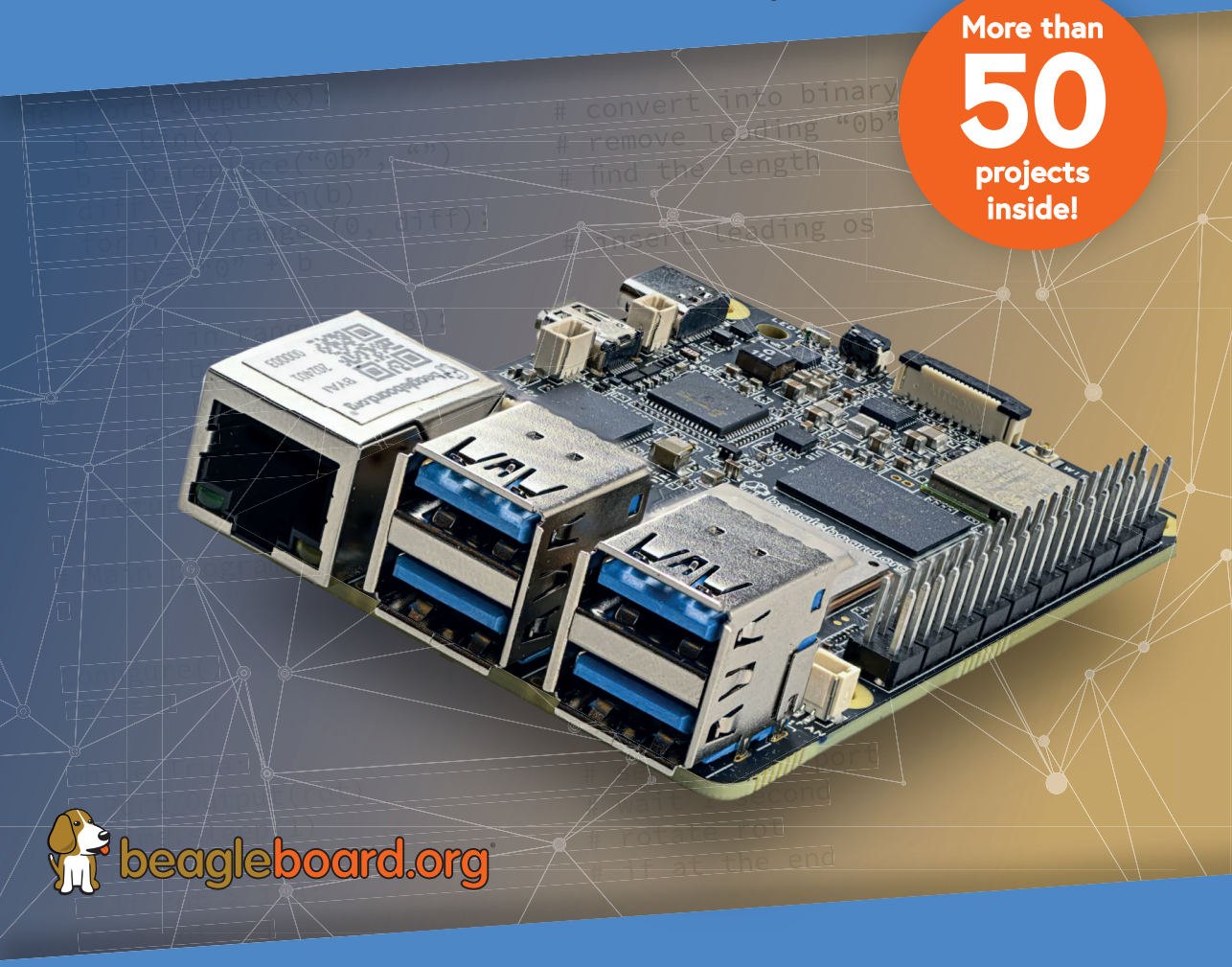
elektor books

The BeagleY-AI Handbook

A Practical Guide to AI, Python, and Hardware Projects

More than 50 projects inside!



beagleboard.org

Dogan Ibrahim
Ahmet Ibrahim

elektor
design > share > earn

The BeagleY-AI Handbook

A Practical Guide to AI, Python, and Hardware Projects

Dr. Dogan Ibrahim
Ahmet Ibrahim BSc, MSc

elektor
design › share › earn

This is an Elektor Publication. Elektor is the media brand of
Elektor International Media B.V.
PO Box 11, NL-6114-ZG Susteren, The Netherlands
Phone: +31 46 4389444

Declaration

The author and publisher have made every effort to ensure the accuracy of the information contained in this book. They do not assume, or hereby disclaim, any liability to any party for any loss or damage caused by errors or omissions in this book, whether such errors or omissions result from negligence, accident, or any other cause.

ISBN 978-3-89576-656-5 Print
ISBN 978-3-89576-657-2 eBook

www.elektor.com
Editor: Glaucileine Vieira
Prepress Production: D-Vision, Julian van den Berg
Printers: Ipskamp, Enschede, The Netherlands

Contents

Chapter 1 • Introduction

1.1 The BeagleY-AI Single Board Computer (SBC)

BeagleY-AI is a low-cost, open-source, and powerful 64-bit quad-core single-board computer, equipped with a GPU, DSP, and vision/deep learning AI accelerators, designed for developers and makers. Developed by BeagleBoard.org Foundation, it is designed to meet the needs of both professional developers and educational environments. It is affordable, easy to use, and eliminating barriers to innovation. Developers can explore in-depth lessons or push practical applications to their limits without restrictions.

