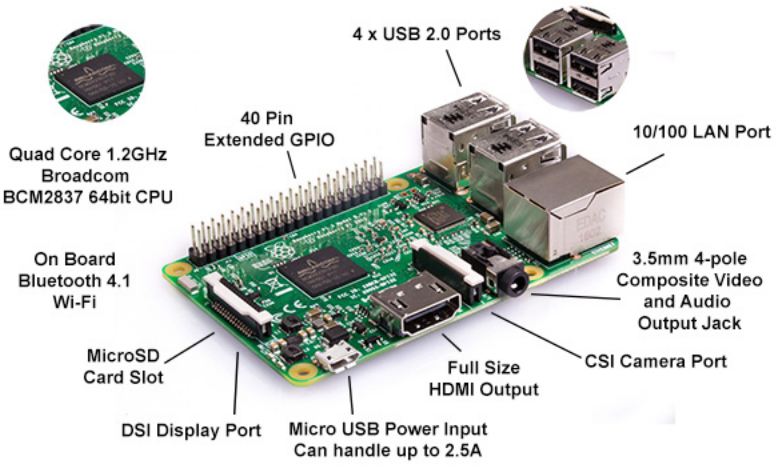
dmt

fast-secure&resilient
media-search-p2p-iot

+


4 x USB 2.0 Ports
40 Pin Extended GPIO
10/100 LAN Port
Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
On Board Bluetooth 4.1 Wi-Fi
3.5mm 4-pole Composite Video and Audio Output Jack
MicroSD Card Slot
CSI Camera Port
Full Size HDMI Output
DSI Display Port
Micro USB Power Input Can handle up to 2.5A

1 🌀 DMT-OS vs. Raspbian OS

Raspbian is the basis for **dmt-os** *for Single Board Computers // RaspberryPi series.* We get from Raspbian to **dmt-os** in a few relatively easy steps.

Imagine dmt-os less as an operating system for your particular single board computer and more as an interconnected operating system for multiple devices and people behind these devices. **dmt-os connects.**

Raspbian is still the operating system on your RaspberryPi when you use dmt-os and you can do everything you can do with Raspbian itself. **dmt-proc** and a few other services make your little computer capable to become one node in the broader **dmt-os connectome** concept. You cannot pinpoint **dmt-os** to one device because it partially exists "between devices". Connections are important and as direct as possible. **Connectomes** can be cut off from one another or they can be joined together to create broader connectomes. An example is a set of devices which particular user owns — this is this users' connectome. When s/he connects at least one device with a device from another user, these two connectomes are joined and new connectome now covers broader set of devices and users (two instead of one). What data and how exactly is shared between two connectomes is defined by both users on two devices that are joined with a connection.

2 🌀 Compatible RPi hardware

Very recent and still great **RaspberryPi 3B or 3B+**

Go to: http://downloads.raspberrypi.org/raspbian/images/raspbian-2019-04-09/

or **download this directly:** http://downloads.raspberrypi.org/raspbian/images/raspbian-2019-04-09/2019-04-08-raspbian-stretch.zip

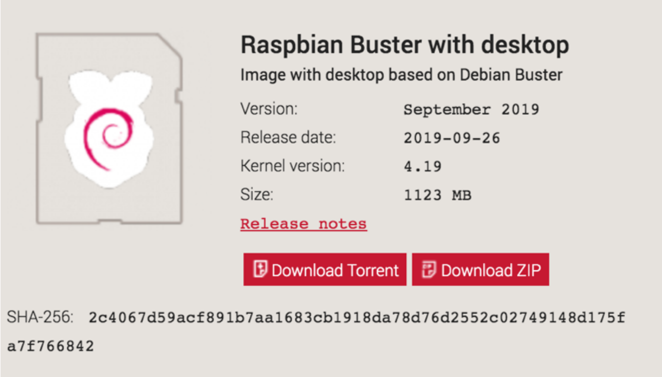
The amazing **mpv** (multimedia player) *does not work* properly on the latest version of Raspbian called Buster (it almost works but still - a glitch that happens too often is not a reliable system overall). This is why you should use the latest earlier version Stretch as in that download link! It has no general disadvantages over Buster of our purposes, or in general. Raspbian does not change that much from version to version, it does not have to because it is quite mature.

