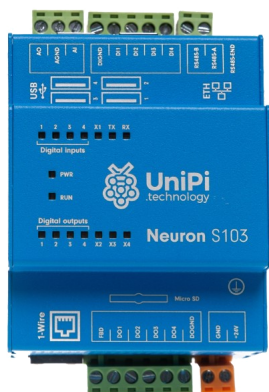


Inhaltsverzeichnis

Unterstützte Geräte	1
Installation	1
Anforderungen	1
Konfiguration	2
Channel	2
Variablen	3
Fehlerdiagnose	3
Kanal	3
Variablen	4
Logdatei	4
Entities	5
Device	5
Channel	5
Variable	5
Migration von Version 1.0.0 auf 1.0.1	6
Ordner & Dateien	6
Ordner	6
Dateien	6
Versionsinformation	6
Dieses Dokument	6
Plugin	6
Assembly	7

UniPi Device Plugin



Das UniPi Device Plugin ist für den Einsatz auf einem [UniPi Gerät](#) konzipiert.

Es ermöglicht die Konfiguration des Geräts und das Lesen und Schreiben von Daten.

Unterstützte Geräte

- [UniPi Neuron Serie](#)
- [UniPi Neuron Extension Serie](#)

Installation

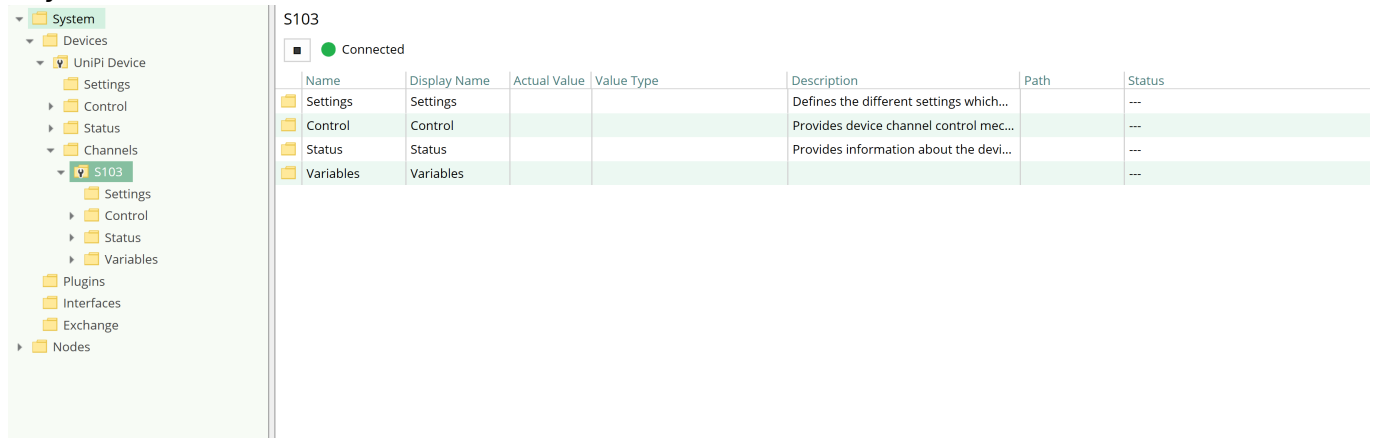
Dieses Plugin ist Bestandteil des CoDaBix® für Raspberry Pi Setups. Bitte konsultieren Sie [CoDaBix® Setup und erster Start](#) für weitere Informationen darüber, wie dieses Plugin installiert und deinstalliert werden kann.

Anforderungen

- UniPi Neuron Gerät
- SD Card mit aktuellem UniPian Neuron OS Image ([Download](#))
- [Standardanforderungen von CoDaBix®](#)