For more information about BeagleY-AI, including detailed specifications, documentation, and resources for getting started, visit the official website at

beagleboard.org

The board is controlled by the Debian Linux operating system, which includes a built-in development environment. This enables the seamless running of AI applications on a dedicated 4 TOPS co-processor, while simultaneously handling real-time I/O tasks with an 800 MHz microcontroller.

BeagleY-AI is based on the Texas Instruments AM67A Arm-based vision processor. It features a quad-core 64-bit Arm®Cortex®-A53 CPU subsystem at 1.4 GHz, dual general-purpose C7x DSP with Matrix Multiply Accelerator (MMA) capable of 4 TOPs each, Arm Cortex-R5 subsystem for low-latency I/O and control, a 50 GFLOP GPU, video and vision accelerators, and other specialized processing capabilities.

In this chapter, you will learn the basic features and hardware details of the BeagleY-AI board. A comparison is made with the popular Raspberry Pi 5 computer which has very similar board layout and features. In the remaining chapters of the book, you will learn how to install the operating system, how to access the BeagleY-AI board remotely, how to create Python programs to run on the board, and how to create software-only and hardware-based projects using the peripheral ports such as GPIO, SPI, UART, I^2C, and many others.

1.2 BeagleY-AI Features

The board has the following features:

Feature	Description
Processor	Texas Instruments AM67A, Quad 64-bit Arm® Cortex®-A53 @1.4 GHz, multiple cores including Arm/GPU processors, DSP, and vision/deep learning accelerators
RAM	4GB LPDDR4
Wi-Fi	Beagleboard BM3301, 802.11ax
Bluetooth	Bluetooth Low Energy 5.4 (BLE)

USB Ports	4x USB 3.0 ports (5Gbps shared) + USB 2.0 Type-C Port with Device-mode capability
Ethernet	Gigabit Ethernet, with PoE+ support (requires separate PoE HAT)
Camera/Display	2 x 4-lane MIPI camera connector (one connector muxed with DSI capability)
Display Output	1 x HDMI display, 1 x OLDI display, 1 x DSI MIPI Display
Real-time Clock (RTC)	Supports external coin-cell battery for power failure time retention
Debug UART	1 x 3-pin debug UART
Power	5 V/3 A DC power via USB-C
Power Button	On/Off included
PCIe Interface	PCI-Express® Gen3 x 1 interface for fast peripherals (requires separate M.2 HAT or other adapter)
Expansion Connector	40-pin header
Fan connector	1 x 4-pin fan connector, supports PWM control and fan speed measurement
Storage	microSD card slot with UHS-1 support
Tag Connect	1 x JTAG, 1 x External PMIC programming port

Table 1.1: BeagleY-AI features

The AM67A scalable processor family is based on the evolutionary Jacinto™ 7 architecture, targeted at Smart Vision Camera and General Compute applications. The AM67A processor family is designed for a broad set of cost-sensitive, high-performance computing applications in factory automation, building automation, human-machine interface, security systems, test and measurement, robotics, industrial PC, and other markets.

For more information about the AM67A processor, visit:

https://www.ti.com/product/AM67A

1.3 BeagleY-AI Board Component Layout

Front view

Figure 1.1 shows the components at the front of the board. Starting from the top-right-hand corner of the board and moving to the left we can see the following components:

- 4-pin External fan connector
- AM67A processor
- 40-pin expansion header
- 4 GB LPDDR4 memory
- BM3301 WiFi (802.1ax) + BLE (v5.4)
- BM3301 antenna
- PCIe port (Gen 3)
- Power On/Off button

- Bicolour LED
- Power management IC
- USB-C power and USB-2 port
- microHDMI monitor port
- 3-pin UART debug port
- 4-lane MIPI CSI connector
- 4-lane MIPI DSI/CSI connector
- Power over Ethernet port (PoE)
- Gigabit Ethernet port
- 2 x USB-3 (5 Gbps) ports
- 2 x USB-3 (5 Gbps) ports

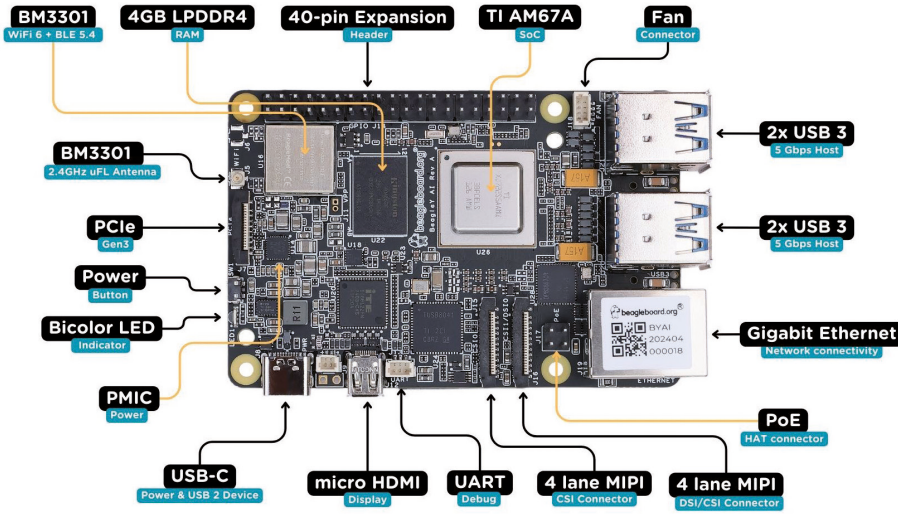


Figure 1.1 BeagleY-AI front view.

