Terramaster F2-223/F2-423/F4-223/F4-423/T6-423 (NAS, 2-4 Cores, Non-ECC, 2-4x 3,5",2x M.2 2280 NVMe, 2x 2,5GBit)

Terramaster mit Proxmox VE, Proxmox Backup Server oder TrueNAS betreiben

Der **[GERÄTENAME]** lässt sich mit wenigen Änderungen als kompakter Server für **Proxmox VE, Proxmox Backup Server (PBS), Ultimate Backup Server oder TrueNAS** einsetzen.

Das System ist insbesondere für **Homelab, Virtualisierung, Backup, ZFS-Storage und kleinere Serverumgebungen** interessant. Nach dem Entfernen des vorhandenen USB-Sticks und dem Einbau geeigneter NVMe-SSDs lässt sich ein eigenes Betriebssystem grundsätzlich wie auf einem normalen Server oder PC installieren.

Hinweis: Das Gerät ist weiterhin erhältlich. Je nach Hardware-Revision, BIOS-Version, eingesetzten NVMe-SSDs und verwendetem Proxmox-Kernel können sich einzelne Schritte oder die Hardwareunterstützung unterscheiden.

Umbau für Proxmox VE, PBS und TrueNAS

Für den grundlegenden Umbau sind nur wenige Schritte notwendig:

1. Vorhandenen USB-Stick entfernen
2. Zwei geeignete M.2-NVMe-SSDs einbauen

3. Gewünschtes Betriebssystem installieren

Anschließend kann das System beispielsweise mit folgenden Plattformen betrieben werden:

- **Proxmox VE** für Virtualisierung mit KVM und LXC
- **Proxmox Backup Server** für Backup und Wiederherstellung
- **Ultimate Backup Server**
- **TrueNAS** für Storage- und ZFS-Anwendungen
- Debian oder andere Linux-Distributionen

Video zum Hardware-Umbau

ApfelCast zeigt den grundlegenden Umbau des Systems im folgenden Video. Der für den Umbau interessante Abschnitt beginnt ungefähr bei **6:38 Minuten**:

Umbau bei ApfelCast auf YouTube ansehen

Geeignete M.2-NVMe-SSDs

Bei der Auswahl der **M.2-NVMe-SSDs** sollte insbesondere die Temperaturentwicklung berücksichtigt werden. Aufgrund der kompakten Bauweise des Systems können sehr warme SSDs problematisch sein.

Nicht empfohlen: Micron 7450 Pro
Bei unseren Tests wurden diese SSDs im Gerät zu heiß. Wir empfehlen sie daher für diese konkrete Konfiguration nicht.

Mit **Samsung PM9A1** haben wir dagegen brauchbare Erfahrungen gemacht. Die Temperaturen blieben bei unseren Systemen in einem normalen Bereich.

Die Angaben beziehen sich auf unsere praktischen Erfahrungen mit den genannten Komponenten. Temperatur und Verhalten können von Gehäuse, Umgebungstemperatur, Firmware, Last und Hardware-Revision abhängen.

Lüftersteuerung unter Proxmox VE und Linux

Eine Besonderheit des Systems betrifft die **Lüftersteuerung unter Linux und Proxmox VE** .

Der Onboard-Controller für die Steuerung der verbauten Lüfter wird standardmäßig nicht vollständig erkannt. Mit einem angepassten **it87-Kernelmodul** kann der Controller jedoch angesprochen und das verbaute Lüfterpaar gesteuert werden.

Das verwendete Kernelmodul ist auf GitHub verfügbar:

bbqlinux/it87 auf GitHub

it87-Kernelmodul unter Proxmox installieren

Wichtig: Die benötigten Kernel-Header müssen zur tatsächlich installierten Proxmox-Kernelversion passen. Der nachfolgend genannte Paketname `proxmox-headers-6.5` stammt aus der ursprünglichen Installation und ist bei aktuellen Proxmox-Versionen entsprechend anzupassen.

```
# Benötigte Pakete installieren
apt install git dkms lm-sensors fancontrol

# Passende Header für den aktuell verwendeten Proxmox-Kernel installieren
apt install proxmox-headers-6.5

# Quellcode herunterladen
cd /opt
git clone https://github.com/bbqlinux/it87
cd it87

# Kernelmodul über DKMS bauen und installieren
make dkms

# Modul beim Systemstart automatisch laden
echo "it87" > /etc/modules-load.d/it87.conf

# Hardware-Sensoren erkennen
```

```
sensors-detect
```

Den Assistenten von `sensors-detect` anschließend durchlaufen lassen. In unserer Konfiguration konnten die vorgeschlagenen Standardwerte verwendet werden.