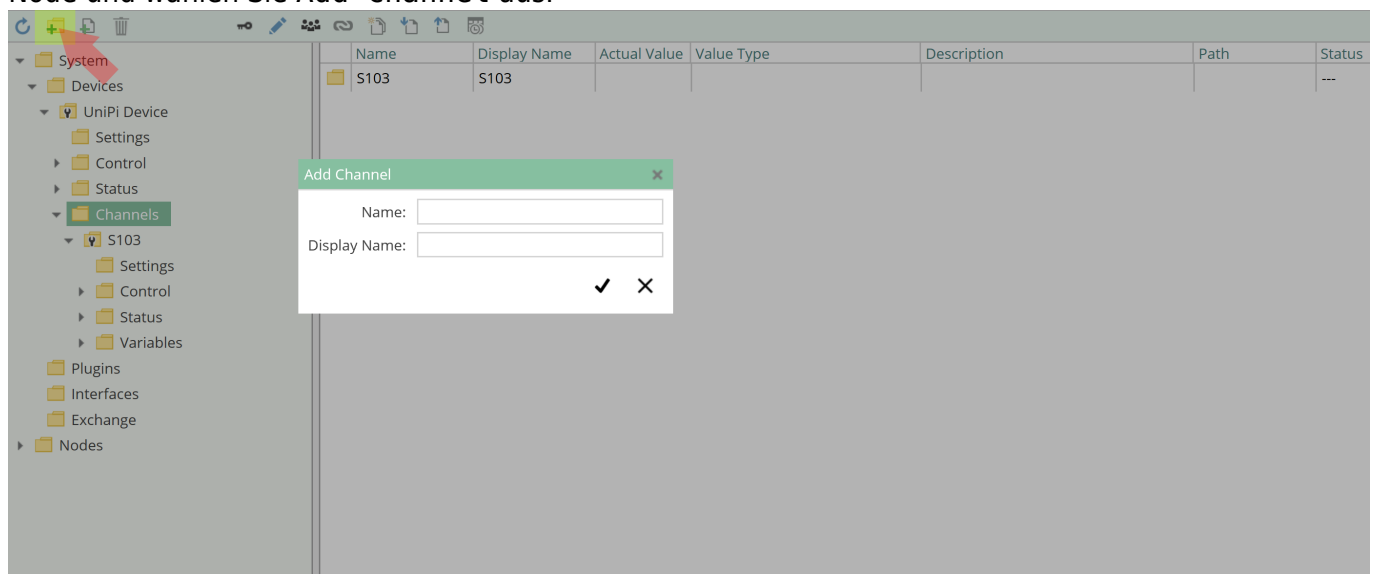
Konfiguration

Die gesamte UniPi Device Plugin-Konfiguration befindet sich unter dem Nodepfad /System/Device/UniPi Device.



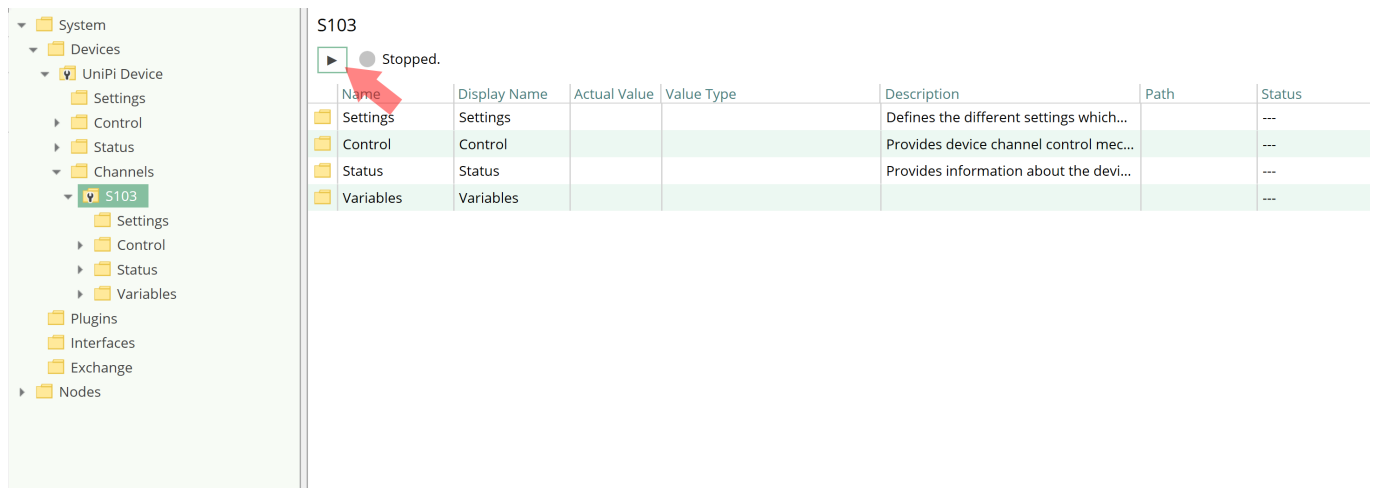
Name	Display Name	Actual Value	Value Type	Description	Path	Status
Settings	Settings			Defines the different settings which...		---
Control	Control			Provides device channel control mec...		---
Status	Status			Provides information about the devi...		---
Variables	Variables					---

Der Nodebaum im oberen Bild zeigt den Standardnodebaum des UniPi Device Plugins. Um einen neuen UniPi-Kanal aufzusetzen, fügen Sie einen Folder Node unter dem Node UniPi Device/Channels hinzu, oder machen Sie einen Rechtsklick auf den UniPi Device/Channels-Node und wählen Sie Add Channel aus.