Back view

Figure 1.2 shows the components at the back of the board, which include the following:

- JTAG SoC debug connector
- JTAG PMIC debug connector
- OLDI display connector
- microSD card adapter

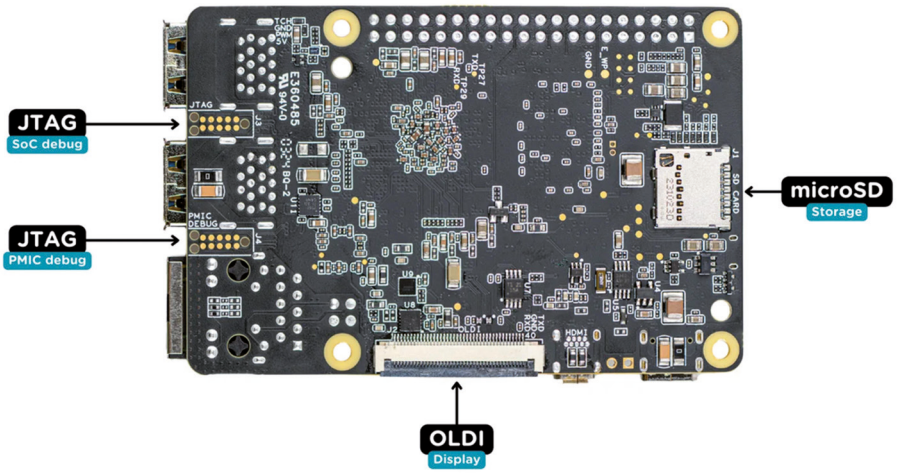


Figure 1.2 BeagleY-AI back view.

1.4 Comparison with the Raspberry Pi 5

Figure 1.3 shows the front views of the BeagleY-AI board and the Raspberry Pi 5 board. The two boards look identical in size and in most component layouts. Table 1.1 shows a comparison of the BeagleY-AI and the Raspberry Pi 5.

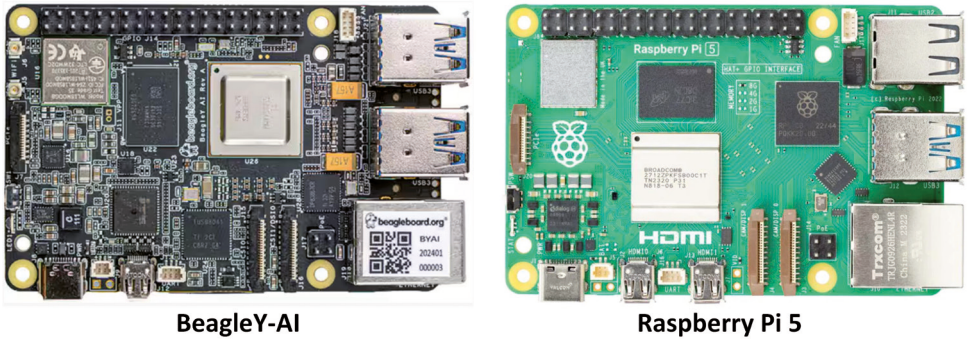


Figure 1.3 BeagleY-AI and the Raspberry Pi 5.

Feature	BeagleY-AI	Raspberry Pi 5
CPU	AM67A, Quad-core 64-bit, Cortex-A53 1.4GHz	BCM2712, Quad-core 64-bit Cortex-A76 2.4GHz
Memory	4GB	2GB, 4GB, 8GB
R5 core	YES	None
microHDMI	1	2
USB-3 ports (5Gbps)	4	2
USB-2 port (480Mbps)	1	2

Display support	3x (1x HDMI, 1x OLDI, 1x DSI)	2x HDMI
Graphics processing unit	IMG-BXS-4-64	Videocore VII
Dual C7x DSP with Matrix multiply accumulator (4 TOPS), NPU	1	
CSI/DSI ports	1	0
Video encode/decode	1	None
CSI port	2	2
Fan connector	1	1
UART connector	1	1
PCIe port	1	1
microSD card slot	1	1
40-pin GPIO header	1	1
Ethernet port (Gigabit)	1	1
Power button	1	1
WiFi + BLE	1	1

Table 1.2 Comparison of the BeagleY-AI and Raspberry Pi 5

1.5 Pros and Cons

Pros:

- **AI Performance**: The dual C7x DSPs and MMAs deliver up to 4 TOPS, making it ideal for deep learning tasks.
- **Connectivity**: With USB 3.0, Gigabit Ethernet, Wi-Fi 6, and Bluetooth 5.4, the board is well-equipped for various applications.
- **Expandability**: The PCIe Gen3 x1 connector and 40-pin GPIO header offer significant customization options.
- **Open-Source Hardware**: Users can access and modify all hardware design files, fostering innovation and adaptation.
- **Industrial-Grade Components**: The use of Texas Instruments hardware ensures reliability and long-term support, making it suitable for both development and deployment.

