



Plug & Funk

LoRaWAN für IoT-Projekte: Einfach einsteigen mit TTN und Node-Red

Ihr persönliches Internet der Dinge oder Smart Home funkt per LoRaWAN weit über WLAN-Grenzen hinweg – sicher, zuverlässig und ohne Mobilfunkgebühren. LoRaWAN-Hardware ist nicht mehr teuer, mit Node-Red steht kostenlose, offene Software bereit und die für den Einstieg benötigte Netzinfrastruktur liefert das „The Things Network“.

Von Andrijan Möcker

Das Internet der Dinge – oder neu-deutsch „IoT“ (Internet of Things) – braucht Netzwerkanschluss. Jenseits der Reichweite einer WLAN-Basis kam dafür lange Zeit nur Mobilfunk infrage; dabei fallen jedoch Gebühren an. Mittlerweile ist die Langstrecken-Funktechnik LoRaWAN (Long Range Wide Area Network) jedoch ihrer „Bastelphase“ entwachsen. Wenn es in Ihrer Umgebung bereits eine Basisstation (Gateway) gibt, können Sie sofort loslegen. Falls nicht, richten Sie ab etwa 100 Euro Ihr eigenes Gateway ein, das die Brücke vom LoRaWAN ins Internet schlägt. Der verwendete Frequenzbereich bei 868 MHz ist in der gesamten EU lizenzfrei nutzbar. Dank des kostenfreien

LoRaWAN-Netzverbunds The Things Network (TTN) können Sie ohne komplexe Netzinfrastruktur einsteigen und Ihre Gateways ohne Sicherheitseinbußen mit anderen teilen.

Die kostenfreie visuelle Programmieroberfläche Node-Red hilft, erste Anwendungen unkompliziert umzusetzen. Außer Englischkenntnissen und LoRaWAN-Hardware benötigen Sie dafür eine Node-Red-Instanz, etwa auf einem Raspberry Pi oder alternativ eine MQTT-fähige Heimautomatisierungssoftware, mit der Sie bereits vertraut sind. Auf den Seiten 152 und 153 finden Sie zudem eine Auswahl direkt benutzbarer, fertiger LoRaWAN-Geräte, die für den Einstieg interessant sind.

(Gemeinsam) Loslegen

Anders als Smart-Home-Funkstandards wie Zigbee oder Z-Wave erlaubt LoRaWAN mehrere unabhängige Netzteilnehmer: Die Gateways erfüllen im LoRaWAN nur eine Durchleitungsfunktion; sie entschlüsseln die Pakete nicht und Geräte werden auf Netzebene registriert. Die Entschlüsselung passiert erst in der Serverinfrastruktur und für jedes Gerät mit

einem individuellen Schlüssel in der jeweiligen abgeschotteten „Application“ (Anwendung), die einem Nutzer fest zugeordnet ist. Das funktioniert auch im kostenfreien „The Things Network“ (TTN), in dem sich Nutzer einfach mit einer E-Mail-Adresse registrieren und es unabhängig vom Besitz eines eigenen Gateways nutzen können – Hauptsache, das TTN-Gateway eines anderen Nutzers ist in Reichweite. Wer selbst LoRaWAN-Server bereitstellt, kann die Anwendungsserver, auf denen die Schlüssel für die Geräte liegen, vom Netzserver trennen und sich so ein Gateway mit anderen teilen. Der Einfachheit halber geht dieser Artikel nur auf das TTN ein. Einen ausführlichen Artikel zu LoRaWAN-Grundlagen haben wir vor einiger Zeit verfasst [1].

LoRaWAN ist ideal geeignet, um im Zusammenschluss – ob privat, behördlich oder unternehmerisch – ein selbstverwaltetes IoT-Netz aufzubauen. Der Vorteil dabei: Sie tragen die Kosten des Netzausbaus nicht alleine und erreichen größere Netzabdeckung, als wenn Sie das Netz alleine installieren. Wollen Sie das Netz in Ihrer Region groß aufziehen, schauen Sie zunächst, ob Sie weitere Interessierte finden – ob als Firma oder privat spielt keine Rolle, denn auch im Verein kann das Vorhaben erfolgreich sein. Sprechen Sie am besten auch mit Ihrer Kommune; nicht selten kann sie gute Standorte anbieten und Digitalisierungsfördergelder abrufen, wenn das Netz auch öffentlich nutzbar ist, wie es etwa bei TTN der Fall ist.