The New **RaspberryPi 4**

In this case you have to download Buster and while we wait for the amazing RPi_Mike to publish his latest script for compiling **mpv** on Raspbian Buster (https://www.raspberrypi.org/forums/viewtopic.php?f=38&t=199775&start=225) you can use dmt-os on your RPi4 in this way:

- in the meantime install **mpv** directly from apt-get and expect minor problems with video and audio play
- use **dmt-os** for other purposes except media play for now on this computer

Download **Raspbian Buster with desktop image**:
https://www.raspberrypi.org/downloads/raspbian/ (the .zip file)



Raspbian Buster with desktop

Image with desktop based on Debian Buster

Version: September 2019

Release date: 2019-09-26

Kernel version: 4.19

Size: 1123 MB

SHA-256: 2c4067d59acf891b7aa1683cb1918da78d76d2552c02749148d175f a7f766842

3 🌀 FLASH RASPBIAN-OS "SYSTEM IMAGE" TO SD CARD

Buy a SD card as described here:
https://github.com/uniqpath/info/tree/master/hardware#sd-cards

1. Prepare image *(optional step if you don't want to use keyboard + monitor for setting up wifi password in step 3.)*

Unzip, then mount the .img on your PC, edit it:

```
touch /boot/ssh

nano /boot/wpa_supplicant.conf
```

```
country=UK
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
  ssid="net"
  psk="password"
}
```

Unmount the .img and flash it to the SD card.

2. Flash the image

Linux, macOS, use **flash** command from dmt-framework on your laptop OR ... find some other solution. On Windows you can use "etcher".

3. Boot

Insert the SD card into the RPi and boot.

If you have monitor attached:

When GUI appears cancel all notifications like:
"ssh is enabled and the default password for the 'pi' user has not been changed. This is a security risk - please login as the 'pi' user and run Raspberry Pi Confguration to set a new password."
"Welcome to the Raspberry Pi Desktop! Before you start using it, there are a few things to set up. Press 'Next' to get started."
PRESS CANCEL on all of them.

Since we have ssh enabled, let's go to our laptop or PC now and forget about the attached keyboard and monitor on the RaspberryPi. They are not the focus of this type of computing. At least not on the small computers. This is a common fallacy people believe as beginners. Single Board Computers are not a cheaper replacement for your normal desktop computer but are a new class of devices that usually don't have monitor attached and are "always on and connected with ability to do low-level electronics interactions as well".

4 🌀 Connect to your Raspberry via exciting ssh command line

1. Find out what ip address was assigned to your RaspberryPi by your ethernet (wifi) router.

This is the hardest part but will be easy once you get dmt-proc on your system, one of its purposes is to make local device discovery a breeze.

When doing the first setup you have to use some IP scanner on your phone or laptop to find out which IP address your Raspberry is using. Alternative and much easier way is to **just go to wifi options via keyboard and monitor** that you must have attached to your RPi. Read device's IP address from there.

2. From the Putty program on Windows or from regular linux / macos shell type this:

```
ssh pi@device_ip
```

For example if your RPi IP address is "192.168.0.70", type:

```
ssh pi@192.168.0.70
```

When prompted for password, type: "raspberry" (this is the default preset password for the *pi* user on every Raspbian OS image).

5 🌀 Setup system users

You have two users on your system by default:

- **pi**
- **root**

Each linux system can have many users (like "pi" in this case) and only one "root user".

Change your default **pi user** password:

1) ssh into your device as already shown above, we repeat:

```
ssh pi@device_ip
```

2) Type:

```
passwd
```

Follow the instructions to change the password from the default "raspberry" to something else you will remember. For example: set this password: **dmtsystem** (for now) or your own once you are sure you won't forget or when you set up your **public/private key login system** for your raspberry (blockchains made that popular for sending digital money) -- see a further section on this. In that case passwords are less important and you can actually forget them -- each raspberry pi that you own has your public key in it and no passwords are needed to access its operating system directly. **dmt-system** provides its own security sistem with PINs and user passwords.

Now change **root user** password:

```
sudo su
```

This will switch from pi user to root user, type *passwd* command as before:

```
root@raspberrypi:~$ passwd
```

And follow instructions to change root users' password.