Cons:

- **CPU Performance**: The 1.4 GHz quad-core Cortex-A53 is underwhelming compared to newer SBCs.

- **RAM Limitations**: 4 GB of LPDDR4 RAM may not be sufficient for all applications.
- **Software Gaps**: Some AI features and tools are not fully supported, limiting the board's out-of-the-box capabilities.
- **Heat Management**: The board runs warm under load, and while it's fanless, some users may prefer active cooling.

Chapter 2 • Installing the Operating System

2.1 Overview

It is necessary to install a compatible operating system on a microSD card before the BeagleY-AI SBC board can be used. In this chapter, you will learn how to install the BeagleY-AI Debian operating system on a blank microSD card. Details on how to access the board remotely are also given in this chapter.

2.2 The Installation of the Operating System

Before installing the operating system, make sure you have the following:

- 5 V 3 A power supply
- 32 GB microSD card
- Boot image (operating system software image)

Using the bb-imager

You can use the bb-imager to install the operating system on the SD card. The steps are as follows::

- Download and install the bb-imager for your operating system from the following link:

  https://beagley-ai.beagleboard.io/bb-imager/

- Click to start the bb-imager. You should see a screen similar to the one shown in Figure 2.1.

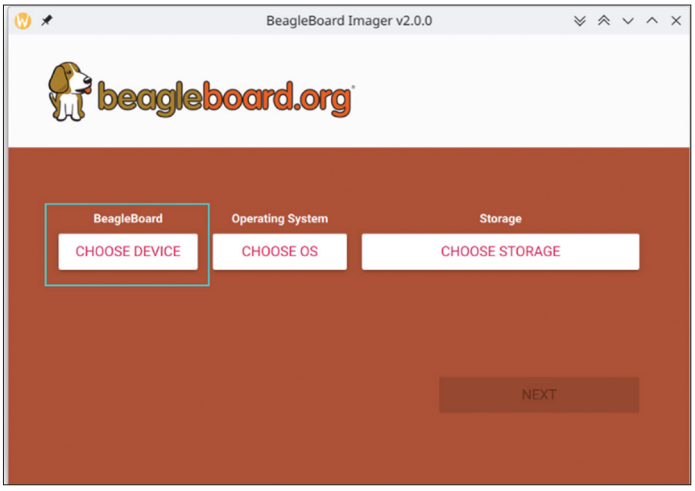


Figure 2.1 bb-imager screen.

- Select **BeagleY-AI** as the device (Figure 2.2)

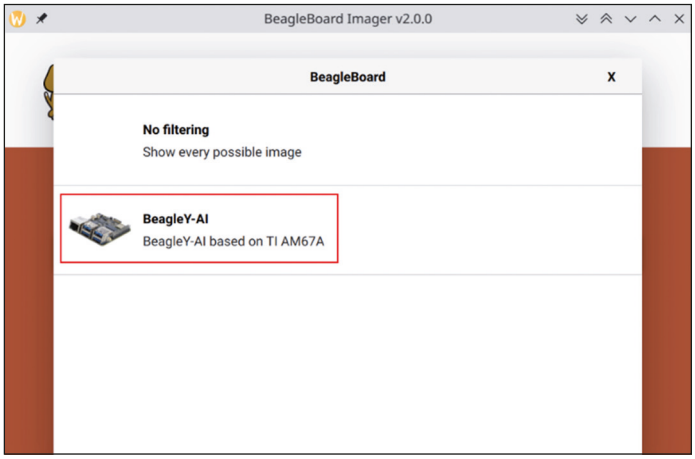


Figure 2.2 Enter the details.

- Choose the operating system as **BeagleY-AI Debian XFCE (Recommended)** as shown in Figure 2.3.

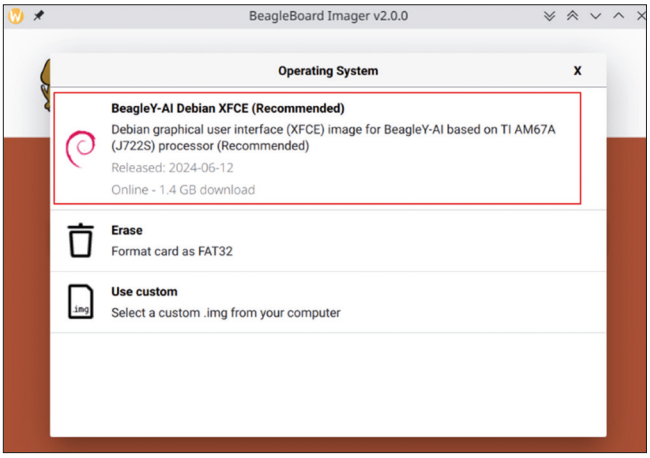


Figure 2.3 Choose the operating system.