Die Auswahl an LoRaWAN-Gateways ist groß und bedient jede Preiskategorie: Privat starten können Sie etwa mit dem Indoor-Gateway LPS8 von Dragino (circa 100 Euro) oder dem derzeit günstigsten Outdoor-Gateway wAP LoRa8 von Mikrotik (circa 140 Euro) [2]. Die Outdoor-Variante des LPS8, das DLOS8, gibt es mit und ohne LTE-Modem für 250 beziehungsweise 300 Euro. Das Lorix One liegt bei rund 500 Euro [3]. Jenseits der 500 Euro bekommt man professionelle Gateways mit Zusatzfunktionen wie GPS, LTE-Backup und integrierter USV von Herstellern wie Rakwireless, Cisco und Milesight (Ursalink). Aufgrund standardisierter Protokolle sind sie zum TTN und weiteren LoRaWAN-Servern kompatibel.

Wer weitere Akteure vom gemeinsamen Ausbau überzeugen will, sollte sich zunächst mit der Technik vertraut machen und Erfahrungen sammeln. Das geht unter Umständen sogar ohne ein eigenes Gate-

way. Auf der TTN Map und mit dem TTN Mapper können Sie prüfen, ob sich in Ihrer Nähe bereits ein Gateway befindet; beide Websites haben wir unter ct.de/yzrj verlinkt.

Gateway einrichten

Bevor Sie im The Things Network loslegen, sollten Sie sich zunächst mit der Konfigurationsoberfläche Ihres Gateways vertraut machen. Die Hersteller kochen hier meist ihr eigenes Süppchen. Prüfen Sie gegebenenfalls die Anleitung. Derzeit kommen bei LoRaWAN hauptsächlich zwei Protokolle zum Einsatz: der ältere „UDP Packet Forwarder“ und die „LoRa Basics Station“. Ersterer ist das üblichere, hat aber keine Authentifizierung und Transportverschlüsselung; letztere besitzt Authentifizierung und Verschlüsselung, machte in unseren Versuchen aber Probleme bei der Verbindungsherstellung nach Internetverbindungsabbrüchen. Da der UDP Packet Forwarder zuverlässiger funktioniert und LoRaWAN ohnehin Ende-zu-Ende-Verschlüsselung nutzt, beschränkt sich der Artikel auf dieses Protokoll.

Damit Sie Gateways im TTN anmelden und später Anwendungen erstellen können, benötigen Sie ein Nutzerkonto: Rufen Sie mit einem beliebigen Browser <https://thethingsnetwork.org> auf, klicken Sie oben rechts auf „Sign Up“ und erstellen Sie einen Account. Bestätigen Sie Ihre E-Mail-Adresse und klicken Sie im selben Browserfenster auf „Console“ – so nennt sich die TTN-Verwaltungsoberfläche. Auf der folgenden Seite werden Sie gefragt, welchen TTN-Server Sie aufrufen möchten; meist ist das Europa. Achten Sie darauf, nicht versehentlich auf „Legacy V2 Console“ zu klicken; sie gehört zur alten Netzversion und erlaubt ab Juni 2021 keine Gateway-Registrierung mehr.

In der Konsole angekommen, gehen Sie zunächst in den Gateway-Verwaltungsbereich und klicken auf „Add gateway“ um Ihr erstes Gateway hinzuzufügen. Vergeben Sie dafür eine beliebige „Gateway ID“ bestehend aus Kleinbuchstaben, Nummern und Bindestrichen. Anschließend fügen Sie die „Gateway EUI“ hinzu; diesen Extended Unique Identifier zeigen die meisten Gateways im Konfigurationsinterface an. „Gateway name“ und „Gateway description“ sind optional. Wenn Sie wollen, dass Ihr Gateway für andere TTN-Nutzer auf der Karte zu sehen ist, setzen Sie den „Gateway Status“ auf „Pu-

blic“. Der „Frequency plan“ ist in der Regel „Europe 863-870 MHz (SF9 for RX2)“. Bestätigen Sie die Einstellungen mit „Create gateway“.

Danach wechseln Sie zurück in die Konfigurationsoberfläche Ihres Gateways, wählen den UDP Packet Forwarder aus und tragen „eu1.cloud.thethings.network“ als Server sowie „1700“ als Zielport ein. Je nach Gateway muss der Forwarder händisch gestartet beziehungsweise neu gestartet werden. Einige Sekunden später sollten Sie in der TTN-Konsole unterhalb des Gateway-Namens eine aktuelle „Last seen“-Zeit sehen.

Sofern Sie sich dazu entschieden haben, Ihr Gateway öffentlich zu machen, wechseln Sie in der TTN-Konsole links ins Menü „Location“ und setzen den Haken bei „Publish location“. Besitzt Ihr Gateway einen GPS-Empfänger, ermittelt es die Position automatisch und Sie können den Haken bei „Update from status messages“ setzen. Wenn nicht, navigieren Sie auf der Karte zum Gateway-Standort, setzen den blauen Marker per Klick und ergänzen die Höhe im Feld „Altitude“. „Save changes“ bestätigt die Einstellungen.

