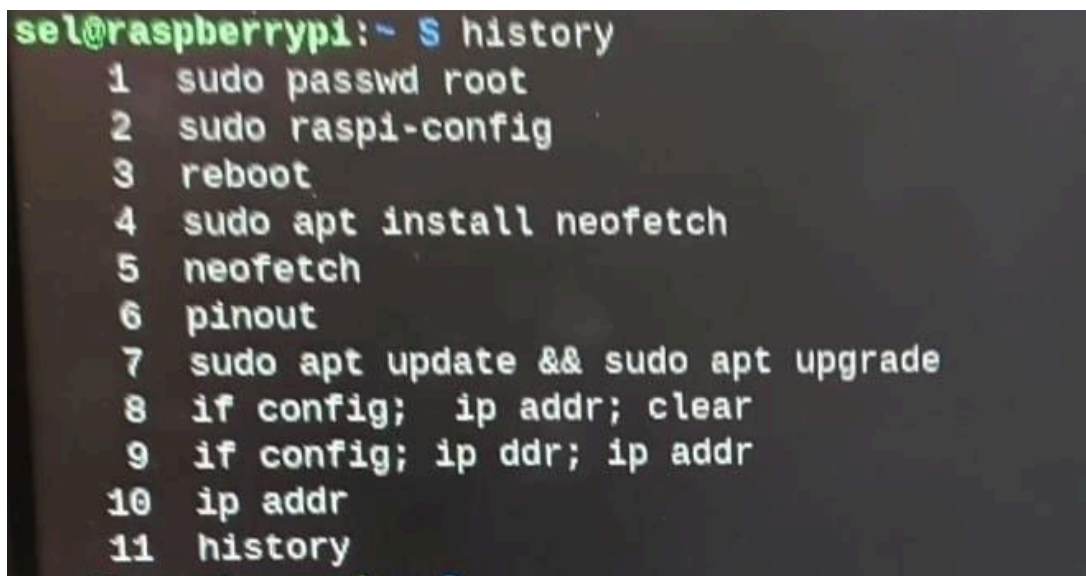
Fábio Roberto Alcazar Frias Junior - N° USP 14569060  
Yuri Thadeu Oliveira Costa - N° USP 14754821

## **1.0 Objetivos**

Esta prática teve como objetivo realizar a instalação do Raspberry Pi OS, um sistema operacional utilizado em SBCs (Single Board Computers) da Raspberry. Neste documento de entrega, será apresentado o procedimento realizado para a instalação do sistema operacional, conforme indicado no roteiro da prática. Além disso, também serão apresentadas as respostas das questões listadas no final do roteiro. Por fim, vale ressaltar que, por mais que o roteiro indique a captura de tela das linhas de comando do terminal, este documento conta com fotos tiradas do próprio monitor, já que não havia como exportar as capturas de tela da Raspberry.

## **2.0 Desenvolvimento**

Na figura 1, é possível ver o histórico de comandos utilizados no terminal durante a prática.



```
sel@raspberrypi:~ $ history
1  sudo passwd root
2  sudo raspi-config
3  reboot
4  sudo apt install neofetch
5  neofetch
6  pinout
7  sudo apt update && sudo apt upgrade
8  if config; ip addr; clear
9  if config; ip ddr; ip addr
10 ip addr
11 history
```

Figura 1: foto do histórico de comandos utilizados na prática.