- Choose your SD card storage and click **NEXT**

- Click **EDIT SETTINGS** and enter your chosen username, password, Wi-Fi SSID, Wi-Fi password, and time zone (Figure 2.4)

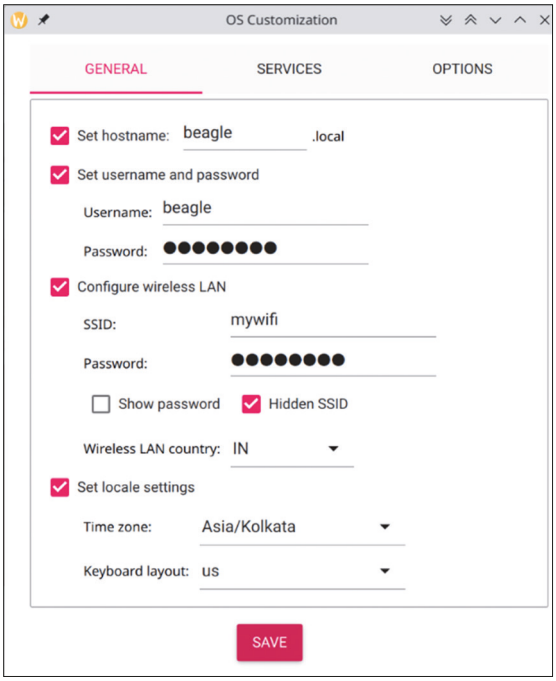


Figure 2.4 Edit the settings.

- Click **SERVICES** and make sure that the **Enable SSH** and **Use password authentication** are checked.
- Click **SAVE**, and then click **YES** on the screen **Would you like to apply OS customization settings?**
- Click **YES** to confirm that all existing data will be deleted on the SD card and to continue writing the operating system image on the SD card. Wait until the writing and the verification processes are complete.
- Remove the microSD card adapter from the PC and insert the microSD card into the slot on your BeagleY-AI as in Figure 2.5.
- Connect a monitor to the micro HDMI port of your BeagleY-AI board.
- Connect a keyboard and mouse to the USB-3 ports.
- Connect 5 V 3 A power supply to the USB-C power port of the BeagleY-AI.
- Figure 2.11 shows a typical setup with a monitor.

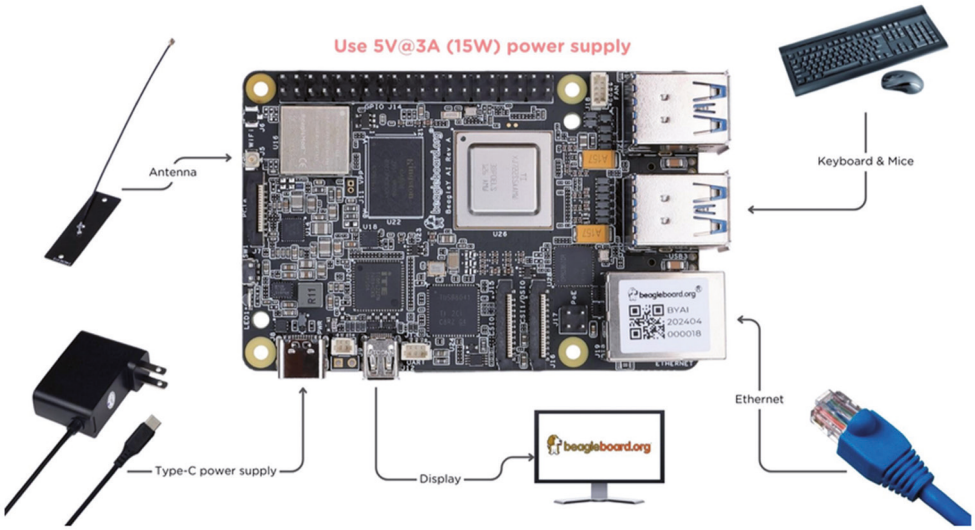


Figure 2.5 A typical setup (BeagleBoard.org).

- After a while you should see the green LED heartbeat and the GUI desktop displayed as shown in Figure 2.6. Please note, it may take several minutes.

2.3 Connection to a Wi-Fi

Follow these steps to connect to a Wi-Fi network:

- Click the wireless icon at the top right-hand side of the screen.
- A list of Wi-Fi networks will be displayed.
- Click **Connect** to connect to your network and enter your password.
- Click **Submit** (Figure 2.6).

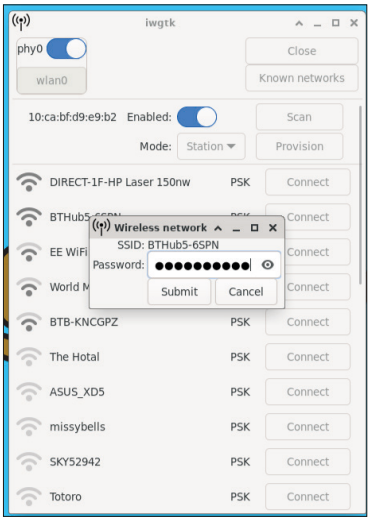


Figure 2.6 Click Submit.

- After a short wait, your BeagleY-AI will connect to your Wi-Fi. Click **Close** to exit the window. You should see the Wi-Fi icon change color to green, indicating a successful connection.

You can display the IP address of your connection as follows:

- Click **Applications**, then **Terminal Emulator.**
- In the terminal, enter the following command:

  ```
  sudo ifconfig
  ```

- You should see your IP address displayed under wlan0. In the author's setup, the IP address was 192.168.1.127 (see Figure 2.7).