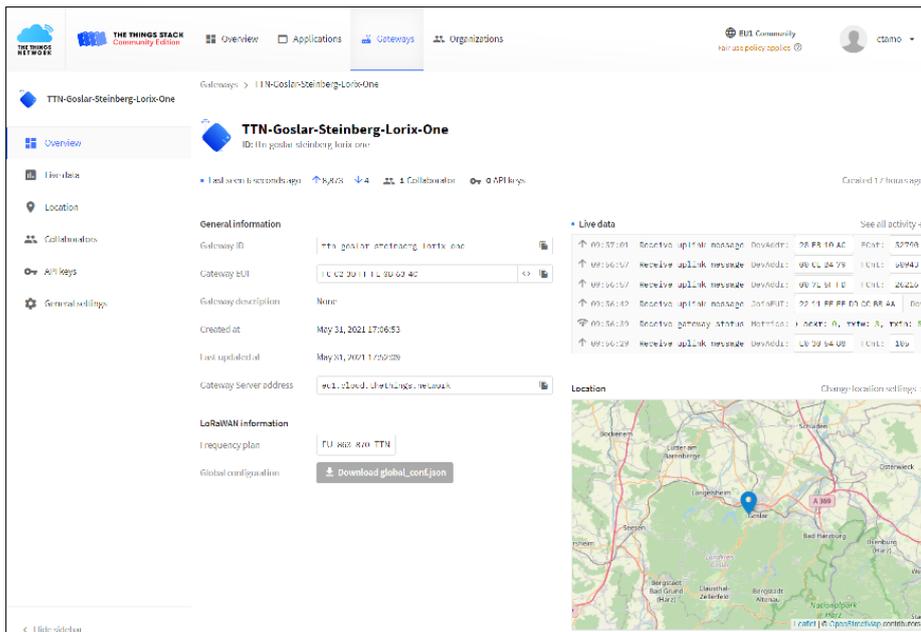
Anmeldeverfahren

LoRaWAN verwendet zwei Provisionierungsverfahren, um Endgeräte mit dem



Ein LoRaWAN-Gateway an einem exponierten Standort – hier das Lorix One [3] – deckt problemlos eine kleine Stadt ab und dient, mit passender Serverinfrastruktur und ohne Sicherheitsnachteile mehreren Firmen, Behörden und der Allgemeinheit.

Praxis | IoT-Einstieg mit LoRaWAN



Alle für die Konfiguration im TTN notwendigen Schritte erledigen Sie über die „Console“. Sie mag auf den ersten Blick etwas komplex wirken, ist jedoch gut strukturiert und, wenn man die Terminologie kennt, leicht zu bedienen.

Netz bekannt zu machen und Schlüssel auszutauschen: Over-the-Air-Activation (OTAA) und Activation by Personalization (ABP). OTAA ist das sicherere der beiden Verfahren, da je nach Endgerätefirmware bei jedem oder alle paar Sendevorgänge ein neuer Schlüssel ausgehandelt wird. Dazu schickt das Endgerät einen Join Request (JR) ans Netz und bekommt die Sitzungsschlüssel per Join Accept (JA) vom Gateway geschickt.

Bei OTAA werden die Application EUI, die Device EUI und der Application Key auf dem LoRaWAN-Server (des TTN etwa) eingetragen. Viele fertige Geräte kommen mit vorgenerierten OTAA-Zugangsdaten; in der Regel übernimmt man die Daten dann einfach vom Gehäuse oder einem Beipackzettel, trägt sie auf seinem LoRaWAN-Server ein und kann das Gerät anschließend benutzen. In seltenen Fällen besitzen Geräte noch keine Schlüssel und müssen über eine serielle Schnittstelle konfiguriert werden. Die gegebenenfalls benötigten Schlüssel generieren das TTN und die meisten anderen LoRaWAN-Server per Mausklick.