Temperaturen und Lüfter mit lm-sensors prüfen

Nach erfolgreicher Installation zeigt der Befehl:

```
sensors
```

die erkannten Hardware-Sensoren an. Mit dem angepassten Kernelmodul sollte zusätzlich der Controller `it8613-isa-0a30` erscheinen.

```
root@px2:/mnt/opt/it87# sensors
it8613-isa-0a30
Adapter: ISA adapter
in0:           1.72 V  (min =  +0.00 V, max =  +2.81 V)
in1:           1.23 V  (min =  +0.00 V, max =  +2.81 V)
in2:           2.06 V  (min =  +0.00 V, max =  +2.81 V)
in4:           2.05 V  (min =  +0.00 V, max =  +2.81 V)
in5:           2.02 V  (min =  +0.00 V, max =  +2.81 V)
3VSB:          1.21 V  (min =  +0.00 V, max =  +5.61 V)
Vbat:          3.06 V
+3.3V:         3.34 V
fan2:             0 RPM  (min =    0 RPM)
fan3:          1670 RPM  (min =    0 RPM)
fan4:             0 RPM  (min =    0 RPM)
fan5:             0 RPM  (min =    0 RPM)
temp1:        +34.0°C  (low  = -128.0°C, high = +127.0°C)  sensor = thermal diode
temp2:        +29.0°C  (low  = -128.0°C, high = +127.0°C)  sensor = thermal diode
temp3:        +40.0°C  (low  = -128.0°C, high = +127.0°C)
intrusion0:  ALARM

nvme-pci-0300
Adapter: PCI adapter
Composite:    +32.9°C  (low  = -273.1°C, high = +80.8°C)
                       (crit = +84.8°C)
Sensor 1:     +32.9°C  (low  = -273.1°C, high = +65261.8°C)
Sensor 2:     +31.9°C  (low  = -273.1°C, high = +65261.8°C)

acpitz-acpi-0
Adapter: ACPI interface
temp1:        +27.8°C  (crit = +119.0°C)

coretemp-isa-0000
Adapter: ISA adapter
Package id 0:  +44.0°C  (high = +105.0°C, crit = +105.0°C)
Core 0:        +40.0°C  (high = +105.0°C, crit = +105.0°C)
Core 1:        +40.0°C  (high = +105.0°C, crit = +105.0°C)
Core 2:        +40.0°C  (high = +105.0°C, crit = +105.0°C)
Core 3:        +40.0°C  (high = +105.0°C, crit = +105.0°C)

nvme-pci-0400
Adapter: PCI adapter
Composite:    +36.9°C  (low  = -273.1°C, high = +80.8°C)
                       (crit = +84.8°C)
Sensor 1:     +36.9°C  (low  = -273.1°C, high = +65261.8°C)
Sensor 2:     +34.9°C  (low  = -273.1°C, high = +65261.8°C)

root@px2:/mnt/opt/it87#
```

In unserer Konfiguration zeigt `fan3` die Drehzahl des verbauten Lüfterpaares an. Dieser Ausgang wird anschließend für die automatische Lüftersteuerung verwendet.

Automatische Lüftersteuerung mit pwmconfig und fancontrol

Mit `pwmconfig` kann eine passende Konfiguration für **fancontrol** erstellt werden:

```
pwmconfig
```

Das geführte Skript prüft die verfügbaren Lüfterausgänge und ermöglicht die Zuordnung eines Temperatursensors. Anschließend können die gewünschten Temperatur- und Drehzahlschwellen definiert werden.

Die fertige Konfiguration wird üblicherweise unter:

```
/etc/fancontrol
```

gespeichert.

Nach Änderungen kann die Lüftersteuerung neu gestartet werden:

```
systemctl restart fancontrol.service
```

Der Status lässt sich anschließend kontrollieren:

```
systemctl status fancontrol.service
```

Einsatzmöglichkeiten als kompakter Proxmox-Server

Nach dem Umbau bietet das System verschiedene interessante Einsatzmöglichkeiten. Mit zwei NVMe-SSDs lässt sich beispielsweise ein kleiner **Proxmox VE Host** mit redundantem ZFS-Mirror aufbauen.

Alternativ eignet sich die Plattform für einen kompakten **Proxmox Backup Server**, einen Linux-Server oder – abhängig von der gewünschten Storage-Konfiguration – für **TrueNAS und ZFS**.

Damit ist das Gerät insbesondere für Anwender interessant, die einen **kompakten und flexiblen Server für Virtualisierung, Backup, Homelab oder Open-Source-Storage** suchen und bereit sind, die Hardware entsprechend anzupassen.

Technischer Hinweis: Diese Dokumentation basiert auf unseren praktischen Erfahrungen mit dem Gerät. Insbesondere Kernelmodule, Proxmox-Kernel, Paketnamen und Hardwareunterstützung können sich mit neueren Softwareversionen ändern. Prüft daher vor der Installation insbesondere die aktuell verwendete Kernelversion und installiert die dazu passenden Header-Pakete.

Version #8
Erstellt: 2024-01-26 18:22:21 UTC von Thorsten Spille
Zuletzt aktualisiert: 2026-07-28 14:07:13 UTC von Christian Zengel (sysops GmbH)