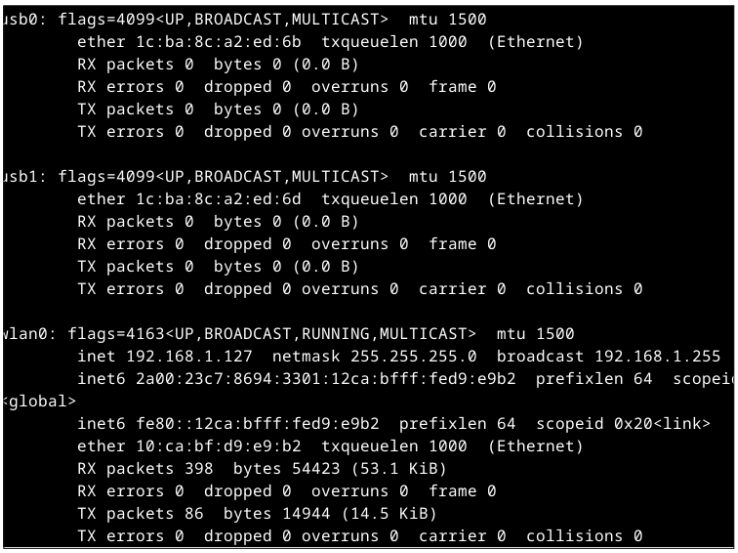


Figure 2.7 Command ifconfig (part of the display is shown).

2.4 Accessing Your BeagleY-AI Console from Your PC – The PuTTY Program

In many applications, you may want to access your BeagleY-AI from your PC over the Wi-Fi link. This can be done using a terminal emulator program on your PC. The author uses the popular PuTTY for this purpose. You can download PuTTY from the following website:

https://www.putty.org

- PuTTY is a standalone program and there is no need to install it. Simply double-click to run it. You should see the Putty startup screen as shown in Figure 2.8.

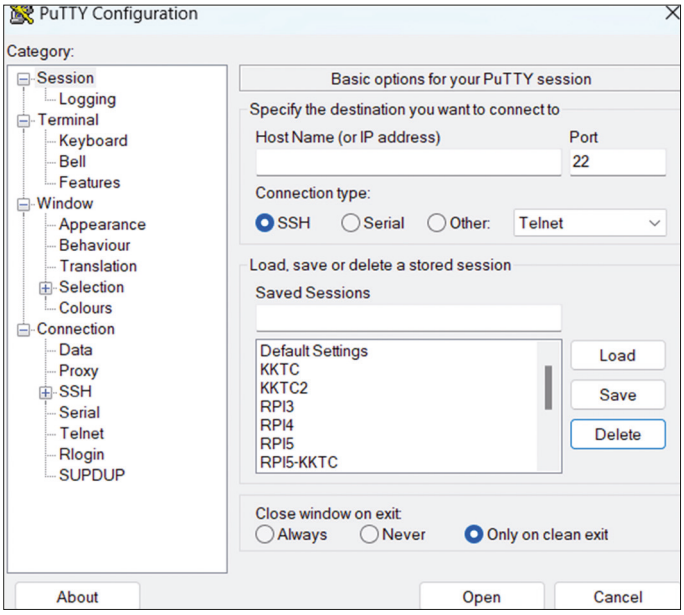


Figure 2.8 Putty startup screen.

- Make sure that the Connection type is SSH and enter the IP address of your BeagleY-AI. You can obtain the IP address by entering the command **ifconfig** as shown earlier.

- Click **Open** in PuTTY after entering the IP address and selecting **SSH.**

- The first time you run PuTTY, you may get a security message. Click **Yes** to accept this security alert.

- You will then be prompted to enter the BeagleY-AI username and password (these were entered in the **sysconf.txt** file during installation of the operating system). You can now enter all Console-based commands through your PC. Figure 2.9 shows the PuTTY screen with default screen settings.

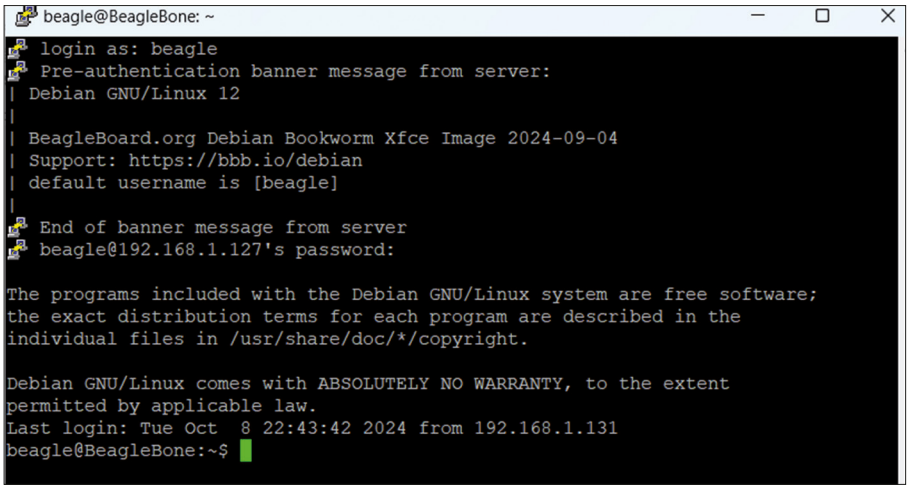


Figure 2.9 PuTTY screen with default settings.