Auch bei ABP spielt es keine Rolle, ob die Zugangsdaten vom Hersteller oder vom Server kommen: Der Unterschied ist, dass bei ABP die Sitzungsschlüssel, die bei OTAA bei jeder Übertragung neu ausgehandelt werden, dauerhaft zum Einsatz kommen. Daraus ergeben sich einerseits

Sicherheitsprobleme, andererseits aber auch Vorteile: Wird der Schlüssel kompromittiert, können alle zuvor mitgeschnittenen Pakete entschlüsselt und falsche Pakete geschickt werden – ob das im jeweiligen Anwendungsfall ein Problem wäre, müssen Sie selbst beurteilen. ABP bedeutet jedoch auch, dass das IoT-Gerät unverzüglich seine Daten loswerden kann und nicht erst die Join-Prozedur durchlaufen muss. Das ist sinnvoll bei batteriebetriebenen Geräten oder solchen, die aus tiefen Kellern oder anderen stark abgeschirmten Umgebungen senden sollen. Denn nur weil das Gateway das IoT-Gerät empfängt, muss das nicht zwangsläufig umgekehrt funktionieren, zumal Gateways Join Accepts oft mit höherer und damit weniger robuster Datenrate versenden, um schnell wieder auf Empfang gehen zu können. ABP verwendet die Gerätezuordnung (Device Address), einen Netzsitzungsschlüssel (Network Session Key) zur Verschlüsselung auf Netzebene und einen Anwendungssitzungsschlüssel (App Session Key) zur Verschlüsselung auf Anwendungsebene.

Geräte hinzufügen

Unsere Gerätebeispiele auf den Seiten 152 und 153 sind, ausgenommen die RAK-Geräte mit USB-Anschluss, mit OTAA vorkonfiguriert, sodass die Einrichtung leicht fällt. Wir demonstrieren die Konfiguration

und spätere Verwendung in Node-Red anhand des Bewegungsmelders Tabs TBMS100-868. Der Prozess ist jedoch bei einem Großteil der LoRaWAN-Geräte sehr ähnlich.

Wechseln Sie in die TTN-Konsole und die Registerkarte „Applications“; mit „Add application“ starten Sie den Einrichtungsprozess. Die „Application ID“ (Kleinbuchstaben, Zahlen, Bindestriche) muss im TTN einzigartig sein – ist sie schon vergeben, können Sie etwa Ihren Benutzernamen anhängen. Name und Beschreibung sind optional. Klicken Sie auf „Create application“, um die Einstellungen zu bestätigen. Sollte Ihnen die Konsole danach nur noch einen Fehler anzeigen, laden Sie die Seite neu; in der neuen v3-Serversoftware werden derzeit noch Kinderkrankheiten behoben und nach einem Reload sollte die Anwendungskonsole erscheinen.

Über „Add end device“ fügen Sie ein neues Gerät hinzu, wofür Sie die OTAA- oder ABP-Daten bereithalten müssen. Lassen Sie das LoRaWAN-Gerät vorerst abgeschaltet, bis Sie alle Daten eingetragen haben; unter Umständen gibt es das Warten auf einen Join Accept sonst auf, bevor Sie alle erforderlichen Parameter eingetragen haben. Im folgenden Menü bietet die Konsole einige vorbereitete Geräteprofile an. Sofern für Ihr Gerät verfügbar, können Sie das Angebot wahrnehmen und die Zugangsdaten direkt eintragen. Für den Hersteller unseres Sensors gibt es noch keine Profile, daher wechseln wir in die Registerkarte „Manually“, wählen OTAA, die vom Hersteller im Datenblatt angegebene LoRaWAN-Version 1.0.3 und klicken auf „Start“. In den „Basic settings“ erfragt das TTN zunächst eine individuell einstellbare „End device ID“, die App EUI – gelegentlich fälschlich als Join EUI bezeichnet – und die DevEUI sowie optional Name und Beschreibung. In den „Network layer settings“ wählen wir nur den europäischen Frequenzplan mit „SF9 for RX2“ aus; beherrscht Ihr Gerät laut Datenblatt zusätzlich zu Class A auch Class B und C, haken Sie diese entsprechend an. In den „Join settings“ fehlt nur noch der „App Key“. Tragen Sie ihn ein und schließen Sie die Einrichtung mit „Add end device“ ab.

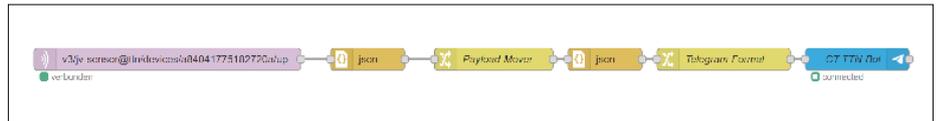
Die Konsole leitet Sie anschließend in die Geräteübersicht weiter. Rechts im Fenster „Live data“ können Sie den Datenverkehr des neuen Gerätes beobachten; schalten Sie es ein und warten Sie ab. In

der Regel steht wenige Sekunden nach dem Einschalten „Accept join-request“ und eine erste Nutzlast im Log. Wenn nicht, prüfen Sie die Schlüssel auf Korrektheit; unter Umständen ist auch das LoRaWAN-Gateway zu weit weg.