Já na figura 2, é possível ver o detalhamento do comando “neofetch”. Ele nos mostra algumas informações referentes ao sistema operacional instalado e ao hardware utilizado. Na linha “OS”, temos qual Linux está sendo, bem como sua versão e sua arquitetura. Na linha “Host”, temos as informações de hardware do SBC utilizado. Na linha “Kernel”, temos a versão do kernel utilizada. Por fim, na linha “Memory”, temos a quantidade de memória disponível em relação a quantidade de memória total.

```
File Edit Tabs Help
sel@raspberrypi:~$ sudo apt install neofetch
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
neofetch is already the newest version (7.1.0-4).
0 upgraded, 0 newly installed, 0 to remove and 200 not upgraded.
sel@raspberrypi:~$ neofetch

_,metSSSSSgg.
,gSSSSSSSSSSSSSP.
,gSSP" ""YSS.".
,SSP' `SSS.
',SSP ,ggs. `SSb:
`dSS' ,SP" `SSS
SSP dS' ,SSP
SS: SS. - ,dSS'
SS; YSb._ ,dSP'
YSS. . "YSSSP"
`SSb "-._
`YSS
`YSS.
`SSb.
`YSSb.
`"YSb._
`""

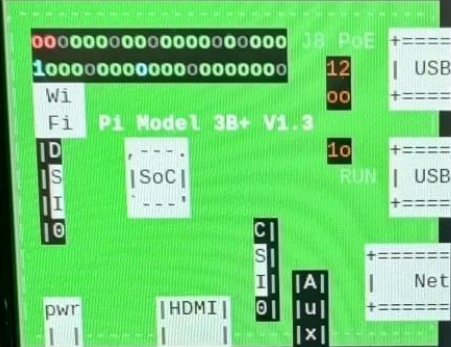
sel@raspberrypi
-----
OS: Debian GNU/Linux 12 (bookworm) aarch64
Host: Raspberry Pi 3 Model B Plus Rev 1.3
Kernel: 6.12.25+rpt-rpi-v8
Uptime: 1 min
Packages: 1600 (dpkg)
Shell: bash 5.2.15
Resolution: 1920x1080
DE: labwc:wlroots
WM: dwc
Theme: PiXflat [GTK3]
Icons: PiXflat [GTK3]
Terminal: lxterminal
Terminal Font: Monospace 10
CPU: (4) @ 1.400GHz
Memory: 379MiB / 906MiB

sel@raspberrypi:~$
```

Figura 2: foto do comando neofetch.

Na figura 3, temos o comando pinout detalhado e a vista superior do modelo de Raspberry Pi utilizado. Ele nos mostra a função de cada um dos pinos presentes no barramento do SBC. Além disso, também fornece algumas informações referentes ao hardware, como modelo da Raspberry Pi, o SoC utilizado, portas USB e Ethernet, se possui wi-fi e Bluetooth e, por fim, se possui conectores para displays e câmeras.

```
pi@raspberrypi:~$ S pinout
Description      : Raspberry Pi 3B+ rev 1.3
Revision         : a020d3
SoC              : BCM2837
RAM             : 1GB
Storage         : MicroSD
USB ports       : 4 (of which 0 USB3)
Ethernet ports  : 1 (300Mbps max. speed)
Wi-fi          : True
Bluetooth      : True
Camera ports (CSI) : 1
Display ports (DSI): 1
```



```
J8:
3V3 (1) (2) 5V
GPIO2 (3) (4) 5V
GPIO3 (5) (6) GND
GPIO4 (7) (8) GPIO14
GND (9) (10) GPIO15
GPIO17 (11) (12) GPIO18
GPIO27 (13) (14) GND
GPIO22 (15) (16) GPIO23
3V3 (17) (18) GPIO24
GPIO10 (19) (20) GND
GPIO9 (21) (22) GPIO25
GPIO11 (23) (24) GPIO8
GND (25) (26) GPIO7
GPIO0 (27) (28) GPIO1
GPIO5 (29) (30) GND
GPIO6 (31) (32) GPIO12
GPIO13 (33) (34) GND
GPIO19 (35) (36) GPIO16
GPIO26 (37) (38) GPIO20
GND (39) (40) GPIO21

RUN:
POWER ENABLE (1)
RUN (2)

POE:
TR01 TAP (1) (2) TR00 TAP
TR03 TAP (3) (4) TR02 TAP

For further information, please refer to https://pinout.xyz
```

Figura 3: foto do comando pinout.

Já na figura 4, temos o comando addr detalhado.



```

sel@raspberrypi:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether b8:27:eb:27:fc:81 brd ff:ff:ff:ff:ff:ff
3: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1280 qdisc pfifo_fast state UP group default qlen 1000
    link/ether b8:27:eb:72:a9:d4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.111/24 brd 192.168.1.255 scope global dynamic noprefixroute wlan0
        valid_lft 6678sec preferred_lft 6678sec
    inet6 fe80::f09e:6476:4db5:97ea/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
sel@raspberrypi:~$

```

Figura 4: foto do comando ip addr.

Por fim, na figura 5 temos o comando rpi-update detalhado.

```

sel@raspberrypi:~$ sudo rpi-update cac01bed1224743104cb2a4103605f269f207b1a#6.1.54
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** Performing self-update
*** Relaunching after update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
FW_REV:cac01bed1224743104cb2a4103605f269f207b1a
BOOTLOADER_REV:360324a17edebeb625c3fffe872321a0e2c20dc4
*** We're running for the first time
*** Backing up files (this will take a few minutes)
*** Backing up firmware
*** Backing up modules 6.12.25+rpt-rpi-v8
WANT_32BIT:0 WANT_64BIT:1 WANT_64BIT_RT:0 WANT_PI4:1 WANT_PI5:1

Updating a system with initramfs configured is not supported by rpi-update.
If your system relies on drivers provided by the initramfs (e.g. custom filesystem options)
it may not boot without regenerating the initramfs.
If you are unsure, test if your system boots with initramfs options disabled from config.txt

Would you like to proceed? (y/N)

#####
WARNING: This update bumps to rpi-6.1.y linux tree
See: https://forums.raspberrypi.com/viewtopic.php?t=344246

'rpi-update' should only be used if there is a specific
reason to do so - for example, a request by a Raspberry Pi
engineer or if you want to help the testing effort
and are comfortable with restoring if there are regressions.

DO NOT use 'rpi-update' as part of a regular update process.
#####
Would you like to proceed? (y/N)

