

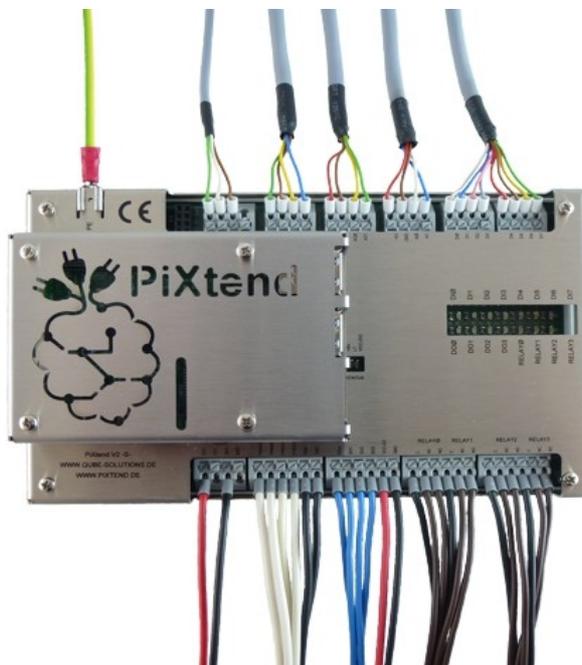


PiXtend V2 Professional

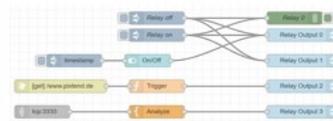
Application-Note: PiXtend V2 mit Node-RED

Application-Note PiXtend V2 mit Node-RED

Einrichtung und Programmierung



PiXtend V2 -S-/-L-
Node-RED Support



APP-PX2-311

Stand 03.12.2019, V1.02

Qube Solutions GmbH
Arbachtalstr. 6, 72800 Eningen, Germany

<http://www.qube-solutions.de/>

<https://www.pixtend.de>



Versionshistorie

Version	Beschreibung	Bearbeiter
1.00	Dokument erstellt	RT
1.01	Hinweise zur Kompatibilität mit PiXtend V2 -L- hinzugefügt Die Beschreibung zeigt weiterhin den Ablauf für das PiXtend V2 -S-, dieser gilt jedoch analog auch für das PiXtend V2 -L-, es müssen zur Programmierung später lediglich andere Knoten verwendet werden.	RT
1.02	Update der URLs auf Node-RED Knoten Version 0.1.2. Die Knoten funktionieren ebenfalls mit Node-RED Version 1.x und später	RT

Inhaltsverzeichnis

1. Einleitung.....	3
1.1 Voraussetzungen.....	5
1.2 Haftungsausschluss.....	5
1.3 Sicherheitshinweise.....	5
2. SD-Karte erstellen.....	6
3. <i>Node-RED</i> starten.....	8
4. Beispiel Flow erstellen.....	10
5. Grafische Oberfläche erstellen.....	13
6. Frequently Asked Questions (FAQ).....	15

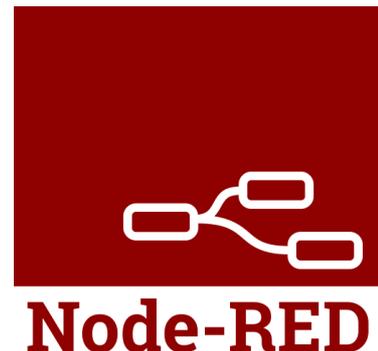


1. Einleitung

Wir freuen uns Ihnen einen *Node-RED* Knoten für das PiXtend V2 -S- vorstellen zu können, damit Sie noch einfacher und schneller in die IoT-Welt (Internet of Things) einsteigen können.

Node-RED ist ein Programmierwerkzeug mit dem es möglich ist verschiedene Geräte, APIs und Onlinedienste auf neue und interessante Art und Weise „virtuell“ zu verbinden.

Es bietet einen Webbrowser basierten Editor an, der es einem einfach macht verschiedene Datenflüsse miteinander zu „verkabeln“, unter Verwendung verschiedenster „Knoten (Nodes)“ die das System bereitstellt.¹



Fluss- bzw. Datenfluss basierte Programmierung wurde in den 1970er Jahren von J. Paul Morrison² erfunden und bietet die Möglichkeit das Verhalten einer Applikation unter Verwendung eines Netzwerks bestehend aus „Black-Boxes“, in *Node-RED* heißen Sie *Knoten (Nodes)*, zu beschreiben.

