Raspberry PI mit xfce4 nachrüsten

Es kam die Frage auf, den Raspberry mit **Pi Star** am Monitor zu betreiben und darauf das ***Dashbord*** direkt anzuzeigen. Vorab der Hinweis. Mach eine Sicherung der Konfiguration des Pi-Star-System. Es kann immer mal was schief laufen, sicher ist sicher.

Versuchen wir es Schritt für Schritt. Vorgabe ist ein Rasbian mir PiStar lauffähig im Einsatz. Mit http://pi-star:2222/ auf die SSH Konsole gehen und einloggen mit *pi-star* und dem bekannten *Password*.

In den Write-Mode umschalten

- rpi-rw

Einmal Update aufrufen mit

- sudo apt update ; sudo apt upgrade

Wenn das fehlerfrei geklappt hat geht es mit Insatllieren des XFCE weiter

- sudo apt install -y xserver-xorg xfce4 xfce4-goodies xfce4-whiskermenu-plugin mugshot libpam-gnome-keyring gnome-system-tools network-manager-gnome

Es werden etwas mehr als 1GB Daten nachinstalliert. Bitte einen **Kaffee** oder **Tee** genießen, bis der Vorgang abgeschlossen ist. Eventuell wird noch FIX MISSING benötigt

- sudo apt install –fix-missing

Wenn das alles ohne Fehler geklappt hat, wird der XFCE zum Start als Default angegeben

- sudo systemctl get-default

Hier wird eine Info ausgegeben (zum Beispiel *graphical.target*), die man mit dem nächsten Befehl übergibt

- sudo systemctl set-default *graphical.target*

-> Dann geben wir den Manager für die Session an

- sudo update-alternatives –config x-session-manager

-> und wählen im Anschluss den XFCE als Desktop aus

- sudo update-alternatives –config x-window-manager

-> Als nächstes ändere den Wert von greeter-hide-users in false:

- sudo nano /usr/share/lightdm/lightdm.conf.d/01_debian.conf

Es kann mit sudo raspi-config (*Eventuell muss dies mit #sudo*

apt install raspi-config noch nachinstalliert werden) bei ***Display Options*** unterpunkt ***Underscan*** wunschgemäß noch die Funktion ***overscan*** eingestellt bzw. geändert werden.

TIPP : Beachtet auch die Zeichensatz Einstellungen US / DE

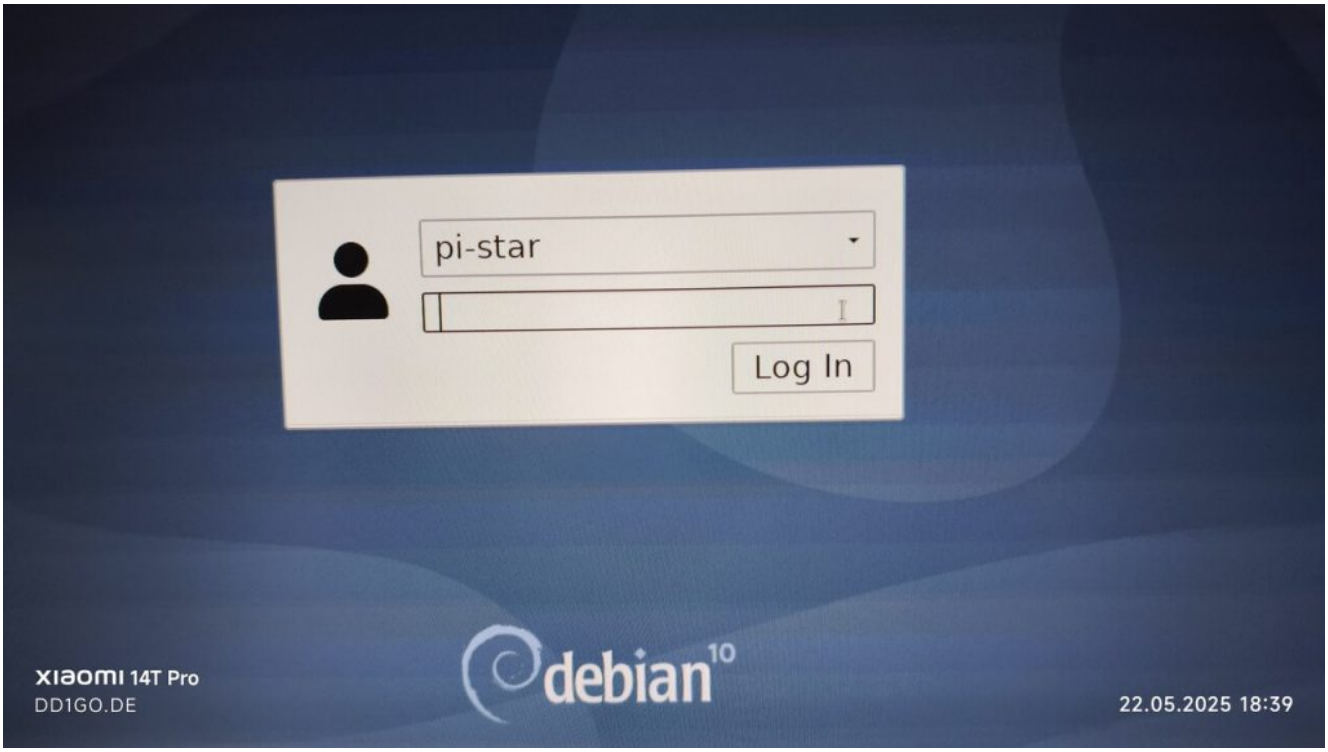
Für den Browser bieten es sich an was schmales zu wählen.