```

Figura 5: foto do comando rpi-update.

### 3.0 Questões

1 - Em até 10 linhas, compare de forma geral as placas SBCs Raspberry Pi 3B+, Raspberry Pi 4B e Raspberry Pi 5 em termos de recursos, limitações e novidades presentes na última versão.

A Raspberry Pi 3B+ conta com CPU quad-core Cortex-A53 de 64 bits a 1,4 GHz, GPU VideoCore IV e 1 GB de RAM. Apesar de útil em automação e IoT, ela não comporta tarefas mais pesadas, principalmente quando rodadas em paralelo. Já Raspberry Pi 4B possui uma CPU Cortex-A72 até 1,8 GHz, GPU VideoCore VI, suporte a até 8 GB de RAM e interfaces modernas como USB 3.0, Ethernet gigabit e saída dupla HDMI, o que amplia seu potencial em executar tarefas mais pesadas. Por fim, a Raspberry Pi 5 trouxe uma CPU Cortex-A76 de 2,4 GHz, GPU VideoCore VII, RAM de até 16 GB e um chip dedicado aos I/Os, que permitiu melhorias em desempenho de periféricos. Além disso, com a inclusão de um barramento PCIe abre novas possibilidades de expansão, como uso de SSDs. Como limitação, a Raspberry Pi 5 exige fonte de alimentação e resfriamento mais robustos.

**2 - Em até 10 linhas, compare os seguintes SoCs em termos de núcleos, arquitetura, GPU e conectividade:**

- **BCM2837B0 (Raspberry Pi 3B+)**
- **BCM2711 (Raspberry Pi 4)**
- **BCM2712 (Raspberry Pi 5)**

**Relacione esses SoCs com outros do mercado (escolher somente outros dois modelos a partir das alternativas abaixo):**

- **Amlogic S905X3 (Set-Top-Box BTV11 - disponível no laboratório)**
- **Texas Instruments AM3358 (SBC BeagleBone Black - disponível no laboratório)**
- **Rockchip RK3288 (SBC ASUS Tinker Board) ou RK3588S (SBC Orange Pi 5), ou outra SBC comercial de sua escolha.**
- **NXP i.MX6 - Computador em Módulo (CoM) Toradex Colibri iMX6 e placa base Aster (disponível no laboratório).**

O SoC BCM2837B0 possui quatro núcleos ARM Cortex-A53 de 64 bits a 1,4 GHz e GPU VideoCore IV. Ele oferece conectividade básica com Wi-Fi, Bluetooth e portas USB 2.0, sendo usado em tarefas simples, porém com limitações de capacidade de processamento. Já o BCM2711 conta com quatro núcleos Cortex-A72 a até 1,8 GHz, GPU VideoCore VI, suporte a até 8 GB de RAM e interfaces como USB 3.0 e HDMI 4K, resultando em um desempenho mais elevado. Por fim, o BCM2712 conta com quatro núcleos Cortex-A76 de 2,4 GHz, GPU VideoCore VII, suporte a até 16 GB de RAM e recursos como lane PCIe 2.0, saídas HDMI duplas 4K60 e melhor desempenho de I/Os via o chip auxiliar RP1.

Comparando com outros modelos do mercado, o Rockchip RK3588S é mais avançado em termos de paralelismo, pois possui 8 núcleos (4× Cortex-A76 e 4× Cortex-A55) e GPU Mali-G610 MP4, além de suporte a vídeo até 8K. Isso o torna superior em aplicações de alto desempenho, embora o Pi 5 reduza essa distância em tarefas gerais. Já o Texas Instruments AM3358 é um SoC mais simples, com apenas um núcleo Cortex-A8 de até 1 GHz, GPU limitada e conectividade voltada a aplicações industriais. Ele é ideal para automação e sistemas embarcados de baixo consumo, mas não compete em desempenho com as gerações mais novas da Raspberry Pi.