

► **Statische Inhalte:** Statische Inhalte werden mit der Programmlogik direkt in der App implementiert. Dazu gehört beispielsweise die Anzeige der Zeit mit einer besonderen visuellen Darstellung.

Diese Aufzählung zeigt beispielhaft Implementierungen von Inhalten für die App HomeInfoSystem. Für Anwendungen in Unternehmen (EnterpriseInfoSystem) muss man für bestimmte spezifische Services ebensolche Anbindungen schaffen.

Erweiterungen

Die zuvor beschriebenen Services sind lediglich Beispiele und sollen Sie zum Start eines eigenen HIS-Projekts animieren. Sie werden sicher schnell eigene Services finden, die Sie implementieren möchten. Aus meiner Sicht kann man das System sinnvoll erweitern durch:

- Das HomeInfoSystem könnte auch die Werte von verschiedenen Sensoren anzeigen. Die Werte der Sensoren werden an ein IoT-Backend übertragen und von der App in bestimmten Intervallen abgegriffen. Elektroniker können umfassende Hardwareerweiterungen bauen und diese in das System flexibel integrieren.
- **Inhalte mit Zeitbezug einblenden:** Einige Inhalte können mit einem Zeitbezug versehen werden. So ist es sinnvoll, Wochentags zwischen 7 Uhr und 8 Uhr die aktuellen Informationen zu lokalen Nahverkehrsverbindungen einzublenden. Am Wochenende dagegen ist man vielleicht eher an Informationen zu aktuellen Filmen im örtlichen Kino interessiert, die man über eine Kinoprogramm-API ermittelt.
- **Mobile Services:** Man könnte die Systemumgebung um eine mobile App erweitern. Nutzer können Daten an einen Server (Backend) senden und die App ruft diese Daten ab. Dabei sind vielfältige Anwendungen denkbar. Eine weitere spannende Anwendung ist die Nutzung von Geoservices. Mobile Anwender können zum Beispiel (nach Zustimmung und Freigabe) ihre aktuelle Geoposition an ein Backend übermitteln. Die App ruft diese Position regelmäßig ab und zeigt diese auf einer Karte an. Auf diese Weise weiß man, wo sich alle Mitglieder einer Familie gerade befinden.
- **Online Services:** Weitere interessante Services, die man über eine öffentliche API einbinden kann, sind: Kalenderintegration verschiedener Dienste wie Microsoft- oder Google-Kalender, Fahrplan der Deutschen Bahn oder des Verkehrsverbundes (nach Angabe des Ortes für die Abfrage der Verbindungsdaten).
- **Alternative Interface-Struktur:** Statt eines manuellen oder zeitgesteuerten Wechsels zwischen Inhalten könnte man auch eine Gesamtansicht erstellen. Das dürfte insbesondere für größere Bildschirme interessant sein. Beispielsweise könnte man auf der einen Seite das aktuelle Wetter anzeigen und auf der anderen Seite im Wechsel die Nachrichten und aktuelle Fotos.

Sie sehen, es gibt genügend Ideen, um das HomeInfoSystem zu erweitern. Ein besonderer Reiz liegt darin, externe Sensoren (IoT) anzubinden und so zum Beispiel kurz einmal einen Blick auf das Ferienhaus (Kamera) zu werfen. Im unternehmerischen Kontext dürfte es ebenso genügend Anwendungsmöglichkeiten geben.

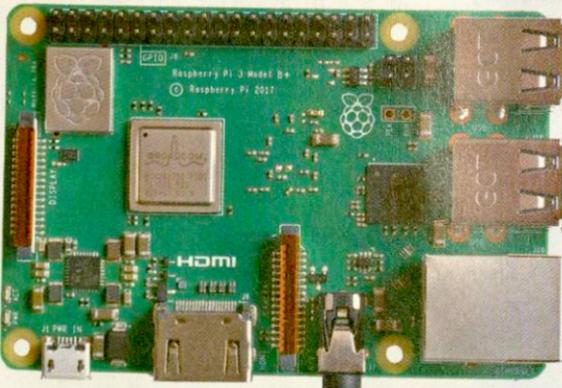
Fazit und Ausblick

Ein solches Infosystem im Heimbereich wie auch in einem Unternehmen ist eine faszinierende Sache. Zu Hause kann es aktuelle Informationen liefern (Wetter, News, Termine) und im Unternehmen wichtige aktuelle Daten präsentieren (Stückzahlen, Termin Teammeeting). Das Interessante ist jedoch, dass man es mit überschaubarem Aufwand an die eigenen Bedürfnisse anpassen kann. Nach der Auswahl der Hardwarekonfiguration, insbesondere des Bildschirms, steht mit der hier beschriebenen Software bereits ein Grundstock an Funktionen zur Verfügung, welche man dann flexibel erweitern und anpassen kann. Der Quellcode ist Open Source. Sie können den Code [1] in Visual Studio 2019 öffnen und nach Ihren eigenen Bedürfnissen weiterentwickeln und modifizieren. ◀