- sudo apt install qutebrowser

oder auch mit

- sudo apt-add-repository ppa:midori/ppa
- sudo apt-get update
- sudo apt-get install midori

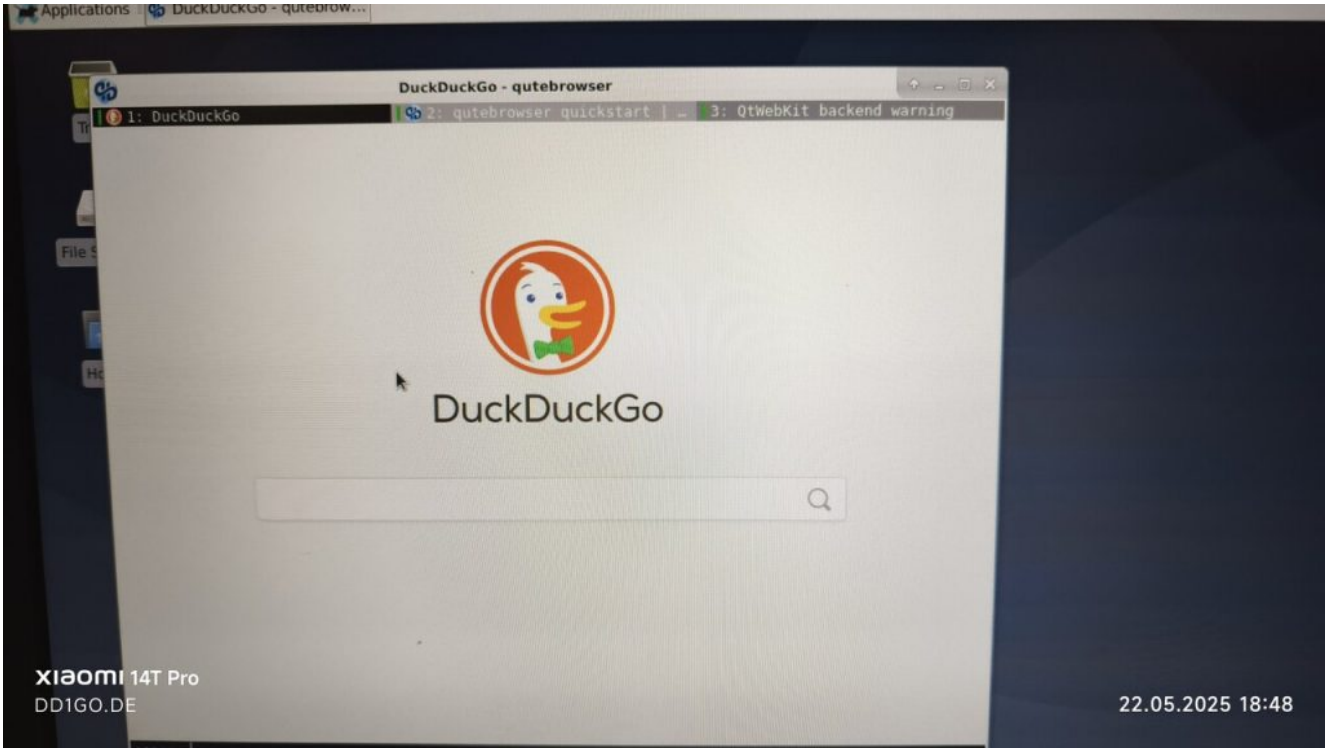
Im Anschluss ist der XFCE einsatzbereit und kann mit dem nächsten Neustart auch verwendet werden. Also Tastatur, Maus und Monitor dran und los geht es . Viel Spass



Nun kann, wie gewohnt, im Browser die Startseite angepasst

*werden und der Browser mit Autostart von **http://pistar/** direkt aufgerufen werden.*