Jeder Knoten hat dabei eine festgelegte und eindeutige Aufgabe. Werden an einen Knoten Daten übermittelt, so kann der Knoten diese Daten verarbeiten und gibt sie dann an den nächsten Knoten weiter. Es ist Aufgabe des Netzwerks, in dem diese Knoten enthalten sind, für den richtigen Datenfluss zu sorgen.

Node-RED selbst basiert auf einer Runtime die in *Node.js* geschrieben wurde, der Fluss-Editor kann über einen Webbrowser aufgerufen und bedient werden. Eine Anwendung kann man erstellen, indem man aus „*Palette*“ den bzw. die gewünschten *Knoten* auf die *Arbeitsfläche* zieht und beginnt diese miteinander per „*Kabel*“ zu verbinden. Mit einem Klick auf die Schaltfläche „*Deploy*“ wird die Applikation an den *Node-RED* Server übertragen und aktiv geschaltet.

Da *Node-RED* für das IoT-Zeitalter gemacht ist, darf auch eine grafische Bedienoberfläche für den PC bzw. das Smartphone und Tablett nicht fehlen. Es gibt mittlerweile viele verschiedene *Node-RED* Module, die es dem Anwender sehr leicht machen, durch *Node-RED* eine web-basierte grafische Bedienoberfläche zu erstellen.

¹ Definition laut Projektbeschreibung auf <https://nodered.org/>, siehe auch <https://nodered.org/about/>

² Siehe https://en.wikipedia.org/wiki/Flow-based_programming



Node-RED wurde Anfang 2013 als Nebenprojekt von *Nick O'Leary* und *Dave Conway-Jones* bei der *IBM Emerging Technology Services Gruppe* ins Leben gerufen.

Was als *Proof-of-Concept* für Visualisierung und Manipulation von MQTT topics begann, entwickelte sich schnell zu einem allgemeinen Tool, das sich leicht erweitern lässt. Im September 2013 wurde *Node-RED* Open-Source und im Oktober 2016 eines der Mitgründungsprojekte der *JS Foundation*, wo es seitdem für alle frei zugänglich ist und weiterentwickelt wird.³

Node-RED wird mit dem Raspbian Betriebssystem für den Raspberry Pi mit ausgeliefert und kann sofort verwendet werden. Es gehört neben Python und vielen anderen Komponenten zur Grundausstattung des Raspberry Pi, was den Einstieg in die flussbasierte Programmierung sehr einfach macht.

Wir wünschen viel Spaß beim Testen, „Verdrahten“ und Experimentieren!

Viele weitere Informationen, Tipps und Tricks finden Sie auch in unserem Support-Forum unter: <https://www.pixtend.de/forum/>

Die jeweils neusten Versionen aller Dokumente und Software-Komponenten finden Sie im Download-Bereich unserer Homepage: <https://www.pixtend.de/downloads/>

³ Siehe <https://nodered.org/about/>



1.1 Voraussetzungen

Diese App-Note wurde für *Node-RED* v0.17.5 (und später) und das *PiXtend V2 -S-* erstellt. Zur Nutzung des *PiXtend V2 -S-* in *Node-RED* verwenden wir das *PiXtend Basis Image* SD-Karten Image. Kenntnisse über die Funktion von *Node-RED*, die Verwendung der Palette und wie die Einstellungsdatei zu bearbeiten ist, werden vorausgesetzt. Diese App-Note richtet sich an Experten. Die Erstellung der SD-Karte und die Benutzung der *PiXtend V2 -S- Knoten* in *Node-RED* werden im Folgenden exemplarisch für einen Anwendungsfall beschrieben, abweichende Anforderungen müssen vom Anwender selbst erarbeitet oder bei Qube Solutions GmbH beauftragt werden.

Es gibt keine spezielle Festlegung auf ein Raspberry Pi Modell.

Wir empfehlen jedoch eines der folgenden Modelle: **B+**, **2 B**, **3 B**, **3 B+**, **4 B**.

Laden Sie das **PiXtend Basis Image** SD-Karten Image aus unserem [Download-Bereich](#) herunter, hier sind *Node-RED* und die *PiXtend V2 -S- Knoten* bereits installiert. Verwenden Sie dieses Image als Ausgangspunkt für Ihre *Node-RED* Entwicklung.

Auf den Raspberry Pi können Sie entweder mit direkt angeschlossener Tastatur und Monitor oder per SSH (TeraTerm / Putty) von einem PC aus zugreifen.

Zur Installation weiterer *Node-RED* Knoten benötigt der Raspberry Pi eine aktive Internetverbindung.