Channel

Für den Kanal sind keine weiteren Einstellungen notwendig. Nachdem Sie „Save“ geklickt haben, wird die Channel-Node erstellt. Sie können den Channel starten, indem Sie diese Node auswählen und den Startbutton klicken:



Name	Display Name	Actual Value	Value Type	Description	Path	Status
Settings	Settings			Defines the different settings which...		---
Control	Control			Provides device channel control mec...		---
Status	Status			Provides information about the devi...		---
Variables	Variables					---

Variablen

Um die Einrichtung der UniPi-Variablen zu vereinfachen, stellen wir für das UniPi-Modell S103 eine [Standard Channel-Konfiguration](#) bereit, in der bereits die wichtigsten Variablen des UniPi Geräts definiert sind. Diese Konfigurationsdatei lässt sich über das Kontextmenü eines UniPi-Channels importieren.

Der Zugriff auf die Daten des UniPi Geräts erfolgt über einen lokalen Modbus TCP Server. Somit werden Variablen analog zum [Modbus Device Plugin](#) definiert, im Abschnitt [Variablen](#) finden Sie die entsprechende Dokumentation.

Fehlerdiagnose

Das UniPi Device Plugin liefert je nach zu untersuchender Schicht verschiedene Statusinformationen. Generell werden die kanalbasierten Diagnoseinformationen durch den Verbindungsstatus des Channels zum UniPi produziert. Die variablenbasierten Diagnoseinformationen werden während des Lese-/Schreibzugriffs auf die verschiedenen Variablen produziert.

Kanal

Um den Status des UniPi-Kanals zu überwachen und zu diagnostizieren, werfen Sie einen Blick auf das folgende Bild:

S103

Could not create connection: No connection could be made because the target machine actively refused it

Name	Display Name	Actual Value	Value Type	Description
Settings	Settings			Defines the different settings
Control	Control			Provides device channel cont
Status	Status			Provides information about t
Variables	Variables			

Das obige Bild zeigt das Bedienfeld des UniPi-Kanals, das alle statusrelevanten Informationen anzeigt. Das Bedienfeld aktualisiert automatisch seine Statusinformation, wenn ein neuer Status verfügbar ist.

Statuskreis

Farbe	Bedeutung
	Der Kanal ist gestoppt. Klicken Sie den ►-Button, um ihn zu starten.
	Der Kanal startet oder stoppt gerade oder wartet auf den Verbindungsaufbau.
	Der Kanal läuft und es wurde erfolgreich eine Verbindung hergestellt. Sie können ihn durch Klick auf den ■-Button stoppen.
	Der Kanal läuft, aber die Verbindung ist momentan fehlerhaft. Bitte überprüfen Sie den Statustext für weitere Informationen.

Variablen

Um den Status der verschiedenen Variablen zu überwachen und zu diagnostizieren, werfen Sie einen Blick auf die in CoDaBix® angezeigte Status-Eigenschaft der Spalte. Benutzen Sie den Button „Read actual Value“, um die Werte aus des UniPi auszulesen und das Ergebnis in den Variablen zu speichern.

Name	Display Name	Actual Value	Value Type	Description	Path	Status
AO1	AO1	0	UInt16		0.2	Good
DO1	DO1	False	Boolean		0.0	Good
DO2	DO2	False	Boolean		0.1	Good
DO3	DO3	False	Boolean		0.2	Good
DO4	DO4	False	Boolean		0.3	Good
DO5	DO5		Boolean		0.12	Bad: Exception of type

Logdatei

Alle kanalbezogenen Statusinformationen werden auch in die kanalspezifische Logdatei im

[LoggingFolder] protokolliert. Jede Logdatei wird nach dem Namensschema UniPi Device.<ChannelName>.log benannt. Der Inhalt einer solchen Logdatei kann wie folgt aussehen:

```
...
2018-04-11 11:32:37.0 +2: [Error] Error (Severity=High): Code=[-1],
Text=[The operation has timed-out.], Details=[]
...
```

Entities

Wie jedes Device Plugin erweitert das UniPi Device Plugin das CoDaBix [Device Modell](#).