Getestet auf Raspi Zero

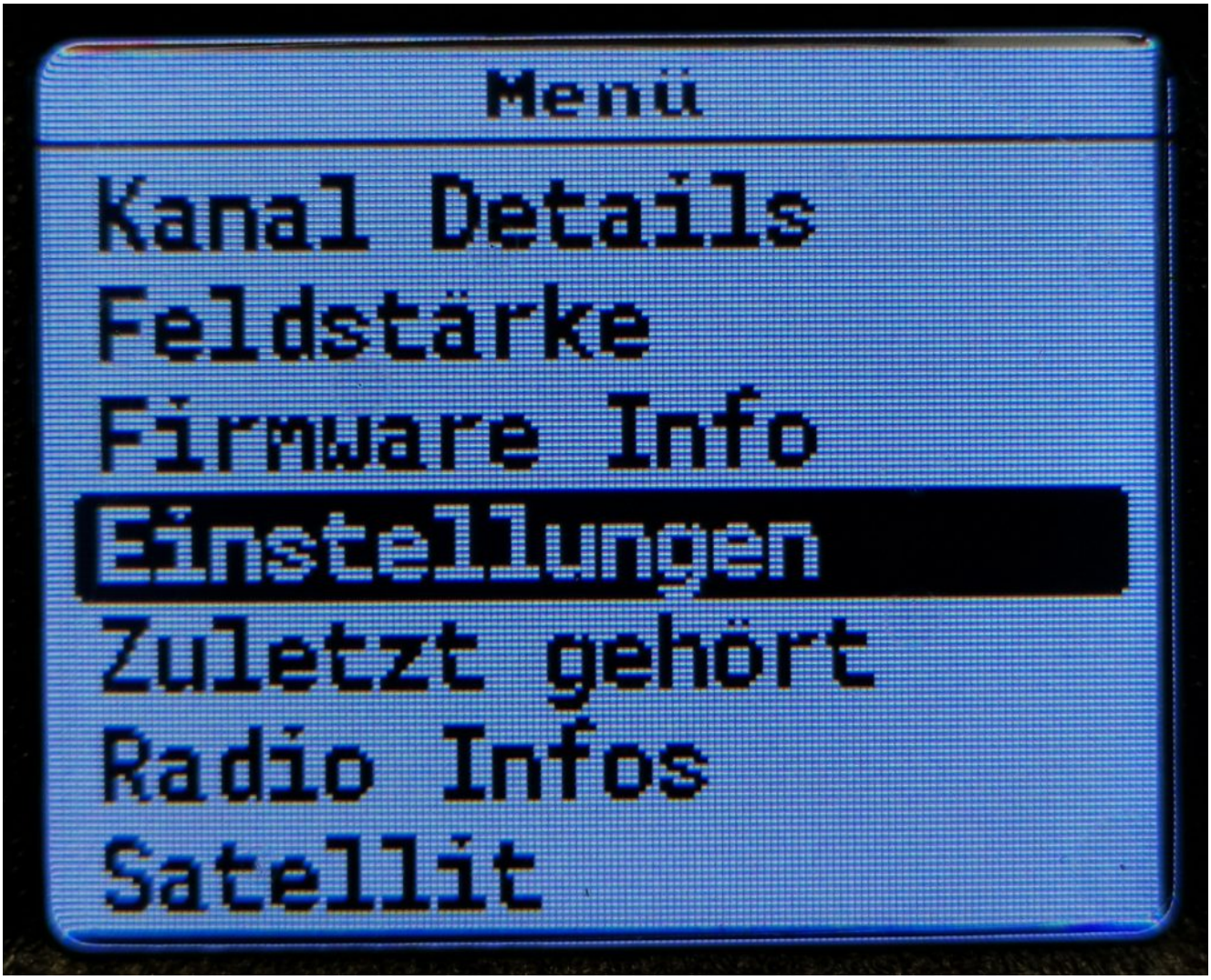


Retevis RT3s mit OpenGD77 als Hotspot

Wer mit OpenGD77 in Berührung kommt, findet schnell den Hinweis, dass das Funkgerät als Hostpot mit dem Pi-Star am Raspberry Pi verwendet werden kann.

Hierzu ein paar kleine Tipps:

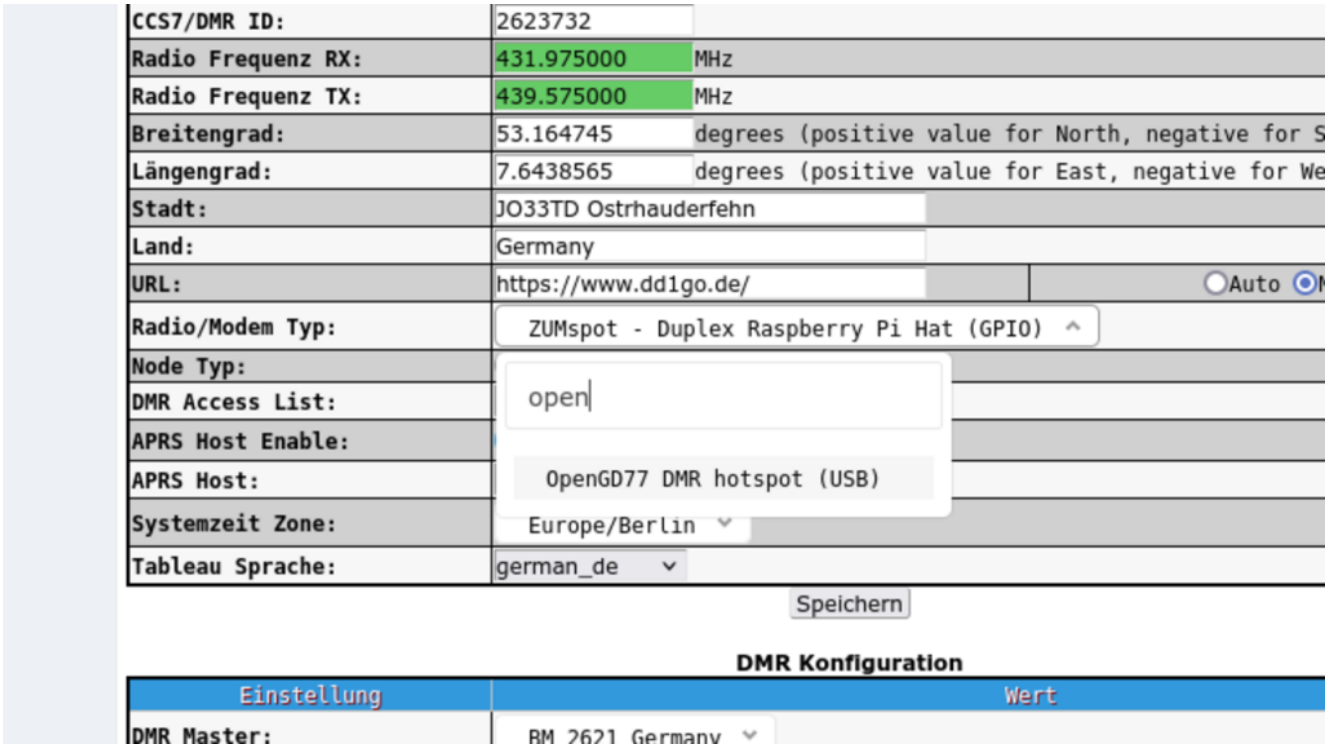
Nachdem die Firmware auf dem RT3s gespielt wurde, wird über den Menüpunkt >**Einstellungen** >**Einstellungen** >**HOTSPOT** von **AUS** auf **MMDVM** umgestellt.