Hinweis:

Ab **Basis Image 2.1.4.0** wird ebenfalls das **PiXtend V2 -L-** unterstützt. Einfach statt der *PiXtendV2S* die *PiXtendV2L* Knoten verwenden.

1.2 Haftungsausschluss

Qube Solutions GmbH kann nicht für etwaige Schäden verantwortlich gemacht werden die unter Umständen durch die Verwendung der zur Verfügung gestellten Software, Hardware, Treiber oder der hier beschriebenen Schritte oder Software von Drittherstellern entstehen können.

1.3 Sicherheitshinweise



PiXtend darf nicht in sicherheitskritischen Systemen eingesetzt werden.

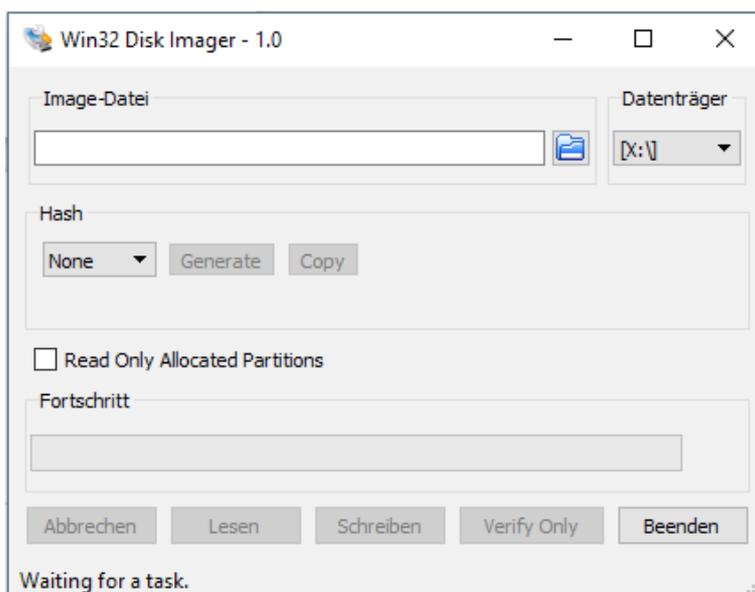
Prüfen Sie vor der Verwendung die Eignung von Raspberry Pi und PiXtend für Ihre Anwendung.



2. SD-Karte erstellen

Um eines unserer Images auf eine leere SD-Karte zu übertragen, wird ein PC mit SD-Karten-Lesegerät benötigt. Als Software zum Schreiben von SD-Karten empfehlen wir das Open Resource Programm „Win32DiskImager“. Das Programm läuft unter *Microsoft Windows* und kann unter folgender Adresse kostenlos heruntergeladen werden:

<http://sourceforge.net/projects/win32diskimager/>



HINWEIS:

Zur Ausführung bzw. für den Laufwerkszugriff muss das Programm *Win32DiskImager* mit Administrator-Rechten bzw. unter einem Administratorkonto gestartet werden, sonst besteht die Möglichkeit, dass die SD-Karte gar nicht oder fehlerhaft beschrieben wird!



Zur Erstellung einer SD-Karte bzw. zum Übertragen eines unserer Images auf Ihre SD-Karte führen Sie folgende Schritte aus:

- SD-Karte in das Kartenlesegerät einlegen bzw. einstecken
- Warten bis Windows das Medium erkannt hat
- Den Arbeitsplatz oder Windows Explorer öffnen und den Laufwerksbuchstaben notieren, dieser muss später im *Win32DiskImager* angegeben werden
- Die Zip-Datei in der sich das SD-Karten Image von unserem Downloadbereich befindet öffnen und entpacken.
- Das Programm *Win32DiskImager* mit Administrator-Rechten starten
- Die entpackte Image-Datei im *Win32DiskImager* unter „Image-Datei“ auswählen
- Unter *Datenträger* den Laufwerksbuchstaben der SD-Karte auswählen

ACHTUNG:



Wählen Sie den falschen Datenträger aus, können alle Daten darauf verloren gehen, sobald Sie den Schreibvorgang mit *Win32DiskImager* gestartet haben.

Entfernen Sie am besten alle Wechseldatenträger (z.B. USB-Sticks oder USB-Festplatten) bevor Sie den Schreibvorgang starten ggf. sogar bevor Sie das *Win32DiskImager* Programm starten.