191247-01

 **IM ELEKTOR-STORE**

► **Raspberry Pi 3 B+**
www.elektor.de/raspberry-pi-3-model-b-plus



► **Elektor Raspberry Pi Elektronik-Kit**
www.elektor.de/elektor-raspberry-pi-elektronik-kit

Einstieg in Node-RED

Ein visuelles blockbasiertes Open-Source-Programmierwerkzeug



Node-RED

Von **Dogan Ibrahim**

Node-RED ist in letzter Zeit sehr populär geworden, da es die Entwicklung von IoT-Projekten erheblich vereinfacht. In diesem Artikel sehen wir, was es mit Node-RED auf sich hat und am Beispiel eines sehr einfachen Projekts wie der Steuerung einer LED, wie es eingesetzt werden kann.

In den letzten Jahren haben die Entwicklung und der Einsatz von blockbasierten visuellen Programmierwerkzeugen einen starken Aufschwung erfahren. Die Idee dahinter ist, dass dem Programmierer ein Satz von visuellen Blöcken zur Verfügung gestellt wird, die er lediglich auf logische Weise miteinander verbinden muss, um das gewünschte Anwendungsprogramm zu erstellen. Node-RED ist ein solches visuelles blockbasiertes Programmierwerkzeug, das kostenlos und open-source ist, aber dennoch die Ausführung sehr komplexer Aufgaben erlaubt, einschließlich Web-Zugang, Twitter, E-Mail, HTTP-Abfragen, Bluetooth, MQTT sowie Steuerung der GPIO-Ports gängiger Computer wie Raspberry Pi, Arduino und ESP32. Das Schöne an Node-RED ist, dass der Programmierer nicht lernen muss, komplexe Programme zu schreiben. Zum Beispiel kann er ein Programm zum Verschicken einer E-Mail erstellen, indem er ein paar Knoten zusammenfügt und nur ein paar Zeilen einfachsten Code schreibt.

Node-RED, die Inbetriebnahme

Node-RED ist bereits im Raspbian-Betriebssystem des Raspberry Pi enthalten und kann entweder vom Desktop (**Bild 1**) oder durch

Eingabe des Befehls

```
pi@raspberrypi:~ $ node-red-start
```

von der Befehlszeile (**Bild 2**) aus gestartet werden.

Node-RED wird als Service gestartet. Eine kurze Liste gültiger Befehle zum Starten und Stoppen dieses Dienstes wird nach dem Start auf dem Bildschirm angezeigt, wie es unten in Bild 2 schon dargestellt ist. Die lokale IP-Adresse 192.168.1.202 ist die des Raspberry Pi des Autors. Sie können die IP-Adresse Ihres Raspberry Pi leicht finden, indem Sie den Befehl `ifconfig` am Prompt eingeben und im Ergebnis auf die Zeile schauen, die mit `wlan0` beginnt.

Wenn Sie „Ihre“ IP-Adresse wissen, können Sie sie im Browser Ihres PCs eingeben, um das grafische Node-RED-Benutzerinterface auf dem Monitor zu starten:

```
https://<Ihre IP-Adresse>:1880
```

Beim Autor beispielsweise wäre die Eingabe im Browser:

```
https://192.168.1.202:1880
```

Bild 3 zeigt den dreiteiligen Startbildschirm von Node-RED. Auf der linken Seite befindet sich eine Liste der Knoten, die bei der Installation des Programms standardmäßig verfügbar sind, sich aber bei Bedarf mit zusätzlichen Nodes aus verschiedenen Quellen im Internet erweitern lässt. Der mittlere Teil ist der Arbeitsbereich, in dem die erforderlichen Knoten per Drag & Drop gezogen und zu einem Ablaufprogramm verbunden werden. Auf der rechten Seite befindet sich der Informations- und Debug-Bereich, der bei der Programmentwicklung sehr nützlich werden kann.