Menü
Kanal Details
Feldstärke
Firmware Info
Einstellungen
Zuletzt gehört
Radio Infos
Satellit

Einstellungen
Kalibration
Theme Optionen
APRS Optionen
Einstellungen
Radio Optionen
Display Optionen
Audio Optionen

Einstellungen
Key lang:0.5s
Key wied:0.3s
Auto Lock:Aus
Hotspot:Aus
Temp.Kal: 0.0°C
Batt. Kal: 0.0V
ECO Stufe:1

Dann wird der Raspberry Pi mit dem aktuellen Pi-Star 4.2.x gestartet. Dort wird über Konfiguration der MMDVM Port (*ttyUSB0* oder *ttyAMA0* , je nach PiStar Version) und zudem das **Radio/Modem Typ** ausgewählt und ***OpenGD77 DMR hotspot (USB)*** ausgewählt (welches mit der Suche schnell gefunden wird).



Beachte bitte, dass das RT3s nur Simplex kann, somit auch der Typ des **Kontroller Mode** nur **SIMPLEX** ist. Andere Einstellungen können ungewollte Effekte haben. Die Einstellungen zu Talk Grupe und Time Slot werden wie gewohnt angegeben. Das RT3s springt automatisch in den Hotspot-Mode, sobald PiStar das Gerät erkannt hat. Bitte auch nur auf **kleine Leistung** (1 Watt) betreiben, da das RT3s nicht für Dauersenden ausgelegt ist; Bei 5 Watt kann das schnell ein „Verbrennen“ der Endstufe bedeuten.

Viel Erfolg

WLAN und Bluetooth mittels Boot-Config ausschalten

Manchmal ist es notwendig am Raspberry das WiFi und/oder Bluetooth zu deaktivieren um nur mit dem LAN arbeiten zu können ohne extra Netzwerke offen zu halten.

Dies lässt sich idealerweise mit dem BOOT Vorgang ein oder ausschalten. Hierzu benötigt man auch nicht viel Wissen. Dazu verbinden wir uns mittels Browser auf dem Pi-Star mit SSH (pi-star.local:2222) auf dem Minicomputer. Zuerst wird der Schreibschutz abgeschaltet. Hierzu tippen wir den folgenden Befehl ein und bestätigen mit Enter:

rpi-rw

Danach tippen wir den folgenden Befehl ein um die Konfiguration zu bearbeiten und bestätigen mit Enter:

sudo nano /boot/config.txt

In der nun dargestellten **config.txt** Datei scrollen wir bis ganz nach unten, bis wir den Textabschnitt mit der Überschrift „**[all]**“ sehen. An dessen Ende können wir nun unsere eigenen Konfigurationsbefehle eingeben, diese werden direkt beim Boot-Vorgang ausgeführt und verhindern damit, dass die WiFi-Verbindung überhaupt erst aufgebaut wird.

dtoverlay=disable-wifi

Falls wir zusätzlich auch noch Bluetooth deaktivieren möchten, so können wir dies indem wir noch die folgende Funktion in eine neue Zeile eingeben:

dtoverlay=disable-bt

Tragt gern eine **#Bemerkung** ein um zu dokumentieren , was und wann geändert wurde. Anschließend speichern wir die Datei mit der typischen ***CRTL + X*** Tastenkombination ab, beim Speicherdialog müssen wir – je nach Spracheinstellung – noch mit der Taste J bzw. Y und Enter bestätigen. Danach können wir den Raspberry Pi ganz normal neu starten über die Administration , das WLAN- bzw. auch das Bluetooth-Modul ist nun inaktiv. Der Internet- und Netzwerkzugriff kann dann ausschließlich mittels LAN oder externer Hardware erfolgen.

Falls ihr zu einem späteren Zeitpunkt die interne WiFi- bzw. Bluetooth-Funktionalität wieder aktivieren möchtet, entfernt

einfach die oben editierten Befehle aus eurer Konfigurationsdatei.