Für ABP ist der Prozess ähnlich: In der Registerkarte „manually“ wählen Sie „ABP“ sowie die korrekte LoRaWAN-Version und klicken sich durch den Registrierungsprozess. Sie können die „Device address“ und die Schlüssel entweder eintragen oder durch Klick auf die Pfeile rechts der Felder generieren lassen. Wie das Gerät damit provisioniert wird, verrät Ihnen die Herstellerdokumentation.

Decoder

Das erste Datenpaket in der Live-Ansicht enthält ausschließlich einige Metadaten wie Empfangsstärke und die Nutzlast in hexadezimaler Darstellung. Keine Sorge, damit müssen Sie nicht arbeiten: Mithilfe eines „Decoder“ genannten JavaScript-Schnipsels, den es für viele Geräte bereits gibt, verwandeln Sie die Hex-Nutzlast in ein menschenlesbares JSON-Objekt mit eindeutigen Beschreibungen. Das ist nötig, weil LoRaWAN-Geräte keine Objektnotationen mit Beschriftungen verwenden, um möglichst kurze Datenpakete zu senden. Stattdessen sind eine oder mehrere Nutzlastlängen sowie die Position der jeweiligen Werte in der Nutzlast festgelegt. Die Dekodierung in ein JSON-Objekt erfolgt nach Eintragung des Decoders bei jedem zukünftigen Datenpaket automatisch. Das JSON-Objekt wird dann an den MQTT- und HTTP-Schnittstellen



Dank MQTT-Schnittstelle ist die Integration von TTN-LoRaWAN-Geräten mittels Node-Red schnell gemacht. In diesem Beispiel wird das Datenpaket an einen Telegram-Bot weitergeleitet.

des TTNs ausgegeben – wie bei einer im Internet üblichen Anwendungsschnittstelle also.

Der schnellste Weg, um an ein Decoder-Skript für Ihr Gerät zu kommen, ist eine Suchmaschine: Suchen Sie etwa nach „<Modellbezeichnung> TTN decoder“ oder „<Modellbezeichnung> JavaScript decoder“. Nicht immer stellen die Gerätehersteller Decoder für Ihre Geräte bereit; oft sind Sie aber nicht der Erste, der einen benötigt. Manchmal veröffentlicht stattdessen ein Besitzer des gleichen Gerätes beispielsweise etwas Passendes, etwa in einem GitHub-Repository. Für den TBMS100 wurden wir im Repository des Digitalisierungsvereins „dasdigidings“ fündig (ct.de/yzrj).

Um den Code hinzuzufügen, wechseln Sie in der TTN-Konsole der jeweiligen Anwendung links in „Payload formatters“ auf „Uplink“. Der Typ „JavaScript“ öffnet ein Textfeld, in das Sie das Skript hineinkopieren können. Sollten Sie mehrere Geräte unterschiedlichen Typs zu einer Anwendung hinzufügen wollen, können Sie in der gleichnamigen Registerkarte des jeweiligen Gerätes einen individuellen

Decoder einbinden. Ohne Änderung verwenden alle Geräte in der Anwendung denselben Decoder.

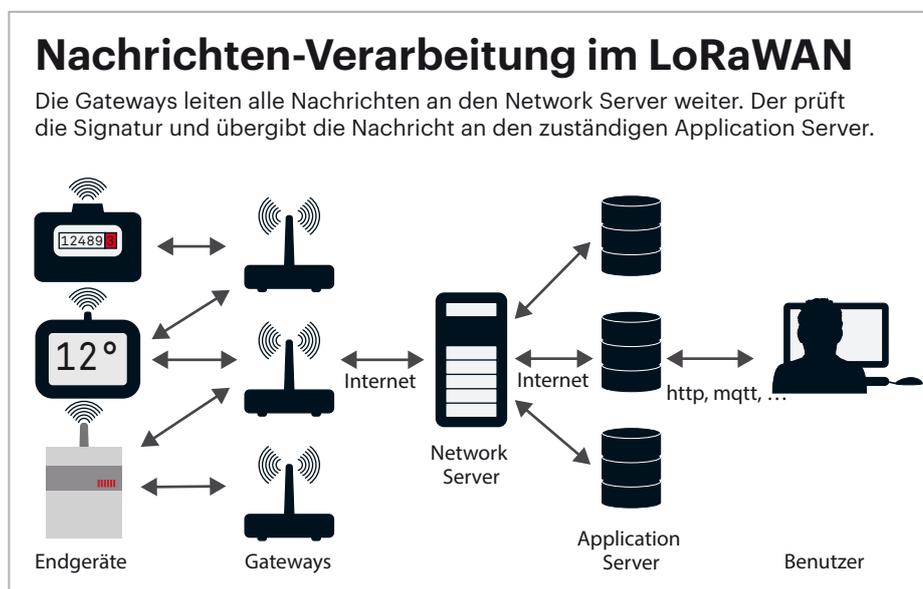
Das dekodierte Datenpaket des TBMS100 sieht beispielsweise so aus:

```
{
  „battery“: 3.6,
  „capacity“: 0,
  „port“: 102,
  „roomStatusCount“: 44,
  „roomStatusOccupied“: false,
  „roomStatusTime“: 5,
  „temperature“: 26
}
```

