
Release Notes

Bookworm 10/2024

REVOLUTION PI
a **KUNBUS** brand

2024-10-09

Release Notes Bookworm 10/2024

Die RevPi Bookworm Images basieren auf Debian Bookworm und integrieren bestimmte Pakete aus dem Raspberry Pi OS.

Die Images enthalten alle Paket-Updates, die bis 2024-10-08 in den Debian, Raspberry Pi OS und Revolution Pi Paketquellen veröffentlicht wurden. Es verwendet Kernel 6.6.46 mit dem RT Patch in der Version 39.

Veränderungen ab Bookworm

- Die *RevPi Bookworm Images* basieren auf *Debian Bookworm* und *Raspberry Pi OS Bookworm*.
- Die *RevPi Bookworm Images* werden nur noch als 64-Bit Version angeboten. Es wird keine 32-Bit (armhf) Varianten mehr geben.
- Ein Upgrade von *Bullseye* wird **nicht** unterstützt.
 - Die Bootpartition wird nun unter `/boot/firmware` eingebunden.
 - Die `config.txt`, die Devicetree- und Overlaydateien sind jetzt in `/boot/firmware` zu finden
- *RevPi Status* wird durch *Cockpit* ersetzt.
 - Der Login in *Cockpit* erfolgt über den User `pi` nicht über den user `admin`.
 - *PiCtory* ist nun über den Menüeintrag “RevPi Konfiguration” in *Cockpit* zu erreichen.
- Die Zugriffsrechte des User `pi` wurden eingeschränkt.
 - Es gibt nun verschiedene Benutzergruppen, um die Zugriffsrechte feingranular zu konfigurieren.

Ein Upgrade von Bullseye auf Bookworm wird, wie auch bei dem Raspberry Pi, nicht unterstützt!
Die Images müssen neu auf die Geräte aufgespielt werden.

Bekannte Probleme (Known issues)

- Konfiguration des Keyboard-Layouts über `raspi-config` ist nur möglich, wenn vorher das Paket `console-setup` installiert wird: `sudo apt install console-setup`.
- Logeinträge von `/etc/udev/rules.d/99-com.rules:7 Unknown group 'X', ignoring` mit dem Wert `gpio, spi` und `i2c`.
- WLAN auf dem RevPi Flat ist nach einem Warmstart derzeit nicht nutzbar. Es wurde festgestellt, dass das Problem mit einer älteren Firmwareversion umgangen werden kann. Ein Fix, der dieses

Problem behebt, wird zur Verfügung gestellt und kann über die Aktualisierung in Cockpit oder `sudo apt update && sudo apt upgrade` installiert werden.

- Um ein Problem beim Bluetooth auf dem RevPi Flat S zu beheben, muss das Paket `revpi-bluetooth` aktualisiert werden. Die Version 1.1.0 behebt das Problem.

Package versions

Eine genaue Liste aller Änderungen kann dem Changelog des jeweiligen Paketes entnommen werden (z.B.: `/usr/share/doc/PAKETNAME/changelog.Debian.gz`).

cockpit-revpi 1.2.0-1+deb12+1

Cockpit Plugin für die Konfiguration des Revolution Pi. Dieses Paket ersetzt `revpi-webstatus`. Als Abhängigkeit wird das gesamte Cockpit-System, inkl. ausgewählter Plugins, installiert. Um Cockpit zu verwenden, muss dieses Paket über `cockpit-revpi-apache` installiert werden.

Weitere Informationen: Cockpit ersetzt RevPi Status

cockpit-revpi-apache 1.2.0-1+deb12+1

Integriert `cockpit-revpi` in den Apache Webserver auf Port 41443 unter `/cockpit-revpi/` und installiert das `cockpit-revpi` Paket als Abhängigkeit auf dem System.

Weitere Informationen: Cockpit ersetzt RevPi Status

cockpit-revpi-redirect-apache 1.2.0-1+deb12+1

Ermöglicht den Aufruf von Cockpit ohne Angabe des Ports. Für eigene Webanwendungen auf dem Revolution Pi System kann das Paket deinstalliert werden, damit werden die Standardports 80 und 443 frei verwendbar: `apt remove cockpit-revpi-redirect-apache`

Cockpit bleibt über Port 41443 erreichbar.

Weitere Informationen: Cockpit ersetzt RevPi Status

Linux Kernel (`linux-image-revpi-v8`) 6.6.46-rt39-revpi2-1+deb12+1

Bis Bullseye wurden der Linux Kernel und piControl in einem Paket unter dem Namen `raspberrypi-kernel` gebaut. Das ändert sich mit Bookworm. Das neue Linux Kernel Paket heißt `linux-image-revpi-v8` und wird anders als das bisherige Kernel Paket gebaut. piControl hat ein eigenes Paket mit den Namen `picontrol`.