WPSD Digital Voice Dashboard

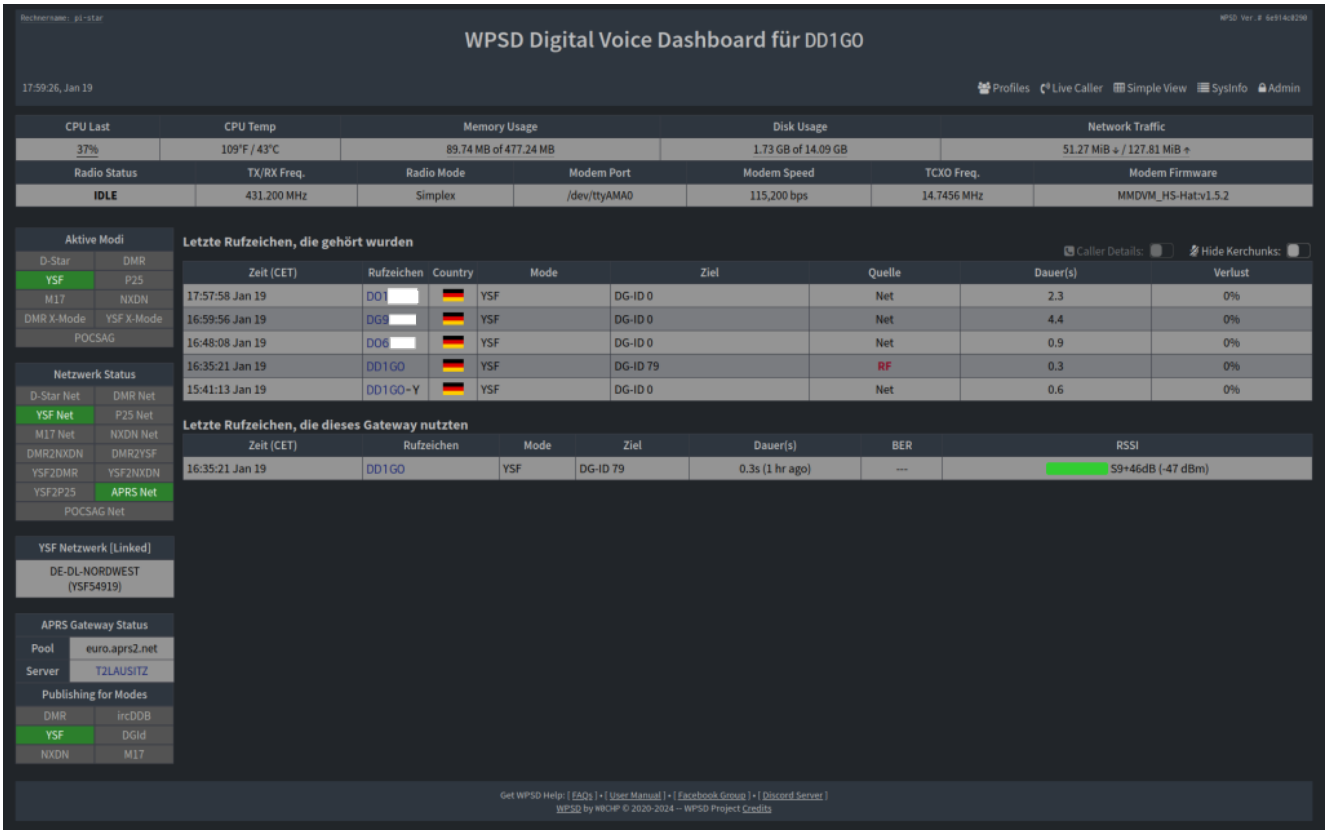
Hallo Pi Star Gemeinde, auf Wunsch habe ich den Link im Menü zum WPSD Projekt hinzugefügt.

Beachtet bitte, dass das Image nicht von Andy, dem Original Pi-Star Projekt stammt.

> *Willkommen auf der offiziellen Webseite des WPSD-Projekts. Screenshot des Haupt-Dashboards*
>
> *WPSD ist eine digitale Sprachsoftware-Suite der nächsten Generation für den Einsatz im Amateurfunk. Es wird sowohl für persönliche Hotspots als auch für Repeater verwendet. Es unterstützt die digitalen Sprachmodi M17, DMR, D-Star, Yaesu System Fusion (YSF/C4FM), P25, NXDN und POCSAG-Daten/Paging.*
>
> *WPSD begann als Ableitung („Fork“) der sehr beliebten Pi-Star-Software und hat sich stark von dieser abgewandelt. und WPSD hat sich zu einer eigenen Software-Suite und -Distribution entwickelt. Es ist als installierbares Disk-Image(s) verfügbar und mehrere Plattformen werden unterstützt. Das WPSD-Projekt ist freie und Open-Source-Software (FOSS).*
>
> *The WPSD Project Webseite*

Für dieses Projekt sollte minimum ein Raspberry ZERO 2 oder Raspberry 3 oder höher verwendet werden. Ein Raspberry 1 oder Zero 1 sind definitiv damit überfordert.

Hier im Test mit einem Raspberry Zero 1 W geht es teilweise bis 100% Auslastung der CPU.