Schauen wir uns den Umgang mit Node-RED anhand eines sehr einfachen Beispiels an, einer LED, die über einen 220-Ω-Strombegrenzungswiderstand an GPIO 2 des Raspberry Pi angeschlossen ist (**Bild 4**). Die LED soll von einem Node-RED-Flussprogramm gesteuert werden. Dieses lächerlich einfache Flussprogramm (**Bild 5**) wird so aufgebaut:

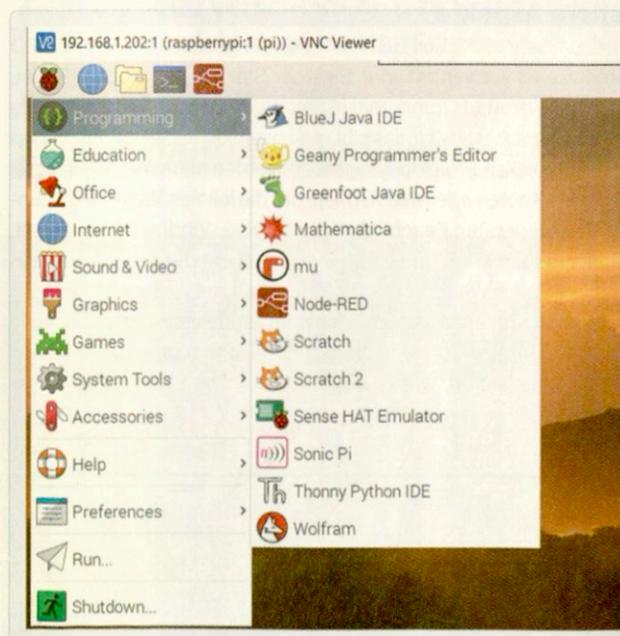


Bild 1. Es beginnt ganz einfach: Node-RED auf dem Raspberry Pi.

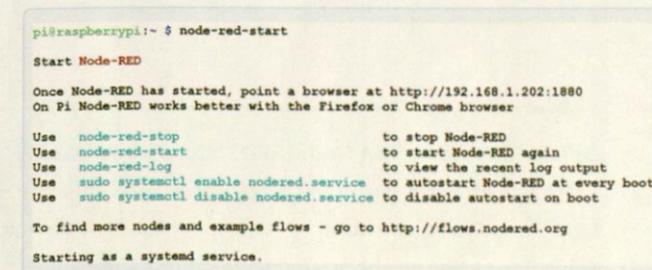


Bild 2. Starten von Node-RED von der Befehlszeile aus.

WEBLINKS

- [1] Projektsoftware: www.elektormagazine.de/191247-01
- [2] VisualStudio: <https://visualstudio.microsoft.com/downloads/>
- [3] OpenWeather: <https://openweathermap.org/>
- [4] Windows 10 IoT-Core: <https://docs.microsoft.com/en-us/windows/iot-core/downloads>
- [5] News-API: <https://newsapi.org/>

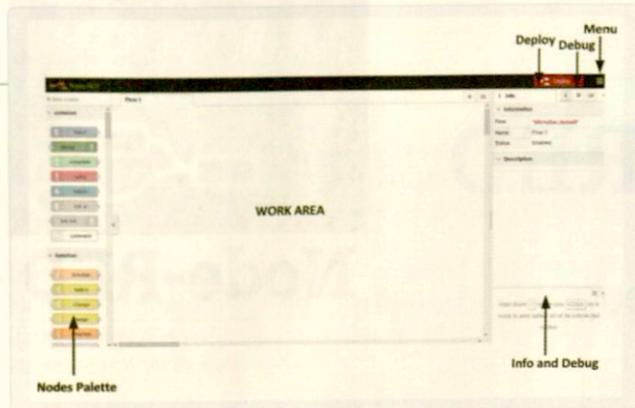


Bild 3. Startmenü von Node-RED.

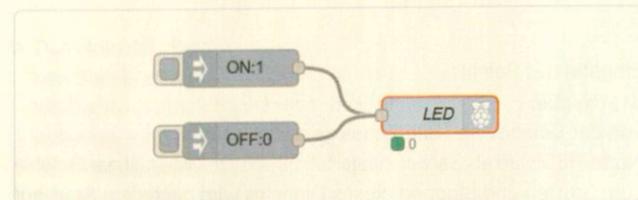


Bild 5. Das Flussdiagramm des Experiments.