Now either:

a)

enable root login over ssh:

```
nano /etc/ssh/sshd_config
```

```
PermitRootLogin Yes
```

```
sudo systemctl restart ssh
```

b)

If you use ssh keys, then add your key to `~/.ssh/authorized_keys` instead of setting `PermitRootLogin Yes`. Or if you do this later, make sure to remove this line so that root login only with password is not possible anymore.

Usually the ~/.ssh directory doesn not exists and you have to do this:

```
mkdir -p ~/.ssh
echo "your_pubkey" > ~/.ssh/authorized_keys
```

OPTIMIZE SSH:

Edit **/etc/ssh/sshd_config** and uncomment:

```
UseDNS no
```

Restart:

```
/etc/init.d/ssh restart
```

6 🌀 SSH INTO DEVICE AS ROOT (use Putty or something similar on Windows)

```
ssh root@192.168.X.Y
```

You want root here because:

- dynamic gui can run on port 80 if it wants to (otherwise you'd have to install one more piece of software)
- you can only scan for nearby wifi access points with root user
- you can interact with GPIO on your SBC from dmt-proc if you wanted to
- mpv and pulse run as root for great audio and you want to control both directly

7 🌀 Basic device setup through raspi-config

```
sudo raspi-config
```

1. Localisation options:

configure locale:

Default is *en_GB.UTF-8* .... scroll a bit further down to *en_US.UTF-8* and press space to select this new option.

Select default locale for the system **en_US.UTF-8** on the next screen.

OPTIONAL FOR LATER:
if something gets corrupted along the way use this to fix:

> Fixing your Locale:
>
> So, for the people who had the same problem as me, here is the way you fix this issue:
>
> 1. Edit `sudo nano /etc/locale.gen` and uncomment the line with `en_US.UTF-8`
> 2. Run `sudo locale-gen en_US.UTF-8`
> 3. Run `sudo update-locale en_US.UTF-8`
>
> And there you have it. You won’t get any error messages about your locale anymore!

2. Interfacing options: enable some things for possible use later:

SPI, I2C, (Camera) [if you attach one]

3. Boot options: disable Splash Screen (B3)
4. Timezone

Localisation options -> Change Timezone

5. Set **128MB** GPU memory inside the *advanced / memory split* menu option
(if not already set)

Select Finish, the Pi will reboot, wait for it to come back up, then follow to the next step.

6. Change the hostname of your computer

Network Options / Hostname / OK

Press backspace a few times to delete the existing hostname and type "**dmt-os**" in the box.

Now exit raspi-config with <Finish> on the main menu.

When asked if you want to reboot (because of hostname change), DON'T DO IT.

8 🌀 Install basic linux packages

```
sudo apt-get update
```

```
sudo apt-get -y install build-essential tar unzip curl git screen ntp zip tree
lsof colordiff rsync net-tools cmake arp-scan netcat nmon highlight evtest iperf
iperf3 jq mosquitto mosquitto-clients socat
```

Fix arp-scan issue:

```
sudo chmod u+s /usr/bin/arp-scan

or

sudo chmod u+s /usr/sbin/arp-scan
```

TIP 💡 Recommended:

```
sudo apt-get -y remove avahi-daemon
```

Avahi-daemon which is the frequent source of networking issues.

9 🌀 INSTALL node.js ENVIRONMENT

```
curl -L https://git.io/n-install | bash
```

CONTINUE (y/N)? y → Node LTS version will install in a minute or two ...

1 0 🌀🌀🌀 INSTALL DMT-SYSTEM ENVIRONMENT 💡🚀🎸

```
git clone https://github.com/uniqpath/dmt.git ~/.dmt
cd ~/.dmt
./install
```

Configure **user.def** for full shell:

```
nano ~/.dmt/user/def/user.def
```

It should look like this:

```
user:
  shell: full
```

Remember that dmt-system can come from many sources. GitHub has the official version. Other branches are floating around and you can use them (one at a time) by adding this to your user.def:

```
user:
  shell: full
  dmtSource: [some-pointer]
```

The safest to use is the GitHub version on https://github.com/uniqpath/dmt.git

To update after changing the source, execute:

```
dmt next
```

TIP 💡 Autostart dmt process on boot:

RaspberryPi and other Linux systems (servers, laptops etc):