Pi Star 4.1.7 zum neuen Jahr

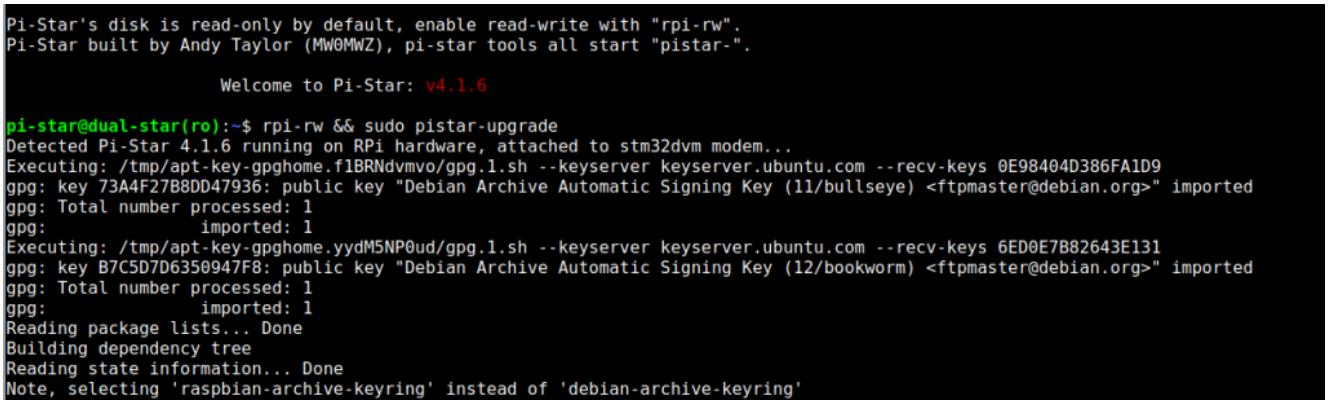
Pünktlich zum neuen Jahr erscheint am 05.01. die Pi Star Version 4.1.7. Dies soll dann weiter unter **Buster** laufen und löst die Offizielle ISO 4.1.5 und das Update 4.1.6 ab. Ein Update auf Bookworm , so der Entwickler Andy, kommt mit einer zukünftigen Version.

> *05-Jan-2024* ***Version 4.1.7*** *– Available for download and upgrade.*
> *It has beeen a while since I pushed a full update. Mostly that*
> *has been owing the to the very stable nature of the*
> *current build, however, it was getting to have some*

issues creep in.
What is new? – Actually nothing new, this update us just fixing
the expired keys for the apt repos. This caused the OS and other packages not to get some updates.
Upgrade to 4.1.7 from the dashboard (it's faster) and the new keys get installed, allowing all of the update goodness to function once more.
I know this is not terribly exciting, but there are other things
being worked on that are more exciting, more news on that later.

Andy Taylor (MW0MWZ)

Neues Gibt es nicht, dieses Update wird nur die abgelaufenen Schlüssel für die Apt-Repos reparieren. Dies verursachte, das das Betriebssystem und andere Pakete einige Updates nicht erhalten haben.



Aktualisieren Sie über das Dashboard auf 4.1.7 (es ist schneller) und die neuen Schlüssel werden installiert, sodass alle Updates möglich sind.
(Hierzu auf der SSH Seite im Expertenmodus „sudo pistar-update" verwenden)

Hinweis: Dies bedeutet jedoch auch , das ein UPDATEn der alten Versionen (vor V4.1.5) nur noch bedingt möglich ist oder für Profis auf der Konsole ausgeführt werden muss. Ein Hinweis beim Update mit „sudo pistar-update" auf

„Zeitüberschreitung auf dem GPG-Schlüsselserver“ kann in dem Fall übersehen werden, die 4.1.7 wird noch aktualisiert. Sollte jedoch ein KEY ERROR direkt mit Symbolen wie §0 oder ähnlichen abgebrochen werden, ist ein neues ISO erforderlich und die SD Karte muss neu geschrieben werden. Hier ist dann die Version zu alt, bzw nicht aktualisiert worden.

Bei neuer SD Karte, in dem Fall bitte erst unter ***Datensicherung*** die **CONFIG** speichern !

Raspberry Pi Zero und Router per USB LAN verbunden

Ja,dasgeht,sovielschonvorweg.Mit

vielenkleinen Mobilrouter ins Internet ist theute ke in Problem mehr. Nun kann

man diese über ein USB HOST Kabel (Bild Links: ***OTG USB Cauf Micro USB Kabel***

oder auch ***OTG USB MICRO-USB auf MICRO-USB Kabel*** sind verfügbar in den bekannte

n
Ve
rk
au
fs
hä
us
er
n)
so
ve
rb
in
de
n
,
da
s
es
pe
r
LA
N
An
ge
sp
ro
ch
en
wi
rd
.
So
mi
t
en
tf

ält
lt
au
ch
di
e
zu
sät
tz
li
ch
e
Wi
Fi
Ze
it
un
d
vo
r
al
le
m
is
t
es
mö
gl
ic
h
Mo
bi
le
En
dg
er
ät