- Haben Sie alle Einstellungen vorgenommen, starten Sie den Schreibvorgang mit einem Klick auf „Schreiben“
- Warten Sie bis der Schreibvorgang abgeschlossen ist

Nach erfolgreicher Übertragung des Images auf die *SD-Karte*, kann diese am *PC* ausgeworfen und in den *SD-Karten Slot* des *Raspberry Pi* gesteckt werden.

Sollten Sie während oder nach der Erstellung der *SD-Karte* Fenster sehen, mit oder ohne Inhalt, oder eine Fehlermeldung erhalten, dass auf ein Laufwerk nicht zugegriffen werden kann bzw. Sie werden aufgefordert das Medium zu formatieren, klicken Sie in solchen Fällen bei diesen Dialogen immer auf die „Abbrechen“ Schaltfläche und schließen Sie eventuell aufgegangenen Fenster.

In einer solchen Situation liegt kein Fehler vor. Sie erhalten diese Meldungen, da *Microsoft Windows* nicht in der Lage ist die Daten der *SD-Karte* zu lesen, da diese für das *Raspbian* Betriebssystem bestimmt sind. Werfen Sie, wie erwähnt, die *SD-Karte* aus und stecken Sie diese in den *SD-Karten Slot* des *Raspberry Pi*. Schalten Sie die Versorgung ein und Ihr *Raspberry Pi* startet das *Raspbian* Betriebssystem.



3. Node-RED starten

Nach dem Start vom *Raspbian* Betriebssystem kann *Node-RED* sofort gestartet werden und alle *PiXtend V2 -S- Knoten* sind in der Palette in der Kategorie „*PiXtend V2S*“ zu finden. Jeder Knoten verfügt über eine Beschreibung die weitere Hilfe und Informationen bietet und was mit dem Knoten gemacht werden kann.

WARNUNG:



Betreiben Sie *Node-RED* nicht parallel zu anderen *PiXtend Softwarekomponenten* wie den *PiXtend Linux Tools (pxdev)* oder Python Programmen welche die *PiXtend Python Library V2 (PPLV2)* verwenden.

Bauteile auf dem *PiXtend V2 -S-* könnten Schaden erleiden, sollten mehrere unterschiedliche Programme gleichzeitig mit dem Mikrocontroller des *PiXtend V2 -S-* kommunizieren und unterschiedliche Befehle senden.

Node-RED können Sie mit folgendem Befehl starten:

node-red-start

Der Start der *PiXtend V2 -S- Node-RED* Knoten ist beim Start von *Node-RED* gut zu sehen, der Start bzw. die Einbindung der Knoten wird im Log in der Konsole ausgegeben.

Zusätzlich ist zu sehen dass *Node-RED* ein weiteres Programm startet, den *PiXtend Server V2 -S-*, dieser kümmert sich um die Kommunikation mit dem Mikrocontroller auf dem Board.

```
pi@raspberrypi:~/noderedinst $ node-red-start
Start Node-RED
Once Node-RED has started, point a browser at http://xxx.xxx.xxx.xxx:1880
On Pi Node-RED works better with the Firefox or Chrome browser
Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot
To find more nodes and example flows - go to http://flows.nodered.org
Starting as a systemd service.
Started Node-RED graphical event wiring tool..
9 Dec 10:11:32 - [info] Welcome to Node-RED
=====
9 Dec 10:11:32 - [info] Node-RED version: v0.17.5
9 Dec 10:11:32 - [info] Node.js version: v6.12.1
9 Dec 10:11:32 - [info] Linux 4.9.41-v7+ arm LE
9 Dec 10:11:33 - [info] Loading palette nodes
9 Dec 10:11:37 - [info] Starting PiXtend V2 -S- Node-RED node!
pigtendsrv2s - Starting PiXtend Server V2 -S-
9 Dec 10:11:38 - [info] Settings file : /home/pi/.node-red/settings.js
9 Dec 10:11:38 - [info] User directory : /home/pi/.node-red
9 Dec 10:11:38 - [info] Flows file : /home/pi/.node-red/flows_raspberrypi.json
9 Dec 10:11:38 - [info] Creating new flow file
9 Dec 10:11:38 - [info] Starting flows
9 Dec 10:11:38 - [info] Started flows
9 Dec 10:11:38 - [info] Server now running at http://127.0.0.1:1880/
```



Haben Sie *Node-RED* über die Linux Konsole manuell gestartet und möchten es jetzt wieder beenden, dann führen Sie die zwei folgenden Schritte durch:

1. Drücken Sie die Tastenkombination Strg+C (Ctrl+C), dadurch verlassen Sie die Log-Ausgabe des Programms, *Node-RED* wurde aber noch nicht beendet.
2. Geben Sie nun folgenden Befehl ein um Node-RED zu beenden (stoppen):

```
node-red-stop
```

Möchten Sie Node-RED automatisch starten lassen oder Sie suchen mehr Informationen zu diesem Thema, dann können Sie folgende Internet-Adresse in Ihrem Browser aufrufen:

<https://nodered.org/docs/hardware/raspberrypi>

Hinweis:

Ab **Basis Image 2.1.4.0** wird ebenfalls das **PiXtend V2 -L-** unterstützt. Einfach statt der PiXtendV2S die PiXtendV2L Knoten verwenden.



4. Beispiel Flow erstellen

Nach dem Start von *Node-RED* können wir den ersten Flow, also das erste Programm, erstellen.

Für einen schnellen Einstieg bietet es sich an, eines der 4 Relais anzusteuern und dieses ein- und auszuschalten.

1. Einen *Node-RED* kompatiblen Browser öffnen, z.B. Firefox oder Chrome.

2. Den *Node-RED* Editor im Browser aufrufen:

<http://ip-adresse-rpi:1880>

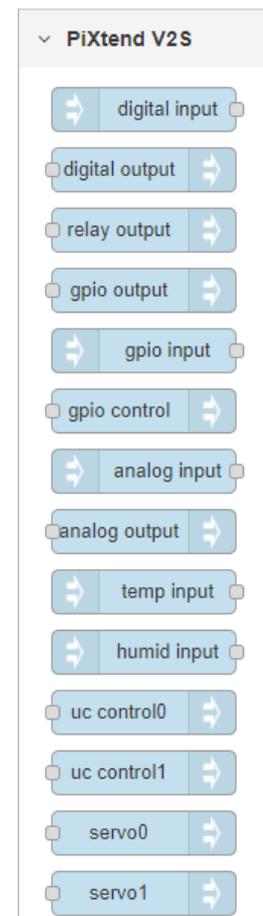
Für „*ip-adresse-rpi*“ die Netzwerkadresse des Raspberry Pi eingeben, üblicherweise wird diese von *Node-RED* beim Start auf der Konsole ausgegeben.

Der *Node-RED* Web-Server selbst ist über den Port 1880 zu erreichen.

3. Hat der Web-Browser den *Node-RED* Editor geladen, kann es mit dem erst Flow los gehen. An dieser Stelle bietet es sich an in der Palette von *Node-RED* nach unten zu scrollen und zuschauen welche Knoten in der Kategorie „*PiXtend V2S*“ vorhanden sind. Alle aktuell verfügbaren Knoten sind hier zu sehen.

Hinweis:

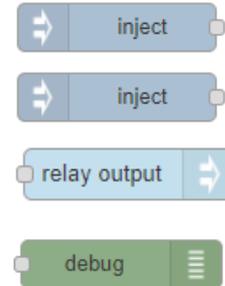
Ab **Basis Image 2.1.4.0** wird ebenfalls das **PiXtend V2 -L-** unterstützt. Einfach statt der **PiXtendV2S** die **PiXtendV2L** Knoten verwenden.





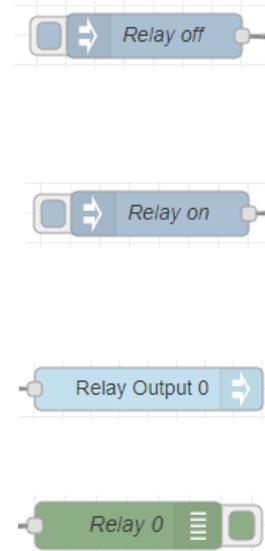
4. Im Folgenden ziehen wir 4 Knoten aus der Palette auf die Arbeitsfläche:

1. *inject* Knoten zum Ausschalten von Relais 0
2. *inject* Knoten zum Einschalten von Relais 0
3. *relay output* Knoten zur Ansteuerung von Relais 0
4. Debug Knoten zur Anzeige der Nachrichten (msg) der *inject* Knoten



5. Jetzt müssen bei jedem Knoten nur noch die *Eigenschaften* angepasst werden, dies geht einfach über einen *Doppelklick* auf den entsprechenden Knoten in der Arbeitsfläche:

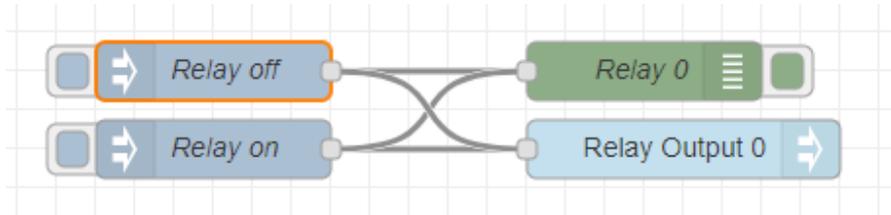
1. *inject* Knoten zum Ausschalten
 - Payload: Typ: *boolean*
 - Payload: Wert: *false*
 - Name: *Relay off*
2. *inject* Knoten zum Einschalten
 - Payload: Typ: *boolean*
 - Payload: Wert: *true*
 - Name: *Relay on*
3. *relay output* Knoten
 - Name: *Relay Output 0*
 - Relay: *Relay 0*
4. Debug Knoten
 - Name: *Relay 0*



6. Im nächsten Schritt müssen die Knoten miteinander „verkabelt“ werden, damit das *Relais 0* auf dem *PiXtend V2 -S-* auch angesteuert werden kann.

Die Knoten wie folgt verbinden:

1. *inject* Knoten zum Ausschalten --> *Relay Output 0* Knoten
2. *inject* Knoten zum Einschalten --> *Relay Output 0* Knoten
3. *inject* Knoten zum Ausschalten --> Debug Knoten „*Relay 0*“
4. *inject* Knoten zum Einschalten --> Debug Knoten „*Relay 0*“



Fertiger Flow zum Steuern von Relais 0

7. Sind alle Knoten verbunden, so ist der erste Flow fertig und mit einem Klick auf „**Deploy**“ wird alles an den *Node-RED* Server übertragen.



8. Ab jetzt kann man das *Relais 0* mittels der *Buttons* vorne an den „*inject*“ Knoten ein- und wieder ausschalten. Jeder Klick erzeugt eine *Nachricht (msg)* die als *Daten (payload)* entweder den Wert „*true*“ oder „*false*“ enthält. Der Wert *true* schaltet das Relais 0 ein und der Wert *false* schaltet es wieder aus.

Geschafft! Glückwunsch, Sie haben Ihren ersten Flow in *Node-RED* erstellt und das Relais 0 angesteuert.

Beispiel als Text im JSON-Format zum Copy & Pasten:

```
[{"id":"5ab39907.55a8f8","type":"inject","z":"cccbeeb2.83ca2","name":"Relay off","topic":"","payload":"false","payloadType":"bool","repeat":"","crontab":"","once":false,"x":400,"y":180,"wires":[[{"id":"6b462b79.d11be4","z":"cccbeeb2.83ca2","name":"Relay 0","relnum":"0","relnitval":"0","relset":"","x":640,"y":220,"wires":[]}, {"id":"6b462b79.d11be4","type":"debug","z":"cccbeeb2.83ca2","name":"Relay 0","active":true,"console":"false","complete":"payload","x":620,"y":180,"wires":[]}, {"id":"de5f8a56.6b4b88","type":"inject","z":"cccbeeb2.83ca2","name":"Relay on","topic":"","payload":"true","payloadType":"bool","repeat":"","crontab":"","once":false,"x":400,"y":220,"wires":[[{"id":"831d4179.f161","z":"cccbeeb2.83ca2","name":"Relay Output 0","relnum":"0","relnitval":"0","relset":"","x":640,"y":220,"wires":[]}]]}]
```



5. Grafische Oberfläche erstellen

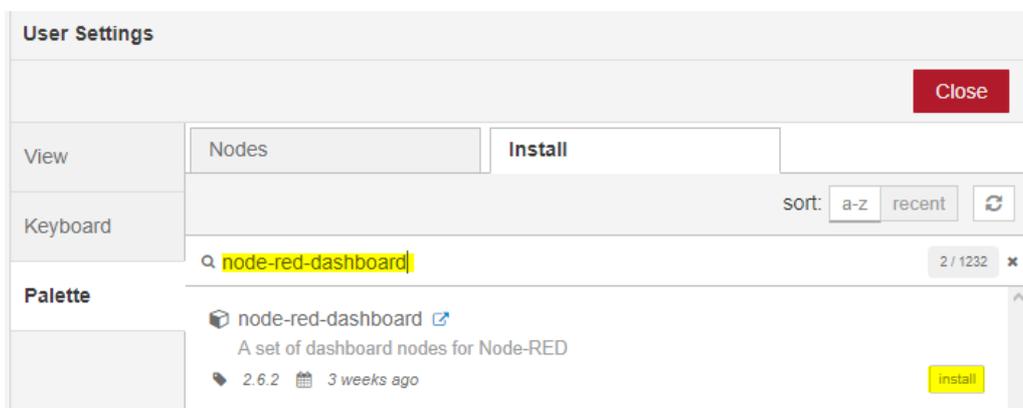
Da *Node-RED* für das IoT-Zeitalter gemacht ist, darf auch eine grafische Bedienoberfläche für den PC bzw. das Smartphone und Tablet nicht fehlen.