Node-Red

Mit der visuellen Programmieroberfläche Node-RED und dem Telemetrieprotokoll MQTT können Sie besonders einfach Logiken erstellen, ohne sich dafür mit einer Textprogrammiersprache beschäftigen zu müssen. Wenn Sie Node-Red und MQTT noch nicht kennen, lesen Sie zunächst unsere Einsteigerartikel [4,5]. Alternativ klappt die Kommunikation mit TTN auch mit jeder anderen Programmiersprache oder Automatisierungssoftware, die HTTP(S) oder MQTT unterstützt. Das folgende Beispiel zeigt jedoch ausschließlich, wie Sie in Node-Red mit MQTT vorgehen. Die Dokumentation der Anwendungsschnittstellen zur Nutzung mit anderen Sprachen und Plattformen finden Sie über ct.de/yzrj.

Die Einrichtung der MQTT-Schnittstelle ist simpel und bedarf nur etwas Kopier- und Klickarbeit: Erstellen Sie der Übersicht halber zunächst einen neuen Node-RED-Flow über das kleine Plus rechts am Ende der Registerkartenreihe. Ziehen Sie dann ein „mqtt in“-Node in den Flow und öffnen Sie dessen Einstellungen über einen Doppelklick. In der Serverliste wählen Sie „Neuen mqtt-broker hinzufügen“ und klicken auf den Stift, um die Serverkonfiguration anzulegen. Tragen Sie einen passenden Namen, etwa „TBMS100“, und die generischen Verbindungsparameter



Praxis | IoT-Einstieg mit LoRaWAN



Dragino LGT-92 GNSS-Tracker

Der LGT-92 von Dragino ist eine Kombination aus GNSS-Tracker (GPS, GLONASS usw.) und Notfallknopf. Die vorinstallierte Firmware genügt für viele Anwendungsfälle, aber man kann sich auch selbst am frei verfügbaren Quellcode probieren. Einen JavaScript-Payload-Decoder stellt Dragino bereit.

Die Positions-, Keep Alive- und Alarmintervalle sind frei konfigurierbar. Bei Betätigung des Knopfes schickt der LGT-92 zusätzlich ein Alarm-Bit mit, das sich per Downlink-Paket wieder zurücksetzen lässt. So kann der Tracker nicht nur Notrufsender sein, sondern etwa auch Türöffner mit Geofencing und eindeutiger Identifizierung [6].

Preis: 45 Euro



Mclimate Vicki Heizkörperthermostat

Das ungarische Start-Up Mclimate bietet sozusagen die eierlegende Wollmilchsau der Funk-Heizkörperthermostate. Das Vicki getaufte Gerät mit M30 x 1,5-Gewinde kann laut technischer Dokumentation (ct.de/yzrj) bis ins kleinste Detail konfiguriert werden: Es besitzt einen eigenen Temperatursensor, kann aber auch per LoRaWAN Temperaturwerte empfangen und sich danach richten; der Besitzer kann unter anderem einstellen, ob und wie lange Vicki offene Fenster erkennen und wie schnell der Motor das Heizkörperventil bewegen soll.

Die technische Dokumentation ist frei verfügbar, sodass Vicki überall integriert werden kann. Mclimate liefert aber auch eine Smartphone-App. Das Gerät befindet sich derzeit im Test bei c't.

Preis: 85 Euro



RAK 7204 Atmosphärensensoren

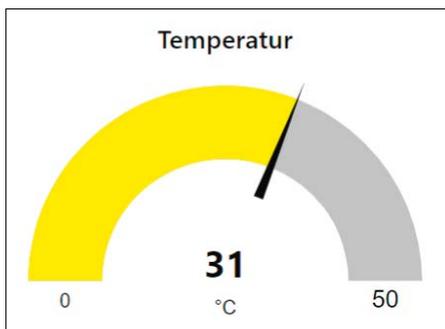
Der RAK 7204 von Rakwireless ist ausschließlich dafür gedacht, die Umgebungsluft zu messen: Luftdruck, Temperatur, relative Luftfeuchte und Gaswiderstand. Aus den Werten kann man den „Indoor Air Quality Index“ berechnen, der die Luftqualität im Raum bewertet und informiert, ob gelüftet werden sollte.