e
im
5G
Hz
am
Ro
ut
er
zu
ve
rw
en
de
n,
da
ja
mi
tt
le
rw
ei
le
2.
4G
Hz
vo
ll
is
t.

```
pi-star@mobil-star(ro):~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.8.100  netmask 255.255.255.0  broadcast 192.168.8.255
        ether 0c:5b:8f:27:9a:64  txqueuelen 1000  (Ethernet)
        RX packets 259  bytes 71201 (69.5 KiB)
        RX errors 0  dropped 2  overruns 0  frame 0
        TX packets 191  bytes 24381 (23.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Der Raspberry Pi mit einem ***Rasbian*** (Debian) richtet dies automatisch ein. Es ist also nur das Kabel und der Router erforderlich. Manche Router können sogar die Stromversorgung liefern , so dass dann tatsächlich nur der Router und der RPi verbunden sind.

Und schon ist DIGITAL in der Hosentasche verstaut. Wer sich dann noch eine WSIM Karte einlegt, hat alle Netze (*Telekom, Telefónica, Vodafone & Drillisch 1&1 mit 2G, 4G oder 5G*) je nach Router in DE verfügbar oder kann in Europa in allen Netzen seine digitalen Funk nutzen.

SIM Karten für MMDVM

Ich hab bei meinen Hotspot mit Pi Star auch unterwegs per Mobilrouter Verbindungen, ohne WiFi in Hotel oder Pensionen nutzen zu müssen. Hier werde ich oft gefragt, was ich für SIM Karten verwende. O2, Vodafone oder Telekom. Tatsache hab ich in den meisten mobilen Routern (*Habe insgesamt 6 Mobil Router von Huawei mit Akku und 5V Versorgung sowie ein Speedport mit 12V Betrieb*) ein O2 Netz mit unlimitierten Datentransfer. Ich kann jedoch auch Weltweit mit einem Router arbeiten, der mit einer IoT-SIM-Karte (*diese SIM-Karten sind für den Einsatz im* ***Internet of Things*** *optimiert*) ausgerüstet ist. Das bedeutet auch in DL kann ich in allen drei Netzen wahlweise arbeiten, je nach Verfügbarkeit und Signal.

Diese ist Aktuell in über 150 Länder nutzbar ohne Extrakosten. Diese ist Speziell nur für meinen Digitalfunk in Betrieb und ist daher auch nur mit 500MB Volumen (was lang reicht, ist ja kein 24/7 Betrieb) ausgelegt und mit ca 6 Euro im Monat vergleichsweise „Billig“ auf dem Markt. Andere IoT Karten sind schnell bei 10 -20 Euro pro Monat. Es gibt auch Karten, die eine Einmalgebühr haben, doch diese haben auch dann nur 500 MB für die gesamte gebuchte Laufzeit (meist 5 oder 10 Jahre) die dann kein Sinn macht.

Hier ist der MMDVM (links) mit einem Huawei Router verbunden, der sogar das Modem mit Strom versorgen kann und so zeitgleich als LAN Netzwerk, also ohne WiFi Nutzung, verbunden ist.

Wer nur mal für kurze Zeit (ohne extra Kosten zu verursachen) mit einem Mobilen Router den MMDVM nutzen möchte, kommt sicher mit so einer 10 Euro für 10 Jahre Karte gut klar und kann bei bedarf auch mal nach buchen. Wer ständig unterwegs ist und immer wieder mit Netzproblemen kämpft , findet mit IoT Karten eine gute alternative zum kleinen Preis um in DE oder auch angrenzenden Nachbarländer alle verfügbaren Anbieter zu nutzen. Da reichen 500mb im Monat lange für ein paar Digitale Gespräche und Vielfunker können auch andere Volumen nutzen oder buchen. Man beachte nur , dass man keine UPDATES über den Mobilen Dienst machen sollte. Dann kann so ein Volumen schon mit einem Systemupdate weg sein. Dafür hat man ja immer noch das gute Heimnetzwerk und die Manuelle Funktion des Update und Upgrade starten. Dies gilt auch für LTE PoC Geräte, um die Netze zu nutzen die am besten vor Ort sind. IoT Karten sind in allen Netzen zuhause auch im 5G Netz und haben keine Internet

Einschränkungen, können optional sogar mit VPN betrieben werden.

Aber nun zum kleinen Nachteil bei IoT , das gibt es nur für Firmen und oft nur in größeren Stückzahlen an SIM Karten. Wer also nicht grade 10 oder mehr IoT SIM in der Firma verwendet , kommt nun nicht in den Genuss der günstigen Kartenvarianten.

Ich kann in dem Fall behilflich sein. Wer Interesse an so einer Karte hat, möge sich gern mit mir in Verbindung setzen.