Als Linux Kernel benutzen wir die Stable Version 6.6.

Es wird nun außerdem ein eigenes device tree overlay zum Aktivieren des DSA-features auf dem RevPi Flat S angeboten.

mqtt-revpi-client 0.1.0-1+revpi12+2

Mit diesem Paket wird das virtuelle MQTT Client Device in den Gerätekatalog von PiCtory installiert und steht für die Benutzung bereit. Mit diesem Device können alle IOs über einen MQTT Broker versendet und empfangen werden.

opcua-revpi-server 0.4.0-1+revpi12+2

Mit diesem Paket wird das virtuelle OPC UA Server Device in den Gerätekatalog von PiCtory installiert und steht für die Benutzung bereit. Mit diesem Device können alle IOs über OPC UA direkt angesprochen werden.

picontrol 2.2.1-1+deb12+1

piControl ist nun, wie der Linux Kernel, auch nicht mehr Teil des `raspberrypi-kernel` Pakets sondern existiert als separates Paket mit dem Namen `picontrol`.

Durch Optimierungen am UART Treiber konnte die von piControl beanspruchte Prozessorlast signifikant reduziert werden.

Es wurde außerdem ein Bug gefixt, bei dem es zu vermehrten Fehlern auf der PiBridge kam. Dies trifft vor allem (aber nicht ausschließlich) auf den Fall zu, dass gleichzeitig noch Daten über eine separate RS-485 Verbindung verschickt wurden.

Das Prozessabbild unter `/dev/piControl0` konnte bisher von allen Nutzern gelesen und beschrieben werden. Mit Bookworm ist das nur noch für Nutzer in der Gruppe `picontrol` möglich. Der Nutzer `pi` gehört standardmäßig dieser Gruppe an.

Wenn andere Nutzer auf dem System Zugriff auf das Prozessabbild haben sollen, dann müssen diese

der Gruppe `picontrol` hinzugefügt werden. Das kann bspw. mit folgendem Befehl geschehen:
`sudo adduser $USER picontrol`.

Zugriff auf `/dev/piControl0` ist nur als Mitglied der Gruppe `picontrol` möglich

picturey-apache 2.10.0-1+deb12+1

Integriert PiCtory in den Apache Webserver auf Port 41443 unter `/picturey/` und installiert das `picturey` Paket als Abhängigkeit auf dem System.

picturey 2.10.0-1+deb12+1

Enthält keine Webserverkonfigurationen mehr. Um die Webanwendung zu verwenden, muss dieses Paket über `picturey-apache` installiert werden.

python3-revpimodio2 2.7.2-1+deb12+1

- Behebt einen Fehler, bei dem auf einem Connect 4 das Relais nicht angesteuert werden konnte.
- Die Farben der LEDs über `.core.A1 - .A5` auf einem Connect 4 werden nun gemäß der Konstanten GREEN, RED und BLUE geschaltet.

Ab den RevPi Bookworm Images muss der Benutzer, welcher den Prozess ausführt, Mitglied der Gruppe `picontrol` sein. Bei einem PermissionDenied-Fehler sollte dies im ersten Schritt der Fehleruntersuchung überprüft werden.

revpi-base-files 0.4.1-1+revpi12+1

Das Paket installiert Basisdateien, die für den minimalen Betrieb eines Revolution Pi Geräts erforderlich sind.

revpi-cert-wizard 1:2.4.5-1+revpi12+4 + revpi-tmpfs-logs 1:2.4.5-1+revpi12+4

Die Funktionen zur Erstellung von TLS Zertifikaten sowie zum Speichern von Logs in einem tmpfs (im RAM) waren Bestandteil von `revpi-webstatus`. Durch das Ersetzen von RevPi Status mit Cockpit wurden diese beiden Komponenten als eigenständige Pakete veröffentlicht.

revpi-nodered-proxy-apache 1.5.1-1+deb12+1

Integriert Node-RED in den Apache Webserver auf Port 41880 und installiert das `revpi-nodered` Paket. `revpi-nodered` installiert die komplette Node-RED Umgebung mit den RevPi Nodes, welche sonst Teil vom Paket `node-red-contrib-revpi-nodes` waren.

Um `revpi-nodered` in ein Lite Image zu installieren, muss `revpi-nodered-proxy-apache` installiert werden, da es sonst nur über `localhost` verfügbar ist. Alternativ kann es über Cockpit im Menüpunkt "RevPi Konfiguration" installiert werden.