Login as root again... then:

```
EDITOR=nano crontab -e
```

delete everything (ctrl+k in nano editor)

Add this single line:

```
@reboot bash -ic ~/.dmt/etc/onboot/onboot
```

Save nano editor and quit: CTRL+O, ENTER, CTRL+X

Make nano editor the default:

```
sudo update-alternatives --config editor
```

There are 5 alternatives which provide `editor’.
Selection Alternative
——————————————————

```
1 /usr/bin/vim
2 /bin/ed
*+ 3 /bin/nano
4 /usr/bin/vim.basic
5 /usr/bin/vim.tiny
Press enter to keep the default[*], or type selection number: 3
```

macOS (additional info although this manual is describing the linux setup, in particular SingleBoardComputers like Raspberry Pi or ODROID):

```
cd ~/Library/LaunchAgents
cp ~/.dmt/etc/onboot/macos-launchagent-script/dmt-proc.plist .
launchctl load -F ~/Library/LaunchAgents/dmt-proc.plist
```

11 🌀 KIOSK MODE SETUP (we will show one browser tab on system boot)

CHROMIUM FULL SCREEN BROWSER INSTANCE AUTOSTART

Login as PI user!

ssh pi@DEVICE_IP

```
sudo apt-get install -y chromium-browser ttf-mscorefonts-installer unclutter x11-xserver-utils xdotool
```

⚠️ (REMINDER) you have to be logged in as "pi" user...

```
sudo nano /etc/lightdm/lightdm.conf
```

Find `xserver-command=X` (it may be commented out - '#' symbol in front of it - **REMOVE THE '#" SYMBOL**) and add things in bold if they are not present yet:

```
xserver-command=X -s 0 dpms -nocursor
```

(*"-s 0 dpms"* part is needed to prevent the display from going to sleep)

Now adjust the autostart script for browser (must be logged in as user "pi"):

```
mkdir -p ~/.config/lxsession/LXDE-pi
nano ~/.config/lxsession/LXDE-pi/autostart
```

Paste this contents:

```
@/usr/bin/chromium-browser --incognito --start-maximized --start-fullscreen --noerrdialogs --disable-translate --no-first-run --fast --fast-start --disable-infobars --disable-features=TranslateUI --overscroll-history-navigation=0 --disk-cache-dir=/dev/null --disable-pinch  --force-tablet-mode --tablet-ui --kiosk http://localhost

@unclutter
@xset s off
@xset s noblank
@xset -dpms
```

Save and exit (CTRL+O, ENTER, CTRL+X)

12 🌀 REBOOT

Use:

```
shut r
```

To restart the device and see if basic setup works. After reboot ssh to the device again.

TIP 💡 Turn off wifi if you use Raspberry's ethernet port (optional but sometimes recommended for even more stability)

To completely disable the onboard WiFi from the firmware on the Pi3, add

```
dtoverlay=pi3-disable-wifi
```

in */boot/config.txt*.

This is documented here

`/boot/config.txt` is the cleanest possible way to disable WiFi, and other peripherals.

TIP 💡 GET NICE **.DEF COLORS** into NANO EDITOR

a) Linux:

```
ln -s ~/.dmt/etc/syntax/nano/def.nanorc /usr/share/nano/def.nanorc
```

b) macOS:

```
brew install nano
```

```
sudo mkdir -p /usr/local/share/nano/
```

```
echo "include /usr/local/share/nano/*.nanorc" >> ~/.nanorc
echo "include ~/.dmt/etc/syntax/nano/*.nanorc" >> ~/.nanorc
```

Save (Ctrl+O, ENTER) and exit (Ctrl+X, ENTER)

1 🎸 AUDIO SETUP ■ SAVE SYSTEM VOLUME BETWEEN DEVICE REBOOTS

First execute

```
alsamixer
```

and set the default volume at 80 (up and down arrows, ESC to quit). Make sure you adjust your directly connected speakers volume so that this represents the maximum volume for that room. You can test playing some mp3 files with

```
mpv file.mp3
```

Now save the mixer settings into a custom file with `alsactl`:

```
alsactl --file ~/.config/asound.state store
```

Put this inside `/etc/rc.local`:

```
...

alsactl --file /root/.config/asound.state restore

exit 0
```

2 🎸 AUDIO AND VIDEO SETUP ■ COMPILE MPV

Move to screen in case ssh connection is lost:

```
screen -S mpv_compile

cd ~/.dmt/etc/scripts

./mpv_compile_rpi
```

Wait around 1h for **mpv** and dependencies to compile and install.