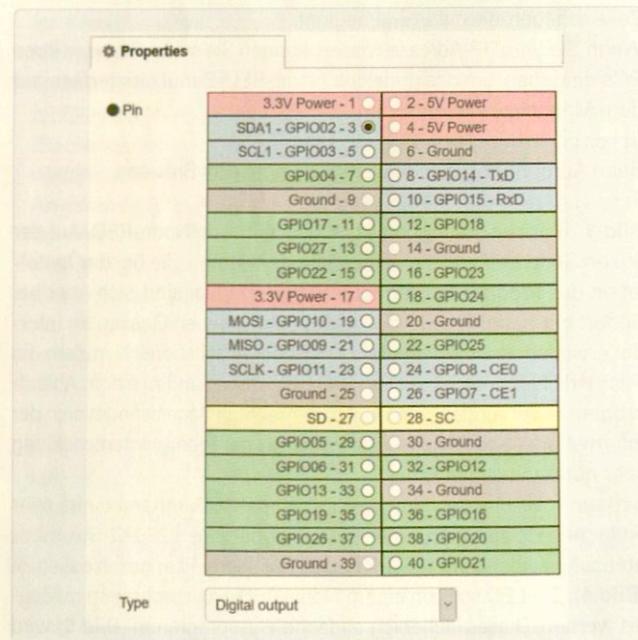


Bild 6. Konfigurieren des Knotens rpi gpio out.

- > Der *Inject*-Knoten wird angeklickt, in den Arbeitsbereich gezogen und abgelegt. Mit Injektionsknoten werden Nachrichten (wie Strings oder Zahlen) in ein Flussprogramm eingebracht. In diesem Beispiel werden wir 1 oder 0 „injizieren“, um die LED ein- und auszuschalten.
- > Doppelklicken Sie auf den Injektionsknoten, um ihn zu konfigurieren. Dabei wird die *Payload* auf *number* gesetzt und eine „1“ eingegeben. Durch Klicken auf das quadratische Feld am linken Rand dieses Knotens wird „1“ ausgegeben, um die LED einzuschalten.
- > Stellen Sie *Topic* des Injektionsmodus auf ON, so dass Sie die Funktion dieses Knotens sehen können. Klicken Sie dann auf *Done*, um das Konfigurationsfenster zu schließen.
- > Klicken, ziehen und legen Sie einen weiteren *Inject*-Knoten in den Arbeitsbereich, setzen Sie *Payload* auf „0“ und das *Topic* auf OFF. Wenn Sie auf diesen Knoten klicken, wird eine „0“ ausgegeben, um die LED auszuschalten.
- > Legen Sie nun einen Knoten *rpi gpio out* in den Arbeitsbereich. Dieser Knoten sendet digitale oder PWM-Daten an einen GPIO-Port des Raspberry Pi. Schließen Sie den Knoten wie in Bild 5 an.
- > Durch einen Doppelklick auf diesen Knoten erscheint ein Fenster (Bild 6), in dem Sie den GPIO-Anschluss des Raspberry Pi konfigurieren können. Setzen Sie *Pin* auf GPIO2, *Type* auf *Digital output*, klicken Sie auf *Initialise pin state* und setzen Sie den Anfangszustand der LED auf „0“. Setzen Sie den Namen des Knotens auf *LED*. Abschließend klicken Sie auf *Done*, um den Konfigurationsbildschirm zu schließen.

Damit ist das Flussprogramm abgeschlossen. Klicken Sie auf *Deploy*, um das Programm zu kompilieren und einen ausführbaren Code zu generieren. Vergewissern Sie sich, ob in diesem Stadium des Entwurfs keine Fehlermeldungen mehr angezeigt werden, dann dürfen Sie auf die quadratische Schaltfläche auf der linken Seite des ON-Injektionsknotens klicken, um die LED einzuschalten. Wenn Sie dann auf die Schaltfläche des anderen Injektionsknotens klicken, wird die LED wieder dunkel.