- To change your password, enter the following command:

  passwd

- To restart the BeagleY-AI enter the following command:

  sudo reboot

- To shut down the BeagleY-AI enter the following command. Never shut down by pulling the power cable, as this may result in the corruption or loss of files:

  sudo shutdown –h now

2.4.1 Configuring PuTTY

By default, the **PuTTY** screen background is black with white foreground characters. The author prefers to have a white background with black foreground characters, and the font size set to 12 points in bold. It is recommended that you save your settings so that they are available the next time you use PuTTY. Follow these steps to configure PuTTY with the desired settings:

- Restart PuTTY.
- Select **SSH** and enter the Raspberry Pi IP address.
- Click **Colours** under **Window.**
- Set the **Default Foreground** and **Default Bold Foreground** colors to black (Red:0, Green:0, Blue:0).

- Set the **Default Background** and **Default Bold Background** to white (Red:255, Green:255, Blue:255).
- Set the **Cursor Text** and **Cursor Colour** to black (Red:0, Green:0, Blue:0).
- Select **Appearance** under **Window** and click **Change** in **Font settings**. Set the font to **Bold 12**.
- Select **Session**, give the session a name (e.g., MyZero), and click **Save**.
- Click **Open** to open the **PuTTY** session with the saved configuration.
- Next time you re-start the **PuTTY**, select the saved session and click **Load,** followed by **Open**, to start a session with the saved configuration.

Figure 2.10 shows the PuTTY screen with black bold characters on a white background. In this example, the PuTTY session was named as beagle.

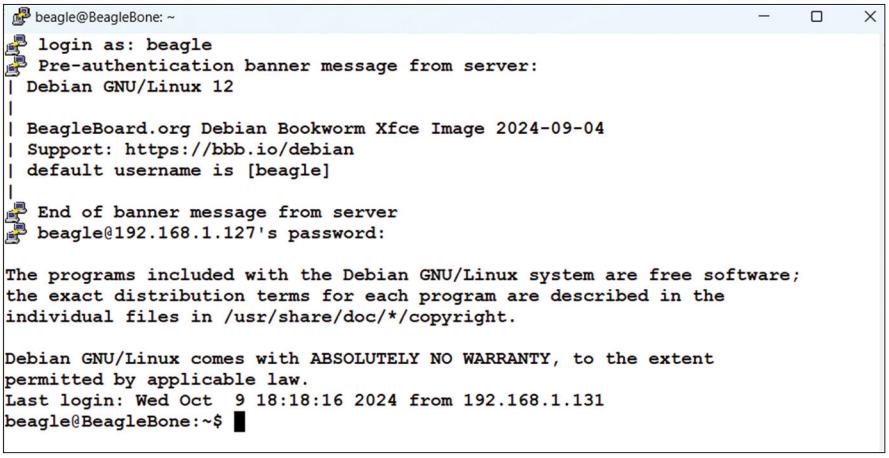


Figure 2.10 Putty screen with white background and black characters.

2.5 BeagleY-AI CPU Temperature

Without a heatsink, the Beagle-Y-AI typically heats up to about 58 - 60°C when idle. With 4 cores running in a complex operation, the CPU temperature can reach nearly 70°C. It is recommended to use a heatsink or an active cooler (such as the Raspberry Pi 5 active cooler) to help lower the device temperature, particularly during CPU-intensive tasks.

The CPU temperature can be displayed by entering the following command. As shown in the example below, the temperature is in milli-Celsius. In this case, the CPU temperature was measured shortly after the board was started, and it was 48.819ºC:

```
beagle@beagle:~ $ cat /sys/devices/virtual/thermal/thermal_zone
[0-2]/temp

47697
48146
48819
beagle@beagle:~ $
```

Chapter 3 • Using the Console Commands

3.1 Overview

BeagleY-AI is based on a version of the Linux operating system, one of the most popular operating systems in use today. Linux shares similarities with other operating systems, such as Windows and UNIX, and is an open-source system based on UNIX, developed collaboratively by many companies since 1991. In general, Linux is harder to manage than some other operating systems like Windows but offers more flexibility and configuration options. There are several popular versions of the Linux operating system, such as Debian, Ubuntu, Red Hat, Fedora, and others.

Linux commands are text-based. In this chapter, you will be looking at some of the useful Linux commands and see how you can manage your BeagleY-AI using these commands.

The console commands can either be entered using the Putty terminal emulator, as described in the previous chapter, or they can be entered using the Terminal Emulator application in GUI Desktop.

3.2 The Command Prompt

Assuming your username is **beagle**, after you log in to BeagleY-AI, you will see the following prompt displayed where the system waits for you to enter a command:

```
beagle@beagle: ~$
```

Here, the **~** character indicates that you are currently in your default directory.

3.3 Useful Console Commands

In this section, you will be learning some of the useful Console commands, with examples provided for each command. **In this chapter, commands entered by the user are shown in bold for clarity**. Also, it is important to remind you that all commands must be terminated by the Enter key.