Passenderweise gibt es genau für diesen Zweck eine Erweiterung bzw. ein Modul das entsprechende Knoten für *Node-RED* bereitstellt und es ermöglicht auf der Basis von fest vorgegebenen grafischen Elementen eine web-basierte Anzeige zu erstellen.

***node-red-dashboard* installieren:**

Im *Node-RED* Hauptmenü den Eintrag „*Manage Palette*“ wählen und den Reiter „*Palette*“ aktivieren. Auf der rechten Seite erscheinen 2 neue Reiter. Der erste (*Nodes*) zeigt alle aktuell installierten Knoten an und mittels des zweiten (*Install*) können neue Knoten direkt vom Internet aus installiert werden.

Den Reiter *Install* anklicken und nach dem Paket *node-red-dashboard* suchen und dieses installieren.



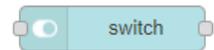
Nach einer kurzen Wartezeit ist das Paket heruntergeladen und installiert worden. Im *Node-RED* Editor taucht eine Meldung mit grünem Rand auf und zeigt an welche Knoten alle installiert wurden.

Das Menü können wir über den *Close* Button verlassen und in der *Palette* nachschauen welche neuen Knoten und damit verbunden, welche grafischen Elemente uns jetzt zur Verfügung stehen.



Schalter-Knoten verwenden:

Da wir in unserem Beispiel das Relais 0 ein- und wieder ausschalten wollen, können wir für diese Aufgabe den *Schalter-Knoten (switch)* verwenden.



Einfach den entsprechenden Knoten auf die Arbeitsfläche ziehen und einen *Doppelklick* auf den Knoten ausführen. Im erscheinenden Einstellungsdialog müssen wir einige Anpassungen vornehmen, bevor wir den Schalter verwenden können und ein *Deploy* möglich ist.

Den Schalter wie folgt parametrieren:

- *Group* --> Stift klicken --> *Name* --> Relay 0
- *Tab* --> Stift klicken --> *Name* --> Test
- *Add* klicken
- *Add* klicken
- *Label* --> On/Off
- *Done* klicken

node properties

Group: Relay 0 [Test]

Size: auto

Label: On/Off

Icon: Default

If msg arrives on input, pass through to output:

When clicked, send:

On Payload: true

Off Payload: false

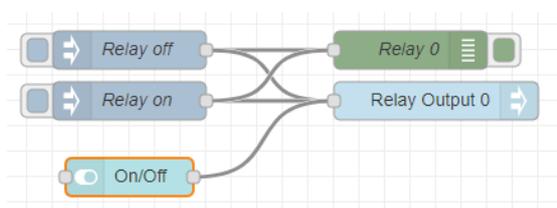
Topic:

Name:

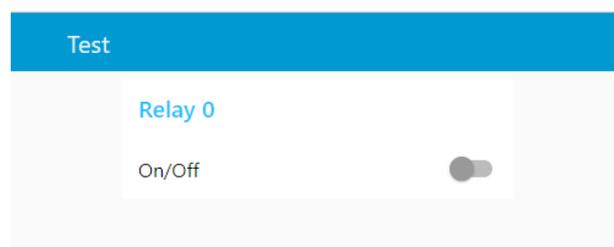
Jetzt muss nur noch der Ausgang des Schalter-Knotens mit dem Eingang von *Relay Output 0* verbunden werden und wir können auf *Deploy* klicken. Als nächstes können wir uns den Schalter auf der grafischen Oberfläche anschauen.

Die grafische Oberfläche lässt über die *Node-RED* Adresse mit dem Zusatz „ui“ erreichen:

<http://ip-adresse-rpi:1880/ui>



Ansicht Node-RED Arbeitsbereich



Ansicht der grafischen Oberfläche im Browser

Glückwunsch! Sie haben *Node-RED* mit den *PiXtend V2 -S-* Knoten erfolgreich in Betrieb genommen. Wir wünschen weiterhin viel Spaß beim *verdrahten*, mit *Node-RED* und *PiXtend*.