Weitere Informationen: Node-RED als eigenes Paket

revpi-tools 4.2.2-1+revpi12+1

In diesem Paket befinden sich weiterhin Tools für den Revolution Pi, jedoch keine Basisdateien, die für den minimalen Betrieb benötigt werden. Diese werden nun über das Paket `revpi-base-files` installiert.

revpi-ui 0.1.0-1+revpi12+1

Installiert alle Abhängigkeiten für die Nutzung der grafischen Benutzerschnittstelle (GUI). Kann in lite Images zur nachträglichen Installation der GUI verwendet werden.

```
1 sudo apt install revpi-ui
```

revpi-webstatus

Dieses Paket existiert nicht mehr. Die Funktionen wurden in folgende Pakete ausgelagert:

- `revpi-webserver-apache`
- `cockpit-revpi-apache`
- `pictory-apache`
- `revpi-cert-wizard`
- `revpi-tmpfs-logs`

revpi-webserver-apache 0.2.0-1+revpi12+2

Grundgerüst für Webserverkonfigurationen aller RevPi Webanwendungen am Port 41443. Pakete wie PiCtory, Cockpit, Node-RED erweitern diese Konfiguration und sind sofort nach Installation über das

Webinterface erreichbar. Kein `systemctl reload apache2` mehr nötig.

Weitere Informationen

Cockpit ersetzt RevPi Status

Ab den RevPi Bookworm Images wird die Basiskonfiguration mit RevPi Status durch Cockpit ersetzt. Die neue Oberfläche bietet, neben den Einstellmöglichkeiten aus RevPi Status, diverse weitere Möglichkeiten das gesamte System zu verwalten.

Cockpit ist eine Web-Applikation zur Konfiguration von Linux-Servern. Cockpit bietet eine anwenderfreundliche grafische Oberfläche, die es authentifizierten Benutzern ermöglicht, über eine sichere Verbindung beliebige Konfigurationen und Befehle auf einem Gerät auszuführen. Netzwerkkonfiguration, Benutzerverwaltung, Status- und Log-Ansichten können über einen Browser verwaltet werden. Cockpit, auch Web Konsole genannt, ist ein von Red Hat gesponsertes Open-Source-Softwareprojekt, das unter der GNU Lesser General Public License (LGPL) veröffentlicht wird.

Cockpit basiert auf einem modularen Prinzip und kann um Plugins erweitert werden. Das Paket `cockpit-revpi` ist ein Cockpit-Plugin von Revolution Pi. Das Plugin ermöglicht die Basiskonfiguration der RevPi Geräte und ein einfaches Aktivieren bzw. Deaktivieren der installierten Dienste.

WICHTIG: Für die Anmeldung wird nun der Benutzer und das Gerätepasswort auf dem Aufkleber an der Gehäusesseite des RevPi Basismoduls verwendet. Der Standardbenutzer ist "pi" (ehemals "admin").

Aktuelle Basismodule, die bereits mit den Bookwork Images ausgeliefert werden, haben den Namen des neuen Standardbenutzers schon auf dem Gehäuse aufgedruckt ("pi" statt "admin").

Basiskonfiguration mit Cockpit Cockpit wird über einen Webbrowser mit `https://revpi[Seriennummer].local:41443` aufgerufen. Die Anmeldung erfolgt mit Benutzernamen `pi` und dem Gerätepasswort.

In Cockpit muss der Modus von *Limited access* auf *Administrative access* gewechselt werden um diverse Einstellungen ändern zu können.

Über das Menü *RevPi Configuration* können die Einstellungen für die Basiskonfiguration wie zuvor im RevPi Status vorgenommen werden.



Figure 1: cockpit_access_button_de.png

Ab den RevPi Bookworm Images wird auch PiCtory über Cockpit gestartet. Über PiCtory erfolgt die Konfiguration des Revolution Pi Systems, also eines RevPi Basismoduls mit Erweiterungsmodulen.

Wurde die lite-Version der RevPi Images installiert, können an dieser Stelle fehlende Pakete wie Node-RED nachinstalliert werden.

Alle Einstellungen werden beim Setzen sofort übernommen.

Software-Updates

Im Cockpit-Menü *Software updates* können alle installierten Softwarepakete auf dem Revolution Pi System aktualisiert werden.

Netzwerkinformationen

Im Cockpit-Menü *Networking* können Netzwerkinformationen abgefragt und z.B. eine feste IP-Adresse für die Schnittstellen gesetzt werden.

Das Hinzufügen vom RevPi zu einem WLAN ist hier nicht möglich. Für diesen Zweck kann die Web Konsole und `sudo nmtui` im Browser verwendet werden.

Wird die IP-Adresse geändert, muss Cockpit neu über den Webbrowser gestartet werden.

Log-Protokolle und SOS-Report

Im Cockpit-Menü *Logs* können die Protokolle zur Fehlersuche gefiltert und eingesehen werden.