3.3.1 System and user information

These commands are useful as they provide information about the system. The command **cat /proc/cpuinfo** displays information about the processor (the command **cat** displays the contents of a file, and in this example, it shows the contents of the **/proc/cpuinfo** file). Figure 3.1 shows an example display, where only part of the display is shown here.

```
beagle@beagle:~$ cat /proc/cpuinfo
processor       : 0
BogoMIPS        : 400.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32 cpuid
CPU implementer : 0x41
CPU architecture: 8
CPU variant     : 0x0
CPU part        : 0xd03
CPU revision    : 4

processor       : 1
BogoMIPS        : 400.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32 cpuid
CPU implementer : 0x41
CPU architecture: 8
CPU variant     : 0x0
CPU part        : 0xd03
CPU revision    : 4

processor       : 2
BogoMIPS        : 400.00
Features        : fp asimd evtstrm aes pmull sha1 sha2 crc32 cpuid
CPU implementer : 0x41
CPU architecture: 8
CPU variant     : 0x0
CPU part        : 0xd03
CPU revision    : 4

processor       : 3
```

Figure 3.1 Command: **cat /proc/cpuinfo** *(part of the display is shown).*

The command **uname –s** displays the operating system kernel name, which is Linux. The command **uname –a** displays complete detailed information about the kernel and the operating system. An example is shown in Figure 3.2.

```
beagle@beagle:~$ uname -a
Linux beagle 6.1.83-ti-arm64-r63 #1bookworm SMP PREEMPT_DYNAMIC Wed Jul 10 23:0
:56 UTC 2024 aarch64 GNU/Linux
beagle@beagle:~$
```

Figure 3.2 Command: **uname – a**.

The command **cat /proc/meminfo** displays information about the memory on your BeagleY-AI, such as the total memory and free memory at the time the command is issued. Figure 3.3 shows an example, where only part of the display is shown here.

```
beagle@beagle:~$ cat /proc/meminfo
MemTotal:        3883876 kB
MemFree:         2566148 kB
MemAvailable:    3027204 kB
Buffers:           30508 kB
Cached:           540800 kB
SwapCached:            0 kB
Active:          1034332 kB
Inactive:          96644 kB
Active(anon):     561736 kB
Inactive(anon):        0 kB
Active(file):     472596 kB
Inactive(file):    96644 kB
Unevictable:          64 kB
Mlocked:              64 kB
SwapTotal:       4194300 kB
SwapFree:        4194300 kB
Zswap:                 0 kB
Zswapped:              0 kB
Dirty:                 8 kB
Writeback:             0 kB
AnonPages:        544692 kB
Mapped:           293488 kB
Shmem:              2060 kB
KReclaimable:      47144 kB
Slab:              93992 kB
SReclaimable:      47144 kB
SUnreclaim:        46848 kB
```

Figure 3.3 Command: **cat /proc/meminfo** *(part of the display is shown).*

The command **whoami** displays the name of the current user. In this case, **beagle** is displayed as the current user.

A new user can be added to your BeagleY-AI using the command **useradd**. In the example in Figure 3.5, a user called **Jane** is added. A password for the new user can be added using the **passwd** command followed by the username. In Figure 3.4, the password for user Jane is set to **mypassword** (not displayed for security reasons). Notice that both the **useradd** and **passwd** commands are privileged and the keyword **sudo** must be entered before these commands. Notice that the **–m** option creates a home directory for the new user.

```
beagle@beagle:~$ sudo useradd -m Jane
beagle@beagle:~$ sudo passwd Jane
New password:
Retype new password:
passwd: password updated successfully
beagle@beagle:~$
```

Figure 3.4 Commands: **useradd** *and* **passwd***.*

You can log in to the new user account by specifying the username and password. You can type the command **exit** to log out from the new account.

The command **sudo apt-get upgrade** is used to upgrade all the software packages on the system.

3.3.2 Some useful commands

To display the default home directory, enter:

```
beagle@beagle: ~$ pwd
/home/beagle
beagle@beagle: ~$
```

To display the directory structure, enter the command **ls /** (Figure 3.5):

```
beagle@beagle:~$ ls /
bin   data  etc   lib         media  opt   root  sbin  sys  usr
boot  dev   home  lost+found  mnt    proc  run   srv   tmp  var
beagle@beagle:~$
```

Figure 3.5 Files in the directory.

To show the subdirectories and files in your working directory, enter **ls** (Figure 3.6)

```
beagle@beagle:~$ ls
Desktop    Downloads  Pictures  Templates  led.py
Documents  Music      Public    Videos
beagle@beagle:~$
```

Figure 3.6 Files in the home directory.

Notice that the subdirectories are displayed in blue and the files in black.

The command **ls** can take a number of arguments. Some examples are given below.

To display the subdirectories and files in a single row (Figure 3.7).

```
beagle@beagle:~$ ls -1
Desktop
Documents
Downloads
Music
Pictures
Public
Templates
Videos
led.py
beagle@beagle:~$
```

Figure 3.7 Files in a single row.

To display the file types, enter the command **ls –F**. Note that directories have a "/" after their names, and executable files have a "*" character after their names:

To list the filenames separated by commas, enter the command **ls –m**.

You can mix the arguments, as shown in Figure 3.8.