1.) Create a Minimal Image with the manjaro-arm-tools and then flash the image to the eMMC Module by using the Balena Etcher program. An eMMC Module with 16GB capacity is sufficient as the whole setup requires only 2.8GB in space when finished. (Manjaro x86 Installation or Live-Disk /Image required)

In the Desktop Environment use the Package Manager and search for manjaro-arm-tools and install.
Or in the Terminal use sudo pacman -S manjaro-arm-tools to install the package.
Then create the image with sudo buildarmimg -d pbpro -e minimal

2.) After burning the image to the eMMC module do not eject the eMMC module!

lsblk to find the mount point for boot partition.

nano /media/user/BOOT_MNJRO/extlinux/extlinux.conf open extlinux.conf and find line below

```
initrd=/initramfs-linux.img console=tty1 console=ttyS0,115200 root=LABEL=ROOT_MNJRO rw rootwait
bootsplash.bootfile=bootsplash-themes/manjaro/bootsplash
```

delete → bootsplash.bootfile=bootsplash-themes/manjaro/bootsplash and save the file.

Now eject the eMMC module and install on Pinebook Pro computer board.

3.) Let the Board boot-up and re-size the eMMC, upon completion finish the Manjaro Installer script and create a user and set root password and answer all remaining Questions asked. It loads the Command Line Interface.

4.) Change to Arch Linux ARM repositories

sudo nano /etc/pacman.conf open pacman config and amend as below

```
#
# /etc/pacman.conf
#
# See the pacman.conf(5) manpage for option and repository directives

#
# GENERAL OPTIONS
#
[options]
# The following paths are commented out with their default values listed.
# If you wish to use different paths, uncomment and update the paths.
#RootDir            = /
#DBPath             = /var/lib/pacman/
#CacheDir           = /var/cache/pacman/pkg/
#LogFile            = /var/log/pacman.log
#GPGDir             = /etc/pacman.d/gnupg/
#HookDir            = /etc/pacman.d/hooks/
HoldPkg             = pacman glibc
#XferCommand        = /usr/bin/curl -L -C - -f -o %o %u
#XferCommand        = /usr/bin/wget --passive-ftp -c -O %o %u
#CleanMethod        = KeepInstalled
Architecture        = aarch64

# Pacman won't upgrade packages listed in IgnorePkg and members of IgnoreGroup
#IgnorePkg =
#IgnoreGroup        =

#NoUpgrade          =
#NoExtract =

# Misc options
#UseSysLog
Color
TotalDownload
CheckSpace
#VerbosePkgLists
```

```
# By default, pacman accepts packages signed by keys that its local keyring
# trusts (see pacman-key and its man page), as well as unsigned packages.
SigLevel    = Required DatabaseOptional
LocalFileSigLevel = Optional
#RemoteFileSigLevel = Required

# NOTE: You must run `pacman-key --init` before first using pacman; the local
# keyring can then be populated with the keys of all official Manjaro Linux
# packagers with `pacman-key --populate archlinuxarm`.

#
# REPOSITORIES
#   - can be defined here or included from another file
#   - pacman will search repositories in the order defined here
#   - local/custom mirrors can be added here or in separate files
#   - repositories listed first will take precedence when packages
#     have identical names, regardless of version number
#   - URLs will have $repo replaced by the name of the current repo
#   - URLs will have $arch replaced by the name of the architecture
#
# Repository entries are of the format:
#       [repo-name]
#       Server = ServerName
#       Include = IncludePath
#
# The header [repo-name] is crucial - it must be present and
# uncommented to enable the repo.
#

[core]
Include = /etc/pacman.d/mirrorlist

[extra]
Include = /etc/pacman.d/mirrorlist

[community]
Include = /etc/pacman.d/mirrorlist

[alarm]
Include = /etc/pacman.d/mirrorlist

[aur]
Include = /etc/pacman.d/mirrorlist

[pinebookpro]
Server = https://nhp.sh/pinebookpro/
SigLevel = Optional

# An example of a custom package repository.  See the pacman manpage for
# tips on creating your own repositories.
#[custom]
#SigLevel = Optional TrustAll
#Server = file:///home/custompkgs
```

```
sudo nano /etc/pacman.d/mirrorlist.pacnew
```

open mirrorlist and change as below

```
#
# Arch Linux Arm repository mirrorlist
# Generated on 2020-04-30
#

## Geo-IP based mirror selection and load balancing
Server = http://mirror.archlinuxarm.org/$arch/$repo
```

```
sudo nano /etc/pacman.d/mirrorlist                open mirrorlist and change as below
```

```
#
# Arch Linux Arm repository mirrorlist
# Generated on 2020-02-11
#

## Geo-IP based mirror selection and load balancing
Server = http://mirror.archlinuxarm.org/$arch/$repo
```

5.) Configure WiFi

```
sudo systemctl status iwd.service        check if iwd is running
sudo systemctl enable iwd.service        installs iwd
sudo systemctl start iwd.service         starts iwd

iwctl                                    bash changes to iwd interactive prompt

station wlan0 scan                       scans for wireless networks
station wlan0 get-networks               shows all available wireless networks
station wlan0 connect SSID               connects to specified network, may ask for passphrase
station wlan0 show                       check connection status of wireless network

quit                                     reverts back to bash

nano /etc/resolv.conf                    should be nameserver 192.168.1.xxx
ip a                                     check if an IP address has been issued
```

6.) Update the system

```
sudo mkdir /etc/pacman.d/hooks
sudo rm -rf /etc/lsb-release
sudo pacman-key --init                          reinitiate the keyring
sudo pacman-key --populate archlinuxarm         and install ALARM package signing keys
sudo pacman -Sy                                 sync the local repo’s
sudo pacman -S linux-pbp                        replace the Manjaro Kernel with Nadia’s Kernel
sudo pacman -Syuu

sudo reboot
```

7.) Install system monitoring

```
sudo pacman -S screenfetch
sudo pacman -S htop
```

8.) Install Wayland Display Server and Gnome Desktop Environment

```
sudo pacman -S wayland
sudo pacman -S gnome                     select → ALL (just press ENTER), then select → 1 & 1
sudo systemctl enable gdm.service
sudo systemctl start gdm.service
```

9.) Install Firefox

```
sudo pacman -S firefox firefox-extension-privacybadger firefox-ublock-origin        select → 1 (GNU Fonts)

sudo reboot
```

10.) Done, enjoy your new setup !