6. Frequently Asked Questions (FAQ)

Gibt es Einschränkungen bei der Verwendung von *Node-RED* auf dem Raspberry Pi die ich beachten muss?

Uns sind derzeit keine Einschränkungen bekannt, einzig die Rechenleistung der Quad-Core CPU und die Menge an Arbeitsspeicher des Raspberry Pi können Ihre Kreativität begrenzen.

Die verfügbaren Knoten und Module in *Node-RED* scheinen begrenzt zu sein, gibt es da nicht mehr?

Da *Node-RED* auf *Node.js* aufbaut gibt es eine ganze Fülle an weiteren Knoten und Funktionen die man nachinstallieren kann, hierfür wird der *npm* Paket Manager verwendet. Im *Node-RED* Menü einfach den Eintrag „*Manage Palette*“ anklicken und im Reiter „*Install*“ die Suchfunktion nutzen um neue Knoten zu finden und zu installieren.

Um sich erst Mal einen Überblick zu verschaffen empfiehlt es sich ggf. zu ersten die Webseite von *npm* zu besuchen und hier nach *Node-RED* Packages zu suchen, um zu sehen was es alles gibt.

Weitere Informationen gibt's z.B. hier:

- <https://nodered.org/docs/getting-started/adding-nodes>
- <https://github.com/node-red/node-red-nodes>
- <https://www.npmjs.com/>

Kann ich die PiXtend V2 -S- Knoten ohne Einschränkung und in beliebigen Kombinationen verwenden?

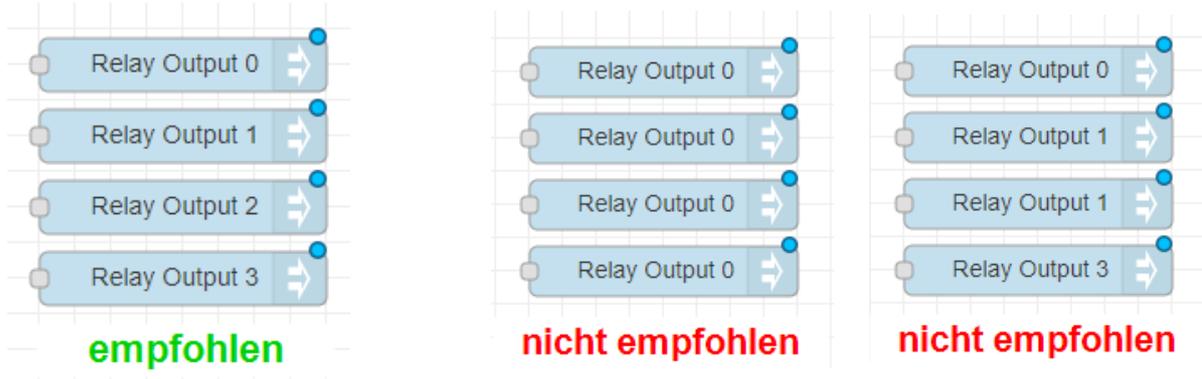
Wir raten von der mehrfach Verwendung der *PiXtend V2 -S-* Knoten ab, da jeder Knoten einem Kommunikationsprozess entspricht und z.B. bei Ansteuerung der *Relais* oder *digital Ausgänge* möglicherweise Probleme entstehen die zu *Schäden* am *PiXtend V2 -S-* oder an der Peripherie führen können.

Jeder Knoten sollte nur in der Anzahl der auf dem *PiXtend V2 -S-* tatsächlich vorhandenen Hardware genutzt werden.

Beispiel: Sind 4 Relais vorhanden, so kann der *relay output* Knoten 4 Mal in einem einzigen *Flow*, bezogen auf alle *Flows* in *Node-RED*, verwendet werden, allerdings wieder nur, wenn jeder Knoten genau einem Relais zugeordnet wird, so dass es keine



Überschneidungen gibt. Jeder Knoten muss so konfiguriert werden, dass er genau einer Hardwarekomponente zugeordnet ist.



Ferner sollten alle Konfigurationsknoten generell nur ein Mal verwendet werden, um die verschiedenen Register des *PiXtend V2 -S-* Mikrocontrollers zu konfigurieren. Hierbei handelt es sich um die Knoten:

- *gpio-control*
- *uc control0*
- *uc control 1*



Gerne möchten wir Sie für den Informationsaustausch in die Foren von Qube Solutions einladen:

<https://www.pixtend.de/forum/>