Weitere Aspekte

An diesem sehr einfachen Beispiel haben wir gesehen, was Node-RED ist und wie es verwendet wird. Eine der Stärken von Node-RED ist, dass es von großen Communities unterstützt wird und dass es viele von Dritten entwickelte Nodes gibt, die kostenlos zur Verfügung stehen und in komplexen Projekten eingesetzt werden können. Zum Beispiel gibt es den Knoten *openweathermap*, der die lokalen Wetterbedingungen wie Temperatur, Feuchtigkeit, Luftdruck, Windgeschwindigkeit, Niederschlagsmenge und so weiter überall auf der Erde angibt. Ein Projekt mit einer Wettervorhersage lässt sich mit diesem und einigen zusätzlichen Knoten in weniger als einer Stunde erstellen. Würde man ein solches Projekt auf herkömmliche Weise mit „eigenen“ externen Sensoren realisieren, wäre dies viel komplizierter, zeitaufwändiger

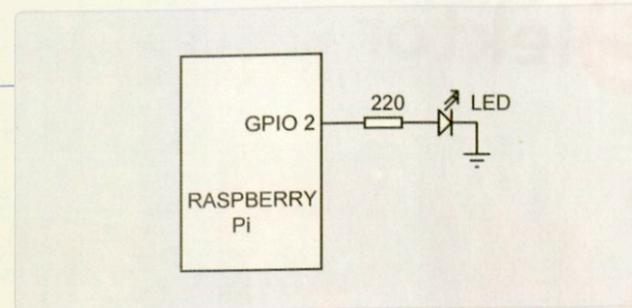


Bild 4. Der Schaltplan des Experiments.

und in der Regel auch erheblich kostspieliger. Node-RED unterstützt UDP- und TCP-basierte Kommunikation mit einem Knoten über eine WLAN-Verbindung. Diese Funktion ermöglicht es Benutzern, drahtlos Daten von entfernten Stellen über eine WLAN-Verbindung von vernetzten Geräten wie Mobiltelefonen, Tablets oder PCs zu senden und zu empfangen. Einige andere interessante Knoten, die kostenlos aus dem Internet installiert werden können, sind:

- > Dashboard
- > Bluetooth
- > Amazons Alexa
- > I²C- und paralleles LCD
- > Weltkarte
- > A/D-Wandler
- > Ultraschall-Sensor

Node-RED ist nicht nur für die den Raspberry Pi gedacht, sondern



IM ELEKTOR-STORE

Ein neues Buch des Autors mit dem Titel *With Node-RED: Design IoT Projects with Raspberry Pi, Arduino and ESP32* enthält eine große Zahl von Node-RED-Projekten mit den derzeit populärsten Mikrocontroller-Entwicklungsboards.

> Das Buch kann im Elektor Store bestellt werden unter: www.elektor.de/programming-with-node-red

> Ebenfalls erhältlich ist eine E-Book-Version: www.elektor.de/programming-with-node-red-e-book

kann (wenn auch eingeschränkt) mit den Prozessorboards Arduino und ESP32 oder Kombinationen von Raspberry Pi und Arduino oder Raspberry Pi und ESP32 verwendet werden.

Einer der sehr wichtigen Node-RED-Knoten ist der *function*-Knoten, der so konfiguriert werden kann, dass er mehrere Ausgänge hat. Benutzen Sie bei der Konfiguration dieses Knotens passende Programmcodes schreiben. So lassen sich mit dem *function*-Knoten aus den lokalen Wetterdaten, die der *openweathermap*-Knoten liefert, beispielsweise die lokalen Temperatur- und Feuchtigkeitsdaten extrahieren. Dies kann man leicht durch ein paar Befehle bei der Konfiguration des *function*-Knotens erreichen. Interessierte Leser bekommen weitere Infos unter den Weblinks [1]..[5] oder im Node-RED-Buch des Autors. ◀

200164-03

Anzeige

WEBLINKS

- [1] **Offizielle Website von Node-RED:** <https://nodered.org>
- [2] **Erste Schritte:** <https://projects.raspberrypi.org/en/projects/getting-started-with-node-red>
- [3] **Ein erstes Flussprogramm:** <https://nodered.org/docs/tutorials/first-flow>
- [4] **Schreibfunktionen:** <https://nodered.org/docs/user-guide/writing-functions>
- [5] **Programmierhandbuch:** <http://noderedguide.com/examples/>

SIGLENT® ONE STEP FOR US AND X PLUS FOR YOU

100 / 200 / 350 MHz Bandwidth

2 GSa/s Sampling Rate

Opt. Built-in 50 MHz AWG

10-bit Resolution Mode

200 Mpts/ch Memory Depth

Bode Plot & opt. Power Analysis

Setup für Bode-Plot Messung

SDS2000X PLUS Oscilloscopes