Im Cockpit-Menü *Diagnostic reports* kann ein SOS-Report zur Fehlerdiagnose erstellt und exportiert werden.

Integriertes Terminal

Über das Cockpit-Menü *Terminal* öffnet sich ein integriertes, voll funktionsfähiges Terminal, über das direkt Befehle eingegeben werden können.

SSH-Verbindung

Es besteht auch mit den RevPi Bookworm Images die Möglichkeit, das RevPi Basismodul über eine SSH-Verbindung zu erreichen.

Node-RED als eigenes Paket

Der Node-RED Server wird über das Paket `revpi-nodered` im System installiert. In diesem Paket sind auch die `node-red-contrib-revpi-nodes` in der aktuellsten Version enthalten und werden automatisch, wie auch Node-RED, über Paketupdates aktualisiert. In Images vor Bookworm wurden diese über ein separates Paket installiert, was nun nicht mehr existiert.

Node-RED läuft auf dem Revolution Pi System als systemd-Dienst und startet aus Sicherheitsgründen nur mit Zugriffsmöglichkeiten über `localhost` auf Port 1881. Damit Node-RED auch remote über eine verschlüsselte Verbindung erreichbar ist, installiert das vorinstallierte Paket `revpi-nodered-proxy-apache` eine Proxy-Konfiguration für den Apache Webserver. Node-RED ist remote über die Adresse des Revolution Pi's auf dem Port 41880 erreichbar.

Der Node-RED Prozess wird als Systembenutzer `nodered` ausgeführt. Alle Einstellungen und Benutzerdaten befinden sich im Ordner `/var/lib/revpi-nodered`. Dieser Pfad ist auch der einzige Pfad, in den der Benutzer Daten schreiben kann. Das restliche Dateisystem ist lediglich lesend verfügbar.

Berechtigungen Soll über Node-RED auf Geräte im `/dev` Verzeichnis zugegriffen werden, wie z.B. RS-485, muss der Benutzer `nodered` in die jeweiligen Gruppen hinzugefügt werden, welche schreibenden Zugriff auf die Geräte haben.

Ein Beispiel ist `/dev/ttyRS485-0`, welches z.B. für Modbus verwendet werden kann. Der Benutzer `nodered` kann entweder über SSH oder über das Terminal im Cockpit mit folgendem Befehl Zugriffsrechte erhalten:

```
1 sudo adduser nodered dialout
```

Der Zugriff auf die IOs des Revolution Pi mit den integrierten Knoten läuft über den `noderedrevpinodes-server`, für den keine weiteren Berechtigungen gesetzt werden müssen.

Standardverhalten herstellen Sollte es aus bestimmten Gründen nötig sein, dass Node-RED über den Standardweg aus der Ferne erreichbar sein muss, können folgende Änderungen am System vorgenommen werden. Diese Änderungen werden von uns nicht unterstützt und sollten nur von erfahrenen Benutzer durchgeführt werden.

- Deinstallation der Webkonfiguration

```
1 sudo apt purge revpi-nodered-proxy-apache
```

- Überschreibung der existierenden systemd Datei anlegen (nicht die originale Datei verändern, da diese bei einem Update wiederhergestellt werden würde)

```
1 sudo mkdir -p /etc/systemd/system/nodered.service.d
2 cat | sudo tee /etc/systemd/system/nodered.service.d/custom-override.
  conf <<"__END__"
3 [Service]
4 # Clear existing configuration
5 ExecStart=
6 # Use start command without bind ip and port
7 ExecStart=/usr/bin/env /usr/share/revpi-nodered/node_modules/node-red/
  bin/node-red-pi $NODE_OPTIONS $NODE_RED_OPTIONS -D uiPort=1880 -D
  uiHost=0.0.0.0
8 __END__
```

CODESYS

- Es muss mindestens “CODESYS Control Linux ARM 64 (v4.10.0.0)” verwendet werden.
- Die RS-485 Schnittstellen haben nun fortlaufende Nummern `/dev/ttyRS485-0`, `/dev/ttyRS485-1`.

Für die Verwendung der RS-485 Schnittstellen muss folgender Eintrag in `/etc/CODESYSControl.cfg` ergänzt werden:

```
1 [SysCom]
2 Linux.Devicefile =/dev/ttyRS485-
```

Die Zeile schließt mit einem Bindestrich ab, dies ist kein Tippfehler!

Daraus ergeben sich in CODESYS folgende Zuweisungen:

- `/dev/ttyRS485-0` -> COM Port 1
- `/dev/ttyRS485-1` -> COM Port 2 (Flat S)

Das Device `/dev/ttyRS485` ist obsolet. Für die Kompatibilität zu Bullseye existiert diese noch und wird mit dem nächsten Debian Release vollständig entfernt.