You could also install it via *sudo apt-get install mpv* but it is not recommended because you will get an older version with glitchy video playback. Do it only if you don't have time to compile yourself and you will only use your device for playing music, not for video.

OPTIONAL: set logrotate for dmt mpv debug log so that log file doesn't grow uncontrollably (you can optionally enable mpv process log into ~/.dmt/log/mpv.log later)
Note: dmt.log is already logroated inside the dmt-proc itself...

```
sudo nano /etc/logrotate.d/dmt-mpv
```

Paste this:

```
/root/.dmt/log/mpv.log {
    daily
    rotate 0
    size 500k
    copytruncate
    dateext
    missingok
}
```

3🎸 AUDIO SETUP ■ INSTALL PULSE AUDIO (RECOMMENDED)

Pure ALSA setup also works but you will experience weird bugs, like no sound after you let playback on pause for more than 12h. We tried to circumvent this issue inside our player code and we succeeded in 90% of the cases, then just gave up and tried PulseAudio ourselves and it works much better with other great benefits as well. Setup is easy with a few steps. **It is worth it in any case !**

INSTALL

```
sudo apt-get install -y pulseaudio
```

CREATE STARTUP SCRIPT

```
cd /etc/systemd/system
```

Create empty file:

```
sudo nano pulseaudio.service
```

Paste this content:

```
[Unit]
Description=PulseAudio system server

[Service]
ExecStart=/usr/bin/pulseaudio --system --realtime --disallow-exit --no-cpu-limit

[Install]
WantedBy=multi-user.target
```

Save and exit (CTRL+O, ENTER, CTRL+X)

Enable:

```
systemctl enable pulseaudio
```

CONFIGURE

a)

```
nano /etc/pulse/client.conf
```

Change

; autospawn = yes

to:

autospawn = no

(note: you have to remove the ';' and change *yes* to *no*)
(tip: you can use CTRL+W in nano to search for "autospawn")

b)

```
nano /etc/pulse/default.pa
```

COMMENT OUT THIS LINE:
# load-module module-suspend-on-idle

(put '#' in front of 'load-module....' as shown)

c)

```
nano /etc/pulse/system.pa
```

Add text in bold to this line:

```
load-module module-native-protocol-unix auth-anonymous=true
```

START THE PULSEAUDIO SERVER
(or reboot the device since you enabled the script for loading on boot)

```
systemctl start pulseaudio
```

You can verify it's running like this:

```
ps aux | grep pulseaudio
```

IMPORTANT ✓ Add this parameter to /boot/config.txt to avoid most audio/video problems when using dmt-os RPi as a set-top box on a standard TV with HDMI inputs

```
hdmi_force_hotplug=1
```

in */boot/config.txt*.

If this line is already present with '#' prepended, just delete this symbol to make the line active (eg. uncomment it).

This forces to display through HDMI event if no HDMI screen is detected. Prevents missing image (black screen) when turning TV on and off independently from RPi.

IMPORTANT ✓ Add this mpv configuration to fix a memory leak / bug

Open *~/.config/mpv/mpv.conf* file or create it if it doesn't exist. Add these lines to the end of the file:

```
# restrict readahead and backbuffer to 40MiB from default of 400MiB
demuxer-max-bytes=41943040
demuxer-max-back-bytes=41943040
```

This will reduce the size of the cache and esp. with web streaming you will have much less problems with RPi becoming unresponsive, slow or simply freezing.

TIP 💡 ACCESS POINT SETUP

Access Point => device shares it's own WIFI and other devices connect to it
Station mode => device is connected to another Access Point (like router)

Install AP / STATION switching mode functionality:

```
ap install
```

TIP 💡 GLITCH FIX -- CHANGE HOSTNAME

Google Chrome won't start after changing hostname

-- "This profile appears to be in use by another Chromium process (543)...

Unlock profile and relaunch

```
rm -rf /home/pi/.config/chromium/Singleton*
```