Der Sensor ist nicht ganz günstig, kommt aber dafür mit einer leicht austauschbaren Lithium-Batterie (2 Jahre Laufzeit bei 15 Minuten Sendeintervall) und kann per USB-Anschluss (im Gehäuse) konfiguriert werden. Zum Einsatz kommt Boschs BME680 der dank Werkskalibrierung besonders präzise sein soll.

Preis: 75 Euro

ein: Der Server ist „eu1.cloud.thethings.network“, der Port 8883; zudem muss die Verbindungsverschlüsselung (SSL/TLS) aktiviert werden.

Dafür benötigt MQTT eine TLS-Konfiguration. Klicken Sie neben dem TLS-Feld auf den Stift, um diese anzulegen: Vergeben Sie einen Namen für die Konfi-



Node-Red-Dashboard liefert mit wenigen Klicks schicke Oberflächen [7].

guration, tragen Sie „eu1.cloud.thethings.network“ als Servername ein, laden Sie das Root-Zertifikat von Let's Encrypt in „CA-Zertifikat“ hoch (Download über ct.de/yzrj) und bestätigen Sie mit „Hinzufügen“.

Stellen Sie sicher, dass die neue TLS-Konfiguration in den MQTT-Einstellungen ausgewählt ist, dann wechseln Sie in die Registerkarte „Sicherheit“. Der Benutzername für Ihre Anwendung lautet „<Anwendungs-ID>@ttn“, also beispielsweise „tbms100magazinct@ttn“. Das Passwort ist ein API-Schlüssel, den Sie über die TTN-Konsole der jeweiligen Anwendung erhalten: Links im Menü finden Sie „API keys“; über „Add API key“ fügen Sie einen neuen Schlüssel hinzu. Vergeben Sie einen eindeutigen Namen, etwa „MQTT-Node-Red“ und nur Rechte, die Sie auch wirklich benötigen (Grant individual rights); hier etwa „Write downlink application traffic“ und „Read application traffic (uplink and downlink)“. API-Schlüs-

sel können später nicht erneut angezeigt werden. Kopieren Sie den Schlüssel aus dem Popup direkt in die MQTT-Konfiguration und speichern Sie diese (Hinzufügen). Mit „I have copied the key“ wird dieser übernommen und die MQTT-Konfiguration ist bereit.

Im MQTT-Node wählen Sie, falls nicht automatisch geschehen, die eben angelegte Serverkonfiguration. Jetzt fehlt nur noch der richtige Topic, um die Anwendungsdaten in Node-Red verfügbar zu machen. Der setzt sich wie folgt zusammen: „v3/<Anwendungs-ID>@ttn/devices/<Device-ID>/up“; beispielsweise: „v3/tbms100magazinct@ttn/devices/tbms100-1/up“. Die Device-IDs finden Sie immer im „Overview“ der TTN-Anwendung oder in der Liste „End devices“. Bestätigen Sie die Einstellungen mit „Fertig“, dann verbinden Sie ein JSON-Node mit dem MQTT-Node und das JSON-Node wiederum mit einem Debug-Node.



RAK 7200 GNSS Tracker

Der RAK 7200 ist ein gerade mal 86 Gramm leichter GNSS-Tracker von Rakwireless. Konfiguriert wird er per USB, und der Hersteller hat sich mit der Dokumentation größte Mühe gegeben: Sämtliche technische Details sind offengelegt, inklusive Schaltpläne. Dabei unterschlägt der Hersteller jedoch, dass der GNSS-Empfänger nicht nur GPS, sondern auch GLONASS, Galileo, BeiDou und QZSS empfangen kann.

Dass Rakwireless nur eine Windows-Software zur Konfiguration liefert, tut der Sache keinen Abbruch: Sie schickt auch nur AT-Befehle über die Schnittstelle; die hat Rakwireless gut dokumentiert, sodass die Konfiguration aus jedem Betriebssystem über eine serielle Konsole klappt.

Preis: 55 Euro



tabs TBMS100 Bewegungsmelder

Der TBMS100 aus der tabs-Reihe von Browan dient der Raumüberwachung im Innenbereich; etwa um zu erfahren, ob ein Sitzungsraum tatsächlich belegt ist. Laut Hersteller erfasst der passive Infrarot-Bewegungsmelder auf 123° horizontal und 93° vertikal Wärmequellen in bis zu sieben Metern Entfernung.

Das Gerät überträgt zudem die Batteriespannung, Temperatur, die Zeit seit der letzten Auslösung in Minuten und die Anzahl der bisherigen Auslösungen. Wann der Sensor den Raum nach der letzten Bewegung wieder als „frei“ meldet, kann der Administrator frei konfigurieren.