Pi-Star:4.1.6 & Tableau 202204xx

In der aktuellen Version wurden viele Moden-Typen hinzugefügt und unter gewissen Umständen passiert es, dass euer voreingestelltes HAT nicht mehr da ist. Bitte prüft dann ob die Einstellung korrekt sind.

What kind of radio or modem hardware do you have?

„idrp2c“ = Icom Repeater Controller ID-RP2C (DStarRepeater Only)
„dvmpis“ = DV-Mega Raspberry Pi Hat (GPIO) – Single Band (70cm)
„dvmpid“ = DV-Mega Raspberry Pi Hat (GPIO) – Dual Band
„dvmuadu“ = DV-Mega on Arduino (USB – /dev/ttyUSB0) – Dual Band
„dvmuada“ = DV-Mega on Arduino (USB – /dev/ttyACM0) – Dual Band
„dvmuagmsku“ = DV-Mega on Arduino (USB – /dev/ttyUSB0) – GMSK

Modem
„dvmuagmska“ = DV-Mega on Arduino (USB – /dev/ttyACM0) – GMSK Modem
„dvmbss“ = DV-Mega on Bluestack (USB) – Single Band (70cm)
„dvmbsd“ = DV-Mega on Bluestack (USB) – Dual Band
„gmsk_modem“ = GMSK Modem (USB DStarRepeater Only)
„dvrptr1“ = DV-RPTR V1 (USB)
„dvrptr2“ = DV-RPTR V2 (USB)
„dvrptr3“ = DV-RPTR V3 (USB)
„dvap“ = DVAP (USB)
„zum“ = MMDVM / MMDVM_HS / Teensy / ZUM (USB)
„stm32dvm“ = STM32-DVM / MMDVM_HS – Raspberry Pi Hat (GPIO)
„stm32usb“ = STM32-DVM (USB)
„zumspotlibre“ = ZUMspot – Libre (USB)
„zumspotusb“ = ZUMspot – USB Stick
„zumspotgpio“ = ZUMspot – Single Band Raspberry Pi Hat (GPIO)
„zumspotdualgpio“ = ZUMspot – Dual Band Raspberry Pi Hat (GPIO)
„zumspotduplexgpio“ = ZUMspot – Duplex Raspberry Pi Hat (GPIO)
„zumradiopigpio“ = ZUM Radio-MMDVM for Pi (GPIO)
„zumradiopiusb“ = ZUM Radio-MMDVM-Nucleo (USB)
„mnnano-spot“ = MicroNode Nano-Spot (Built In)
„mnnano-teensy“ = MicroNode Teensy (USB)
„f4mgpio“ = MMDVM F4M-GPIO (GPIO)
„f4mf7m“ = MMDVM F4M/F7M (F0DEI) for USB
„mmdvmhsdualbandgpio“ = MMDVM_HS_Dual_Band for Pi (GPIO)
„mmdvmhshat“ = MMDVM_HS_Hat (DB9MAT & DF2ET) for Pi (GPIO)
„mmdvmhsdualhatgpio“ = MMDVM_HS_Dual_Hat (DB9MAT, DF2ET & DO7EN) for Pi (GPIO)
„mmdvmhsdualhatusb“ = MMDVM_HS_Dual_Hat (DB9MAT, DF2ET & DO7EN) for Pi (USB)
„mmdvmhshatambe“ = MMDVM_HS_AMBE (D2RG HS_AMBE) for Pi (GPIO)
„mmdvmrpthat“ = MMDVM_RPT_Hat (DB9MAT, DF2ET & F0DEI) for Pi (GPIO)
„mmdvmmdohat“ = MMDVM_HS_MDO Hat (BG3MDO) for Pi (GPIO)
„mmdvmvyehat“ = MMDVM_HS_NPi Hat (VR2VYE) for Nano Pi (GPIO)

„mmdvmvyehatdual“ = MMDVM_HS_Hat_Dual Hat (VR2VYE) for Pi (GPIO)
„genesyshat“ = Genesis – HHDVM_HS_Hat for Pi (GPIO)
„genesysdualhat“ = Genesis Dual – HHDVM_HS_Hat_Dual for Pi (GPIO)
„lshshatgpio“ = LoneStar – MMDVM_HS_Hat for Pi (GPIO)
„lshsdualhatgpio“ = LoneStar – MMDVM_HS_Dual_Hat for Pi (GPIO)
„lsusb“ = LoneStar – USB Stick
„sbhsdualbandgpio“ = SkyBridge – MMDVM_HS_Dual_Band for Pi (GPIO)
„nanodv“ = MMDVM_NANO_DV (BG4TGO) for NanoPi AIR (GPIO)
„nanodvusb“ = MMDVM_NANO_DV (BG4TGO) for NanoPi AIR (USB)
„opengd77“ = OpenGD77 DMR hotspot (USB)

PiStar 4.1.6 Quellcode

How to get TGIF Spot pi-star to post aprs location on aprs.fi in DRM mode

I got it working by Build TGIF Spot new from start.

Download and install pi-star on microSDHC
/downloads/

install using balena etcher. The raspberry Pi imager does not create boot

https://tgifnetwork.createaforum.com/nextion-displays/pi-star-nextion-script/

1.SSH INTO PI-STAR
2.DO A „rpi-rw“(ENTER)
3.DO A „pwd“(ENTER)
4.DO A „git clone https://github.com/EA7KDO/Scripts“(ENTER)
5.DO A „cd Scripts“(ENTER)
6.DO A ./IND.sh“(ENTER)
7.SELECT OPTION 1 (ENTER)
8.SELECT Y TO REBOOT (ENTER)

STEP 2
1. SSH INTO PI-STAR
2.DO A „rpi-rw“(ENTER)
3.DO A „cd Scripts“(ENTER)
4.DO A ./IND.sh“(ENTER)
5.SELECT OPTION 3 (CONTINUE) (ENTER)
6.NEXT SELECT OPTION 1 IDENTIFY YOUR INTERFACE (TTL_USB) (ENTER)
7.NEXT SELECT TEMP MODE #2 for FAHRENHEIT (ENTER)
AFTER SCRIPTS ARE DONE IT IS COMPLETE.

YOU CAN NOW DO YOUR NORMAL PI-STAR CONFIGURATIONS
CALL SIGN
DMR#
SIMPLEX FREQ
ETC, ETC

_.-._.-._

WI0K Pat Tracy

Pi Star 5 Dashboard

im Hintergrund wird schon an der neuen Version gearbeitet. Jedoch ist in den letzten zwei Jahren nicht wirklich was

passiert.