Device

Der Device Typ UniPiDevice des Plugins definiert auch den UniPiDeviceChannel und erweitert somit die grundlegenden CodabixDevice und CodabixDeviceChannel Entities. Während das UniPiDevice nur eine Konkretisierung des CodabixDevice darstellt, erweitert der UniPiDeviceChannel den CodabixDeviceChannel mit den UniPi Variable Entities.

Channel

Der Kanal wird von einem Channel Worker behandelt, der eine TCP Socket Verbindung zum lokalen Server herstellt. Zur Fehlerdiagnosezwecken überprüft der Worker automatisch alle 10 Sekunden den Status der TCP Socket Verbindung, um den Channel Statuscode und seine Beschreibung zu aktualisieren und Verbindungsfehler zu erkennen.

Der Worker liest standardmäßig keine Werte. Wenn ein Anwender oder Plugin in CoDaBix® einen synchronen Lesevorgang der Channel Variablen anfordert (z.B. mit der „Read actual value“-Funktion in der CoDaBix Webkonfiguration), liest der Channel Worker diese vom UniPi und schreibt diese in die entsprechenden CoDaBix Nodes.

Ähnlich schreibt der Channel Worker auch die Werte in den UniPi, wenn ein Client oder Plugin Werte in die Channel Variablen schreibt.

Damit eine UniPi Variable regelmäßig gelesen wird, können Sie in der Webkonfiguration bei dem Node „History Options“ auf Yes stellen (was eine interne Subscription erstellt), oder Sie können zum Beispiel einen OPC UA Client verwenden, der mit dem OPC UA Serverplugin verbunden ist und damit eine Subscription für die UniPi Variablenodes erstellen. In diesen Fällen liest der Channel Worker in regelmäßigen Intervallen die Variablen vom UniPi und schreibt den neuen Wert nach einer Wertänderung automatisch in die entsprechende CoDaBix® Node.

Variable

Unter dem Variables-Node können Sie Datenpunktnodes erstellen, die vom UniPi-Gerät gelesen und geschrieben werden.

Die Value Type-Eigenschaft muss dabei auf den entsprechenden Variablentyp festgelegt werden

(Boolean, Byte, Int16, UInt16, Int32, UInt32 und die zugehörigen Array-Typen).

Migration von Version 1.0.0 auf 1.0.1

Ab der Version 1.0.1 gibt es die Möglichkeit andere Datentypen aus den Modbus-Entities zu lesen, als sie über Modbus bereitgestellt werden (Boolean in Coils / UInt16 in Registern). Um diese Funktion abzubilden, wird die Definition des Modbus-Entität einer Variable nicht mehr wie in Version 1.0.0 über den Variablentypen festgelegt, sondern durch den Pfad der Variable definiert. Dies führt zu Inkompatibilität bei Werten, die aus Modbus-Registern gelesen werden.

Die Migration kann durch Import der [Standard Channel-Konfiguration](#) im bestehenden UniPi-Channel durchgeführt werden. Sollte dies nicht möglich sein, da dort Änderungen vorgenommen wurden, kann die Migration auch manuell mit Hilfe der Dokumentation des [Modbus Device Plugin](#) möglich.

Ordner & Dateien

Ordner

Name	Pfad	Zweck / Verwendung
AssemblyFolder	<CodabixInstallDir>/plugins/UniPiDevicePlugin/	Beinhaltet die Plugin Assembly Datei.
ConfigFolder	<CodabixDataDir>/plugins/UniPiDevicePlugin/	Beinhaltet die Plugin Konfigurationsdatei.
LoggingFolder	<CodabixDataDir>/log/	Beinhaltet die Plugin Log Dateien.

Dateien

Typ	Pfad	Zweck / Verwendung
Assembly	[AssemblyFolder]/CoDaBix.UniPiDevicePlugin.dll	Die Plugin Assembly Datei.
Logging	[LoggingFolder]/UniPi Device.<ChannelName>.log	Die Log Datei.

Versionsinformation

Dieses Dokument

Datum	2019-01-14
Version	2.0

Plugin

Name	UniPi Device Plugin
Node	/System/Devices/UniPi Device
Version	1.0.1

Assembly

Name	CoDaBix.UniPiDevicePlugin.dll
Datum	2018-05-28
Version	1.0.1.0

From:

<https://www.codabix.com/> - **CoDaBix®**

Permanent link:

<https://www.codabix.com/de/plugins/device/unipideviceplugin>

Last update: **2021/07/30 13:39**