Die Stromversorgung übernimmt eine leicht tauschbare ER14250-Lithium-Zelle.

Preis: circa 50 Euro



tabs TBWL100 Wassermelder

Der TBWL100 von Browan sieht auf den ersten Blick aus wie ein einfacher Wassermelder. Er kann aber mehr: Der TBWL100 enthält zusätzlich zum Wassersensor noch digitale Temperatur- und Luftfeuchtefühler. So meldet der Sensor nicht nur den Kontakt mit Wasser, sondern auch schlagartige Temperatur- und Luftfeuchteveränderungen (Schwelle: 2 °C/5 %).

Alarmierende Änderungen meldet er sofort, ansonsten gibts einmal pro Stunde ein Statuspaket mit allen Werten, sodass er auch als Atmosphärenfühler dient, wenn nicht gerade irgendwo Wasser ausläuft. Die Parameter sind fest eingestellt.

Wie die anderen Geräte der Tabs-Reihe arbeitet er mit einer ER14250-Lithium-Zelle. Das Gehäuse kann man leicht zum Tausch öffnen.

Preis: 60 Euro

Anschließend übernehmen Sie den Flow mit „Deploy“.

Wechselt das MQTT-Node nicht nach einigen Sekunden in den Status „verbunden“, prüfen Sie die Zugangsdaten erneut. Klappt alles, lösen Sie an Ihrem LoRaWAN-Gerät eine Übertragung aus und beobachten die Debug-Ausgabe (Käfer-Registerkarte oben rechts). Das eingehende Datenpaket enthält neben dekodierter Nutzlast allerhand Metadaten; die dekodierte Nutzlast finden Sie unter „uplink_message/decoded_payload“ zum Ausklappen.

Derart lange Objektpfade, etwa „msg.payload.uplink_message.decoded_payload.environment.temperature“, machen das Arbeiten in Node-Red nicht gerade komfortabel. Ein „Change“-Node löst das Problem unkompliziert, indem es die dekodierte Nutzlast direkt als Objekt in „msg.payload“ kopiert: Ziehen Sie ein neues Change-Node in den Flow und verbinden Sie dessen Eingang mit dem JSON-Node hinter dem

MQTT-Node Ihres LoRaWAN-Geräts. Das Change-Node benötigt nur eine Anweisung: „Bewegen“ (Move), „msg.payload.uplink_message.decoded_payload“ auf „msg.payload“. Aus dem langen Pfad wird so „msg.payload.environment.temperature“.

Schicke Darstellung und Reichweitentests

„Hinter“ dem Change-Node können Sie dann mit den Daten weiterarbeiten, wie Sie es in Node-Red gewohnt sind – woher sie kommen, spielt keine Rolle mehr. Entwickeln Sie etwa mit Node-Red-Dashboard schicke Darstellungen, um potenzielle Mitstreiter zu beeindrucken [7].

Um sicherzustellen, dass Ihr TTN-Gateway – ob gemeinsam oder alleine aufgebaut – die für Ihr Projekt notwendige Abdeckung liefert, sollten Sie vor einer großen Gerätebestellung eine Abdeckungsüberprüfung machen. Das geht mit dem TTN Mapper [8]. Danach steht dem großen,

selbst geschaffenen IoT-Ausbau nichts mehr im Wege. (amo@ct.de) **ct**

Literatur

- [1] Jan Mahn, Langstreckenfunk, IoT-Funk LoRaWAN: für kleine Datenmengen und hohe Reichweiten, c't 10/2019, S. 140
- [2] Andrijan Möcker, Günstige LoRaWAN-Basis, Dragino LPS8: LoRaWAN-Gateway für 110 Euro, c't 3/2021, S. 64
- [3] Andrijan Möcker, IoT Horchposten, Loric One LoRaWAN-Gateway im Langzeittest, c't 12/2021, S. 68
- [4] Jan Mahn, Reaktionsmaschine, Einstieg in Heimautomation mit Node-Red, c't 15/2018, S. 142
- [5] Jan Mahn, Weltsprache, Das Protokoll MQTT für robusten Datenaustausch in Industrie und Hausautomation, c't 6/2018, S. 164
- [6] Andrijan Möcker, GNSS-Notfallknopf, LoRaWAN-Tracker mit Notrufunktion, c't 5/2021, S. 64
- [7] Merlin Schumacher, Rotfront!, Grafische Oberflächen für Node-Red entwickeln, c't 2/2019, S. 160
- [8] Andrijan Möcker, IoT-Kartografie, LoRaWAN-Abdeckung mit dem Webdienst TTN Mapper darstellen, c't 12/2021, S. 160

Dokumentationen, Decoder und TTN-Karten: ct.de/yzrj